Electronic Journal of Linear Algebra, ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 37, pp. 583-597, August 2021.

A PROJECTIVE APPROACH TO NONNEGATIVE MATRIX FACTORIZATION[*]

PATRICK GROETZNER[†]

Abstract. In data science and machine learning, the method of nonnegative matrix factorization (NMF) is a powerful tool that enjoys great popularity. Depending on the concrete application, there exist several subclasses each of which performs a NMF under certain constraints. Consider a given square matrix A. The symmetric NMF aims for a nonnegative low-rank approximation $A \approx XX^T$ to A, where X is entrywise nonnegative and of given order. Considering a rectangular input matrix A, the general NMF again aims for a nonnegative low-rank approximation to A which is now of the type $A \approx XY$ for entrywise nonnegative matrices X, Y of given order. In this paper, we introduce a new heuristic method to tackle the exact nonnegative matrix factorization problem (of type $A = XY$), based on projection approaches to solve a certain feasibility problem.

Key words. Nonnegative matrix factorization, Symmetric nonnegative matrix factorization, Low-rank approximation, Completely positive matrices.

AMS subject classifications. 15B10, 15A23, 15B48.

1. Introduction.

Throughout the article, let $\mathbb{R}^n_+$, resp. $\mathbb{R}^{n\times m}_+$ denote the sets of entrywise nonnegative vectors and matrices, respectively. We write $\mathcal{S}_n$ for the set of symmetric matrices of order n and let $\|\cdot\|_F$ denote the Frobenius norm.

Nonnegative matrix factorization (NMF) aims for a nonnegative approximation to a given matrix that can be at the same time a low-rank-approximation of given rank. To be more precise, the NMF considers the following setting, cf. [14, Equation 3] or [15, Equation 1.1].

Definition 1.1. *Let $A \in \mathbb{R}^{n\times m}_+$ and $k \ll \min\{n, m\}$, then the solution matrices $X \in \mathbb{R}^{n\times k}_+$ and $Y \in \mathbb{R}^{k\times m}_+$ of the problem*

$$\min_{X\in\mathbb{R}^{n\times k}_+,\ Y\in\mathbb{R}^{k\times m}_+} \|A - XY\|^2_F,$$

yield a NMF XY of A.

Note that a NMF is not unique in general, since for any NMF XY of A and for any nonnegative matrix Z with Z^{-1} being entrywise nonengative as well, $\tilde{X} := XZ$ and $\tilde{Y} := Z^{-1}Y$ also provide a NMF of A. Adding further constraints to the problem in Definition 1.1 or slight changes in the objective function lead to various specially structured NMF problems. If we allow A and one of the matrices X, Y to have negative entries, this defines the *Semi NMF*, cf. [33, Section 2.2]. This problem is motivated by data clustering. For the *Sparse NMF*, cf. [33, Sections 2.7-2.9], we add the (possibly weighted) penalty term $\sum_{i,j} Y_{ij}$ to the objective function to ensure sparsity of the matrix Y. In case $\operatorname{rank}(A) \le k \le \min\{m, n\}$, considering the problem

$$\text{find}\{X \in \mathbb{R}^{n\times k}_+, Y \in \mathbb{R}^{k\times m}_+ \mid A = XY\}, \tag{1.1}$$

defines the so-called *exact NMF*, cf. [31]. In the context of machine learning and data mining, the exact NMF can be used as a tool in mining and image processing, cf. [23]. Moreover, the exact NMF is of theoretical

[*]Received by the editors on May 28, 2019. Accepted for publication on August 19, 2021. Handling Editor: Michael Tsatsomeros.

[†]University of Augsburg, 86135 Augsburg, Germany (patrick.groetzner@math.uni-augsburg.de).

importance: In [32] it forms the basis for the NP-hardness proof of the NMF. As a key result of this article, we will see a new heuristic method to obtain an exact NMF. The minimal integer k such that there exists $X \in \mathbb{R}_+^{n \times k}, Y \in \mathbb{R}_+^{k \times m}$ such that $A = XY$ is called the nonnegative-rank of A denoted by $\operatorname{rank}_+(A)$. It is easy to see that $\operatorname{rank}_+(A) \leq \min\{m, n\}$.

In addition, the following symmetric case can be seen as a special case of the problem in Definition 1.1.

Let $A \in \mathbb{R}_+^{n \times n}$ be symmetric. To introduce the symmetric nonnegative matrix factorization (SymNMF) of A, consider the following setting, cf. [5], [18].

Definition 1.2. *Given $A \in \mathcal{S}_n$ and $k \ll n$, the solution matrix $X \in \mathbb{R}_+^{n \times k}$ of*

$$\min_{X \in \mathbb{R}_+^{n \times k}} \|A - XX^T\|_F^2,$$

yields a symmetric nonnegative matrix factorization XX^T of A.

The SymNMF is related to data clustering, particularly Kernel K-means clustering and Laplacian-based spectral clustering, as discussed in [10]. As a concrete example, it can be used to analyze the structure of a given data set, like facial poses as shown in [17] or heterogeneous microbiome data, as introduced in [25].

There exist several approaches to compute a SymNMF of a given matrix A and given order k. For an algorithm based on certain update rules, the reader is referred to the approach in [10]. Newton-like methods for SymNMF can be found in [20, Section 3]. Borhani et al. [5] introduce an accelerated proximal gradient method and an alternating direction approach. All of these approaches are heuristic in the sense that convergence is not guaranteed.

Coming back to the more general framework of NMF in Definition 1.1, a broad field of applications is connected to this problem. For instance, some aspects in environmental [28] and meteorological sciences [30] demand an NMF. Moreover, the NMF appears in the analysis of financial data [11] and biomedical applications [6], where it helps to classify cancer cells.

NMF itself can be seen as a special subclass of so called constrained low-rank matrix approximation problems as introduced in [14]. Therefore, various applications of the NMF approach are related to this topic. One very illustrative application is hyperspectral imaging, cf. [14], where every pixel of a hyperspectral image is represented via more than 100 channels which correspond to deeper information of several wavelengths of the image, some of them invisible to the human eye. This approach boils down to the NMF framework. See also [13].

Moreover, in the context of data science, NMF can be used for so called intelligent data analysis, as shown in [8]. Especially when the quantities are known to be nonnegative, for example, due to physical laws, NMF can be used to determine part-based representations of given data. Here a concrete example, again given in [8], is educational data mining. For a survey on this topic, the reader is referred to [26]. Here the goal is to collect, store, and analyze data obtained from learning and evaluation processes of students.

Other possible applications are multi-document summarization, cf. [7] and analysis of magnetic resonance spectroscopy data, cf. [21]. As already mentioned for the symmetric case, NMF is closely related to data clustering, cf. [22].

The most popular approach in the context of NMF is to use certain multiplicative update rules as first introduced in [24], see also [10]. Here we should note that we can not modify zero entries and the method

Electronic Journal of Linear Algebra, ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 37, pp. 583-597, August 2021.

is not guaranteed to converge to a stationary point. The authors in [4] provide a summary of the common NMF methods, like gradient descent methods, where only little can be said about the convergence of these methods, see also [18, Section 3.3]. Here the idea is to rewrite the NMF problem as a convex problem over a nonconvex set. Further, adding the nonnegativity projection makes the analysis even more difficult. Alternating least squares methods, as in [28], are dealing with the decompostition matrices X and Y in an alternating manner. Here no general convergence result is known.

In the context of exact NMF as in (1.1), the authors in [31] provide several heursitics to solve this problem. One motivated by simulated annealing, another one based on the greedy randomized adaptive search procedure and a hybridization of the two heuristics. In contrast to the existing literature, this paper introduces a new view on the NMF setting in a different alternating manner and, for the symmetric case, shows the connection to the so-called completely positive matrices in Section 2. In Section 3, we will formulate the NMF and the SymNMF problem as feasibility problems which pave the way to new approaches working with so called alternating projection frameworks. For the general rectangular case in Section 4, we extend the results for the symmetric case introduced in [16] and under the assumption that the best low-rank approximation is nonnegative, we will show that this extension may additionally return NMFs. Especially, we will see a new heuristic method to derive exact NMF. In addition, numerical experiments will be shown in Section 5 to illustrate the performance of the method in concrete settings and we will see a comparison of the heuristic method introduced in this paper to the exact NMF heuristics in [31].

2. Completely positive matrices and their relation to symmetric nonnegative matrix factorization.

A symmetric matrix $A \in \mathbb{R}^{n\times n}$ is called completely positive if there exists an entrywise nonnegative matrix $X \in \mathbb{R}^{n\times r}$ such that $A = XX^T$. We call such a factorization a cp-factorization of A. The set of all completely positive matrices,

$$\mathcal{CP}_n := \{A \in \mathbb{R}^{n\times n} \mid A = XX^T \text{ where } X \in \mathbb{R}^{n\times r}, X \geq 0\} = \operatorname{conv}\{xx^T \mid x \in \mathbb{R}^n_+\},$$

is a proper cone whose extreme rays are the rank-one matrices xx^T with $x \in \mathbb{R}^n_+$, cf. [1]. The minimal possible number of columns r in the factorization matrix X is called the cp-rank $\operatorname{cpr}(A)$ and in general we have $\operatorname{cpr}(A) \gg \operatorname{rank}(A)$. Note that it may well happen (and often does) that $\operatorname{cpr}(A) > n$. Thus, completely positive matrix factorization is a special case of the SymNMF, albeit without the low rank constraint. In a cp-factorization, we have $k \geq \operatorname{cpr}(A)$ and for the SymNMF, $k \leq n$ is assumed.

Nevertheless, the problem in Definition 1.2 can be rewritten as

$$\min_{\substack{B\in\mathcal{CP}_n\\ \operatorname{cpr}(B)=k}} \|A - B\|_F^2.$$

Thus, the SymNMF seeks for the best completely positive approximation of cp-rank k to A. Since it is not possible to compute the cp-rank of a given completely positive matrix in general, cf. [2], we will use the rank of the matrix as a lower bound for the cp-rank. Hence, in case $k \leq \operatorname{rank}(A)$, we will try to factorize a rank-k approximation of A instead of A itself.

3. Nonnegative matrix factorization as a feasibility problem.

To introduce our method to determine a NMF or a SymNMF of a given matrix A, a first step is to compute the best rank-k-approximation of A. Here we use the well-known theorem by Eckart and Young in [12] which was proven to hold for any unitarily invariant norm by Mirsky in [27]. Let $\mathcal{O}_k$ denote the set of orthogonal matrices of order k and for a given matrix A, let A_{i*} resp. A_{*j} denote the i-th row resp. the j-th column of A.

Theorem 3.1. *Let $A \in \mathbb{R}^{n\times m}$ with $\operatorname{rank}(A) = l$ and consider its singular value decomposition $A = U\Sigma V^T$, where $U \in \mathcal{O}_n$, $V \in \mathcal{O}_m$,*

$$\Sigma = \left(\begin{array}{ccc|ccc} \sigma_1 & & & & \vdots & \\ & \ddots & & \cdots & 0 & \cdots \\ & & \sigma_l & & \vdots & \\ \hline & \vdots & & & \vdots & \\ \cdots & 0 & \cdots & \cdots & 0 & \cdots \\ & \vdots & & & \vdots & \end{array}\right) \in \mathbb{R}^{n\times m},$$

$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_l > 0$ are the singular values of A. So A can be written as

$$A = \sum_{j=1}^{l} \sigma_j U_{*j} V_{*j}^T.$$

Let $k \leq l = \operatorname{rank}(A)$, then

$$A_k := \sum_{j=1}^{k} \sigma_j U_{*j} V_{*j}^T,$$

is the best rank-k approximation (in the Frobenius norm) of A, i.e.

$$A_k \in \mathit{Argmin}\left\{ \|A - X\|_F^2 \;\middle|\; X \in \mathbb{R}^{n\times m} \text{ with } \operatorname{rank}(X) \leq k \right\},$$

with corresponding minimal value

$$\|A - A_k\|_F^2 = \sum_{j=k+1}^{m} \sigma_j^2.$$

Moreover, if $\sigma_k > \sigma_{k+1}$, then A_k is the unique global minimizer.

In the following A_k will denote the best rank-k-approximation to A. The subsequent lemmas now indicate that it is sufficient to determine a nonnegative factorization of A_k to obtain an (exact) NMF or a SymNMF in the symmetric case.

Lemma 3.2. *Let $A \in \mathbb{R}_+^{n\times m}$. Further let A_k be as in Theorem 3.1. Then any factorization $A_k = XY$ with $X \in \mathbb{R}_+^{n\times k}$ and $Y \in \mathbb{R}_+^{k\times m}$ gives a NMF of A. In case $k \geq \operatorname{rank}(A)$, the factorization XY is an exact nonnegative matrix factorization of A.*

Proof. Due to Theorem 3.1, $A_k = XY \in \operatorname{Argmin}\left\{ \|A - Z\|_F^2 \;\middle|\; Z \in \mathbb{R}^{n\times m} \text{ with } \operatorname{rank}(Z) \leq k \right\}$ and since $X \in \mathbb{R}_+^{n\times k}$ and $Y \in \mathbb{R}_+^{k\times m}$, we get XY as a nonnegative factorization of A. Note that in case $k \geq \operatorname{rank}(A)$, we have $A_k = A$, completing the proof. □

Thus, to obtain an (exact) NMF of A, it is sufficient to factorize $A_k = XY$, where $X \in \mathbb{R}_+^{n\times k}$ and $Y \in \mathbb{R}_+^{k\times m}$.

Lemma 3.3. *Let $A \in \mathcal{S}_n$, $k \leq n$, and let A_k as in Theorem 3.1. Further assume that $A_k \in \mathcal{CP}_n$ with $\operatorname{cpr}(A_k) \leq k$. Then any cp-factorization $A_k = XX^T$ with $X \in \mathbb{R}^{n\times k}$ of A_k is a solution to the problem in Definition 1.2 and hence a SymNMF of A of rank k.*

Proof. Let $A_k = XX^T$ be a cp-factorization of A_k with $X \in \mathbb{R}^{n\times k}$. Then, due to Theorem 3.1, $XX^T \in \operatorname{Argmin}\left\{\|A-Z\|_F^2 \;\middle|\; Z \in \mathbb{R}^{n\times m} \text{ with } \operatorname{rank}(Z) \le k\right\}$ and since $X \in \mathbb{R}_+^{n\times k}$, we get that XX^T is a SymNMF of A. ∎

To compute a SymNMF, it is therefore sufficient to find a cp-factorization of A_k of order $n \times k$. To obtain a nonnegative factorization of A_k as in Lemmas 3.2 or 3.3, we will start with an arbitrary initial factorization of A_k. Let $A = U\Sigma V^T$ be the SVD of A. In the rectangular case and if $k \le \operatorname{rank}(A)$, we use the SVD $A_k = U_k\Sigma_k V_k^T$, where the matrices $U_k,\ \Sigma_k,\ V_k$ are the truncated versions of U, Σ and V.

$$X_k := U_k\sqrt{\Sigma_k} \in \mathbb{R}^{n\times k} \text{ and } Y_k := \sqrt{\Sigma_k}V_k^T \in \mathbb{R}^{k\times m}, \tag{3.2}$$

now defines an initial factorization $A_k = X_kY_k$ which is not necessarily entrywise nonnegative. In case $\operatorname{rank}(A) \le k \le \min\{m,n\}$, we consider

$$X_k := U\sqrt{\Sigma} \in \mathbb{R}^{n\times k} \text{ and } Y_k := \sqrt{\Sigma}V^T \in \mathbb{R}^{k\times m}, \tag{3.3}$$

as our initial factorization where again negative entries may occur.

For the symmetric case, one can for instance use the Cholesky decomposition $A_k = X_kX_k^T$ where X_k is a lower triangular matrix, or the spectral decomposition $A_k = V\Sigma V^T$ by setting

$$X_k := V\Sigma^{\frac{1}{2}}, \tag{3.4}$$

to obtain an initial factorization. Again, if $k \ge \operatorname{rank}(A)$, we take $A_k = A$.

In both settings, we assume the the matrices X_k and Y_k to have negative entries. If this is not the case, we already generated a NMF or a SymNMF, respectively. Our goal now is to transform the initial factorizations into nonnegative ones.

First, we focus on the symmetric case, where we use the following tool to compute a completely positive factorization of A_k, cf. [16, Lemma 2.5].

Lemma 3.4. *Let $X_k, Z_k \in \mathbb{R}^{n\times k}$. Then $X_kX_k^T = Z_kZ_k^T$ if and only if there exists $Q \in \mathcal{O}_k$ with $X_kQ = Z_k$.*

So to transform the factorization $A_k = X_kX_k^T$ into a nonnegative factorization $A_k = Z_kZ_k^T$, we have to we solve the following feasibility problem:

$$\begin{array}{ll} \text{find} & Q \\ \text{s.t.} & X_kQ \ge 0 \\ & Q \in \mathcal{O}_k. \end{array} \tag{3.5}$$

This problem is feasible if and only if $A_k \in \mathcal{CP}_n$ with $\operatorname{cpr}(A_k) \le k$. Introducing the polyhedral cone $\mathcal{P} := \{Q \in \mathbb{R}^{k\times k} \mid X_kQ \ge 0\}$, we rewrite (3.5) as

$$\text{find } Q \in \mathcal{O}_k \cap \mathcal{P}. \tag{3.6}$$

Hence, we reduced the task to compute a SymNMF to solving Problem (3.6).

In the general nonsymmetric case, Lemma 3.4 does not apply, as the following example substantiates. As it turns out, we need to replace orthogonal matrices by a more general tool.

Example 3.5. Let

$$X = \begin{pmatrix} 6 & 0 \\ 0 & 6 \end{pmatrix},\ Y = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix},\ G = \begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix} \text{ and } H = \begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix}.$$

Then we have $XY = GH$. Moreover, observe that the equations $XQ = G$ resp. $\widehat{Q}Y = H$ have the unique solutions

$$Q = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix} \quad \text{resp.} \quad \widehat{Q} = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} = Q^{-1}.$$

This is the unique Q for which $XQ = G$ and $Q^{-1}Y = H$. This Q is nonsingular, but not in $\mathcal{O}_2$.

This now motivates the following result. In the following, for a given matrix A, let $R(A)$ resp. $S(A)$ denote the subspaces spanned by the rows resp. columns of A and let A^+ denote the Moore-Penrose-inverse of A.

Lemma 3.6. *Let $X, G \in \mathbb{R}^{n\times k}$ and $Y, H \in \mathbb{R}^{k\times m}$ all be of rank k. Then we have $XY = GH$ if and only if there exists a nonsingular matrix $Q \in \mathbb{R}^{k\times k}$ such that $XQ = G$ and $Q^{-1}Y = H$.*

Proof. The if part is obvious. For the reverse part, observe that X and G are of the same rank. It is easy to see that there exists a linear map $f : R(X) \to R(G),\ x^T \mapsto x^T A_f$ such that $f(X_{i*}) = X_{i*}A_f = G_{i*}$ for all $i \in \{1, \dots, n\}$. Similarly, there exists a linear map $g : S(Y) \to S(H),\ y \mapsto A_g y$ such that $g(Y_{*j}) = A_g Y_{*j} = H_{*j}$ for all $j \in \{1, \dots, m\}$. Due to the equality $XY = GH$, we have

$$X_{i*}Y_{*j} = G_{i*}H_{*j} \text{ for every } i \in \{1, \dots, n\} \text{ and } j \in \{1, \dots, m\}. \tag{3.7}$$

Furthermore, since $\operatorname{rank}(X) = \operatorname{rank}(G) = k$, the matrix A_f is nonsingular and f is bijective, so we have that for every i, j

$$X_{i*}Y_{*j} = f^{-1}(G_{i*})Y_{*j} = G_{i*}A_f^{-1}Y_{*j} \text{ and } G_{i*}H_{*j} = G_{i*}A_gY_{*j}.$$

With the help of equation (3.7), we therefore get

$$G_{i*}A_f^{-1}Y_{*j} = G_{i*}A_gY_{*j}, \tag{3.8}$$

for every $i = 1, \dots, n$ and $j = 1, \dots, m$. Moreover, Y is of full row-rank such that $YY^+ = I_{m\times m}$ and G is of full column-rank such that $G^+G = I_{n\times n}$.

Thus, equation (3.8) can be rewritten as

$$GA_f^{-1}Y = GA_gY \quad \Leftrightarrow \quad G^+GA_f^{-1}YY^+ = G^+GA_gYY^+ \quad \Leftrightarrow \quad A_f^{-1} = A_g,$$

which completes the proof. □

This lemma therefore shows that instead of using orthogonal matrices to transform one factorization of type $A = XY$ into another one, we will apply the generalized result of Lemma 3.6, and we will therefore work with nonsingular matrices. For possible regularizations, we use the following approach, for which the proof follows by construction.

Lemma 3.7. *Consider $A \in \mathbb{R}^{n\times n}$ with $\operatorname{rank}(A) = k < n$ and its singular value decomposition $A = U\Sigma V^T$. Furthermore, let $j \le k$ and define a diagonal matrix $\tilde{\Sigma} \in \mathbb{R}^{n\times n}$ via*

$$\tilde{\Sigma}_{ii} := \begin{cases} \Sigma_{ii}, & \text{if } i \le k \\ \Sigma_{jj}, & \text{if } i > k \end{cases},$$

and therewith $\tilde{A} := U\tilde{\Sigma}V^T$. Then $\tilde{A}$ is nonsingular and A is a best rank-k-approximation of $\tilde{A}$.

To again obtain a feasibility problem, we introduce the sets

$$\begin{aligned}\mathcal{P}_{X_k} &:= \{Q \in \mathbb{R}^{k\times k} \mid X_k Q \geq 0\} \text{ and}\\ \mathcal{P}_{Y_k} &:= \{Q \in \mathbb{R}^{k\times k} \mid Q \text{ is nonsingular and } Q^{-1}Y_k \geq 0\}'\end{aligned} \tag{3.9}$$

and note the following result.

Theorem 3.8. *Let $A \in \mathbb{R}^{n\times m}$ and $k \leq \min\{m, n\}$. Consider A_k as in Theorem 3.1 and its initial factorization $A_k = X_k Y_k$ as in (3.2) or (3.3). Then, to obtain a NMF of A of rank k, it is sufficient to find a matrix $Q \in \mathcal{P}_{X_k} \cap \mathcal{P}_{Y_k}$.*

Proof. If $Q \in \mathcal{P}_{X_k} \cap \mathcal{P}_{Y_k}$, then Q is nonsingular by definition of $\mathcal{P}_{Y_k}$. Moreover, we have

$$A_k = X_k Y_k = \underbrace{(X_k Q)}_{\geq 0}\underbrace{(Q^{-1}Y_k)}_{\geq 0},$$

as a NMF of A due to Lemma 3.2, completing the proof. □

We can therefore reduce the problem of finding an NMF to solving the problem

$$\text{find} \quad Q \in \mathcal{P}_{X_k} \cap \mathcal{P}_{Y_k}. \tag{3.10}$$

In summary, to compute a SymNMF resp. a NMF, it is sufficient to solve Problem (3.6) resp. (3.10).

Remark 3.9. The methods introduced in [16] can be used to to derive completely positive factorizations of A_k and therefore to obtain a symmetric nonnegative matrix factorization of A, as Lemma 3.3 substantiates. There the method of alternating projections, cf. [9] is used in order to solve Problem (3.6). Here the idea is to project a starting point orthogonally onto the first set and the resulting element onto the second set and to repeat this procedure in order to obtain an intersection point in the limit.

4. Solving the feasibility problem (3.10). A first idea would be to use a similar alternating projection approach as in [16]. Here the question arises how to project onto the sets $\mathcal{P}_{X_k}$ and $\mathcal{P}_{Y_k}$. As mentioned in [16], the projection onto $\mathcal{P}_{X_k}$ is unique and amounts solving a certain SOCP. However, since $\mathcal{P}_{Y_k}$ is not closed, we can not project onto this set. Nevertheless, in the following, we will show that we can develop a modified version motivated by the heuristic extension in [16].

Let X_k and Y_k be as defined in (3.2). We start with some initial nonsingular matrix Q_0 and project $X_k Q_0$ onto the nonnegative orthant $\mathbb{R}_+^{n\times k}$. We obtain

$$G_{ij} := \max\{(X_k Q_0)_{ij}, 0\} \quad \text{for all } i = 1, \dots, n \text{ and } j = 1, \dots, k. \tag{4.11}$$

Moreover, we also project $Q_0 Y_k$ onto the nonnegative orthant $\mathbb{R}_+^{k\times m}$ and obtain

$$H_{ij} := \max\{(Q_0 Y_k)_{ij}, 0\} \quad \text{for all } i = 1, \dots, k \text{ and } j = 1, \dots, m. \tag{4.12}$$

In order to approximate the projection $\Pi_{P_{X_k}}(Q) \in \mathbb{R}^{k\times k}$ and to obtain an element in $P_{Y_k} \subseteq \mathbb{R}^{k\times k}$, respectively, we modify $G \in \mathbb{R}^{n\times k}$, respectively, $H \in \mathbb{R}^{k\times m}$. This idea now leads to the following tool.

Lemma 4.1. *Let X_k and Y_k be as in (3.2). Furthermore, let G, H be as defined in (4.11) and (4.12). Then $X_k^+ G = \operatorname{Argmin}_{Z\in\mathbb{R}^{k\times k}}\{\|X_k Z - G\|_F\}$ and $H{Y_k}^+ = \operatorname{Argmin}_{Z\in\mathbb{R}^{k\times k}}\{\|Z Y_k - H\|_F\}$.*

Proof. First, we focus on the equation $X_kZ = G$ and assume that there exists a solution Z. It is well known, cf. [19, Theorem 2] and [29], that the complete set of solutions of this equation is given as

$$\left\{Z = {X_k}^+G + (I - X_k^+X_k)T \;\middle|\; T \in \mathbb{R}^{k\times k}\right\} = \left\{X_k^+G\right\}, \tag{4.13}$$

where the equation holds since X_k is of full column rank such that $X_k^+X_k = I$. Thus, $\|X_kZ - G\|_F = 0$ if and only if $Z = X_k^+G$. On the other hand, for the case where there does not exist a solution Z to $X_kZ = G$, again the residual $\|X_kZ - G\|_F$ is minimal if and only if $Z = X_k^+G$. An analogous argument proves the result for the equation $ZY_k = H$. □

With this we can now give an approximation to Q in $\mathcal{P}_{X_k}$ respectively $\mathcal{P}_{Y_k}$, which can be easily computed.

Lemma 4.2.

(a) *Let G be the projection of X_kQ onto $\mathbb{R}^{n\times k}_+$. If $G = X_kQ$, then $Q \in \mathcal{P}_{X_k}$. If $G \neq X_kQ$ and the equation $X_kZ = G$ is solvable for Z, then $X_k^+G \in \mathcal{P}_{X_k}$.*
(b) *Let H be the projection of QY_k onto $\mathbb{R}^{k\times m}_+$. If $QY_k = H$ and Q is nonsingular, then $Q^{-1} \in \mathcal{P}_{Y_k}$. On the other hand, let $H \neq QY_k$ and assume that the matrix HY_k^+ is nonsingular. Further assume that the equation $ZY_k = H$ is solvable for Z. Then we have $(HY_k^+)^{-1} \in \mathcal{P}_{Y_k}$.*

Proof.

(a) If $G = X_kQ$, then $X_kQ \geq 0$, i.e., $Q \in \mathcal{P}_{X_k}$. Otherwise, let $G \neq X_kQ$ and assume that the equation $X_kZ = G$ is solvable for Z. Then Lemma 4.1 yields $Z = X_k^+G$. Therefore, X_k^+G is the projection of Q onto the set $\{Z \in \mathbb{R}^{k\times k} \mid X_kZ = G\}$. Since $G \geq 0$, we get $X_k^+G \in \mathcal{P}_{X_k}$.
(b) If $QY_k = H$ and Q is nonsingular, then $QY_k \geq 0$ and $Q^{-1} \in \mathcal{P}_{Y_k}$ by definition. Otherwise, let $H \neq QY_k$ and assume that $ZY_k = H$ has a solution Z. Hence, $Z = HY_k^+$ due to Lemma 4.1. Thus, HY_k^+ is the projection of Q onto the set $\{Z \in \mathbb{R}^{k\times k} \mid ZY_k = H\}$. Since HY_k^+ is nonsingular by assumption and $H \geq 0$ by definition, we get $(HY_k^+)^{-1} \in \mathcal{P}_{Y_k}$. □

In addition, if the equation $X_kZ = G$ does not have a solution, then $Z := X_k^+G$ minimizes the residual $\|X_kZ - G\|_F$. In this case, we get $X_kZ = X_kX_k^+G$. Here $X_kX_k^+ \neq I$ in general since X_k is not of full row-rank. Thus, it may happen that $X_k^+G \notin \mathcal{P}_{X_k}$; however, this does not seem to impair the good numerical performance.

If on the other hand the equation $ZY_K = H$ does not have a solution, we get with Lemma 4.1 that $Z := HY_k^+$ minimizes the residual $\|ZY_K - H\|_F$. Thus, $ZY_k = HY_k^+Y_k$. In this equation, we have $Y_k^+Y_k \neq I$ in general since Y_k is not of full column rank. Hence, even if HY_k^+ is nonsingular, it may happen that $(HY_k^+)^{-1} \notin \mathcal{P}_{Y_k}$. However, this does not seem to impair the good numerical performance either.

If we combine the results in Lemmas 4.1 and 4.2, we derive matrices in $\mathcal{P}_{X_k}$ or $\mathcal{P}_{Y_k}$ without solving an SOCP. Moreover, in Lemma 4.2, we assumed that HY_k^+ is nonsingular. This is equivalent to a certain rank assumption for H.

Lemma 4.3.

(a) *In our setting, the matrix $HY_k^+ \in \mathbb{R}^{k\times k}$ is nonsingular if and only if $H \in \mathbb{R}^{k\times m}$ is of rank k.*
(b) *In addition, the matrix $X_k^+G \in \mathbb{R}^{k\times k}$ is nonsingular if and only if $G \in \mathbb{R}^{n\times k}$ is of rank k.*

Proof. We only prove (a), part (b) follows analogously: First observe that by Sylvester's inequality, cf. [3, Corollary 2.5.10], we have

$$\operatorname{rank}(H) + \operatorname{rank}(Y_k^+) - k \leq \operatorname{rank}(HY_k^+) \leq \min\{\operatorname{rank}(H), \operatorname{rank}(Y_k^+)\}.$$

Since $\operatorname{rank}(Y_k^+) = \operatorname{rank}(Y_k) = k$, this yields

$$\operatorname{rank}(H) \leq \operatorname{rank}(HY_k^+) \leq \min\{\operatorname{rank}(H), k\}. \tag{4.14}$$

Now observe that HY_k^+ is nonsingular if and only if $\operatorname{rank}(HY_k^+) = k$. Due to (4.14), this is true if and only if $\operatorname{rank}(H) = k$. □

From now on, we will take $X_k^+ G$, respectively, $(HY_k^+)^{-1}$ as approximations of $P_{\mathcal{P}_{X_k}}(Q)$, respectively, $P_{\mathcal{P}_{Y_k}}(Q)$. This reasoning leads to Algorithm 1.

Algorithm 1 Nonnegative Matrix Factorization

Input: $A \in \mathbb{R}^{n \times m}$ with its singular value decomposition $A = U\Sigma V^T$; $k \leq \min\{m, n\}$; initial nonsingular matrix $Q \in \mathbb{R}^{k \times k}$

```
 1: $A_k \leftarrow \sum_{j=1}^{k} \sigma_j U_{*j} V_{*j}^T$
 2: Compute $X_k$ and $Y_k$ as in (3.2) or as in (3.3)
 3: while $X_k Q \ngeq 0$ or $Q^{-1} Y_k \ngeq 0$ do
 4:   $G \leftarrow \max\{X_k Q, 0\}$ entrywise
 5:   $Q_G \leftarrow {X_k}^+ G$
 6:   if $\operatorname{rank}(G) < k$ then
 7:     $Q_G \leftarrow \widetilde{Q_G}$ according to Lemma 3.7
 8:   end if
 9:   $H \leftarrow \max\{Q_G^{-1} Y_k, 0\}$ entrywise
10:   $Q_H \leftarrow H{Y_k}^+$
11:   if $\operatorname{rank}(H) < k$ then
12:     $Q_H \leftarrow \widetilde{Q_H}$ according to Lemma 3.7
13:   end if
14:   $Q \leftarrow Q_H^{-1}$
15: end while
```

Output: Nonsingular matrix Q and a nonnegative matrix factorization $(X_k Q)(Q^{-1} Y_k)$ of A.

In step 3, the main while loop of the algorithm starts. As long as the matrices $X_k Q$ and $Q^{-1} Y_k$ have some negative entries, we at first define G as the entrywise maximum introduced in (4.11). In step 5, we then look for a solution Q_G of $\min \|X_k Q_G - G\|_F$ with the help of Lemma 4.2. Moreover, if the equation $X_k Q = G$ is solvable for Q, we get that $X_k^+ G \in \mathcal{P}_{X_k}$. To obtain a matrix in $\mathcal{P}_{X_k} \cap \mathcal{P}_{Y_k}$, and since the set $\mathcal{P}_{Y_k}$ is based on the inverse of the considered matrices, we check in step 6 whether $X_k^+ G$ is nonsingular, based on the result in Lemma 4.3 and regularize $X_k^+ G$ if necessary. Using its inverse matrix as the matrix Q_0 in (4.12), we obtain the matrix H in step 9 as the introduced entrywise maximum. In step 10, we then look for a solution Q_H of the problem $\min \|Q_H Y_k - H\|_F$ again with the help of Lemma 4.2 and check whether Q_H is nonsingular using Lemma 4.3. According to Lemma 4.2, we know that if Q_H is nonsingular and the equation $QY_k = H$ is solvable, we have $Q_H^{-1} \in \mathcal{P}_{Y_k}$. If necessary, we regularize the matrix Q_H and take its inverse as our next iterate in step 14.

Remark 4.4. Consider A_k as in Theorem 3.1. A necessary assumption for Algorithm 1 to terminate successfully is that there exists an exact factorization $A_k = X_k Y_k$ with $X_k \in \mathbb{R}_+^{n\times k}$ and $Y_k \in \mathbb{R}_+^{k\times m}$. Clearly, this can only be true if $A_k \in \mathbb{R}_+^{n\times m}$. To the best of my knowledge, no general conditions are known for this to hold, but in the following numerical experiments and especially for the benchmark instances introduced in [31], this assumption holds.

Remark 4.5. The main computational burden for each iteration within the while loop lies in the computation of the Moore–Penrose-inverses. We thus obtain $\mathcal{O}(k^3)$ for the worst-case complexity of one iteration.

5. Numerical results for nonnegative matrix factorization.

The following numerical results were carried out on a computer with 88 Intel Xenon ES-2699 cores (2.2 Ghz each) and a total of 0.792 TB Ram. The algorithms were implemented in MatlabR2017a. The algorithm terminates successfully if it returns matrices (X_kQ) and $(Q^{-1}Y_k)$ which are entrywise greater than or equal to -10^{-12} and it terminates without success if these conditions are not fulfilled and the maximum number of iterations is reached.

As a first example, consider the following randomly generated matrix

$$A = \begin{pmatrix} 7 & 3 & 5 & 4 & 13 & 1 & 1 & 4 \\ 6 & 2 & 2 & 4 & 10 & 16 & 8 & 2 \\ 9 & 11 & 1 & 9 & 9 & 5 & 3 & 14 \\ 3 & 10 & 1 & 13 & 19 & 7 & 0 & 13 \\ 21 & 4 & 2 & 2 & 20 & 6 & 4 & 3 \end{pmatrix} \in \mathbb{R}^{5\times 8}, \tag{5.15}$$

of rank 5 and let $k = 3$. Then we compute the best rank-3 approximation A_3 to A:

$$A_3 = \begin{pmatrix} 8.9667 & 4.2210 & 1.5947 & 3.5945 & 11.1316 & 1.6312 & 0.8958 & 4.8216 \\ 6.0643 & 2.0481 & 1.4711 & 3.9834 & 10.0323 & 16.0878 & 7.8358 & 1.9519 \\ 5.5122 & 8.9454 & 1.4337 & 9.6742 & 13.5665 & 4.7817 & 1.0274 & 11.5364 \\ 5.1814 & 11.2683 & 1.5752 & 12.5851 & 15.9546 & 7.0005 & 1.5595 & 14.6929 \\ 20.5074 & 3.6837 & 3.3835 & 2.1058 & 20.3494 & 5.7566 & 4.2303 & 2.8895 \end{pmatrix}.$$

Based on the technique introduced in (3.2), we get the initial factorization $A_3 = X_3Y_3$ with

$$A_3 = \underbrace{\begin{pmatrix} -2.2164 & 0.3624 & 1.0742 \\ -2.4818 & 1.1425 & -3.2988 \\ -3.1257 & -1.7305 & 0.3226 \\ -3.7949 & -2.4724 & -0.0604 \\ -3.8469 & 2.8992 & 1.3067 \end{pmatrix}}_{X_3 \in \mathbb{R}^{5\times 3}} \underbrace{\begin{pmatrix} -3.0116 & -1.9850 & -0.6127 & -2.0773 & -4.6122 & -2.1422 & -0.9359 & -2.3673 \\ 2.4952 & -1.5193 & 0.3004 & -1.8944 & 0.6106 & 0.5318 & 0.8395 & -2.3186 \\ 1.2916 & 0.3463 & 0.1190 & -0.3008 & 0.6402 & -3.0811 & -1.3805 & 0.3863 \end{pmatrix}}_{Y_3 \in \mathbb{R}^{3\times 8}},$$

where $\text{rank}(X_3) = \text{rank}(Y_3) = 3$ by definition. As an initial nonsingular matrix, we take

$$Q_0 = \begin{pmatrix} 0.4397 & 0.0464 & 0.2059 \\ 0.3518 & 0.8796 & 0.0828 \\ 0.2594 & 0.3400 & 0.4412 \end{pmatrix}.$$

Algorithm 1 then takes 153 iterations and 0.0144 seconds to return the factorization

$$A_3 = \underbrace{\begin{pmatrix} 0.7170 & 0.0226 & 0.0047 \\ 0.2196 & 1.3784 & 0.0020 \\ 0.4051 & 0.1073 & 0.0128 \\ 0.3506 & 0.2145 & 0.0165 \\ 1.5761 & 0.4598 & 0.0014 \end{pmatrix}}_{\widetilde{X_3} \in \mathbb{R}_+^{5\times 3}} \underbrace{\begin{pmatrix} 12.2890 & 1.6756 & 1.9019 & 0.1499 & 10.9775 & 0.1221 & 1.0755 & 1.0565 \\ 2.4108 & 0.2940 & 0.6981 & 1.8080 & 4.5600 & 11.2559 & 5.5133 & 0.0000 \\ 21.5553 & 644.1498 & 46.0188 & 736.7893 & 675.0740 & 275.6546 & 0.0000 & 868.9005 \end{pmatrix}}_{\widetilde{Y_3} \in \mathbb{R}_+^{3\times 8}}.$$

Due to Theorem 3.1, we have

$$\|A - A_3\|_F = \|A - \widetilde{X_3}\widetilde{Y_3}\|_F \leq \|A - B\|_F,$$

for every $B \in \mathbb{R}^{n\times m}$ with $\operatorname{rank}(B) \leq 3$. Combined with the fact that $\widetilde{X_3}$ and $\widetilde{Y_3}$ are entrywise nonnegative, we get that $\widetilde{X_3}\widetilde{Y_3}$ is the desired NMF of A of rank 3.

Moreover, in the following experiment, we will have a closer look at the influence of the parameter k for a given matrix $A \in \mathbb{R}^{m\times n}$.

To this end, we consider the randomly generated matrix

$$A = \begin{pmatrix} 16 & 40 & 29 & 9 & 42 & 36 & 24 & 26 \\ 19 & 41 & 30 & 11 & 26 & 31 & 22 & 30 \\ 24 & 34 & 50 & 36 & 25 & 42 & 41 & 48 \\ 13 & 24 & 26 & 25 & 16 & 34 & 28 & 35 \\ 9 & 39 & 29 & 18 & 19 & 39 & 19 & 38 \end{pmatrix} \in \mathbb{R}^{5\times 8}.$$

For the experiment, we test the performance of 100 starting points Q_0 with a maximum of 3000 iterations per starting point in Algorithm 1 for different values of k. The results are collected in Figure 1.

Figure 1 indicates that the performance of Algorithm 1 depends on the choice of k. Whereas for $k = 2$ nearly every starting point returns a NMF, the success rate decreases for higher values of k. Taking $k = 4$ still returns a success rate of more than 90%. For $k = 5 = \operatorname{rank}(A)$, it is still possible to derive a NMF of A. This therefore shows that it is not necessary to add a low-rank constraint to obtain a NMF. Algorithm 1 can thus be applied to compute exact nonnegative matrix factorizations. Figure 1 therefore substantiates the good performance of Algorithm 1 for different choices of k for the concrete test instance.

The next experiment will analyse the influence of the order $n \times m$ of the given matrix A on the performance of Algorithm 1. To this end, we consider randomly generated matrices $A \in \mathbb{R}^{n\times m}$ for different

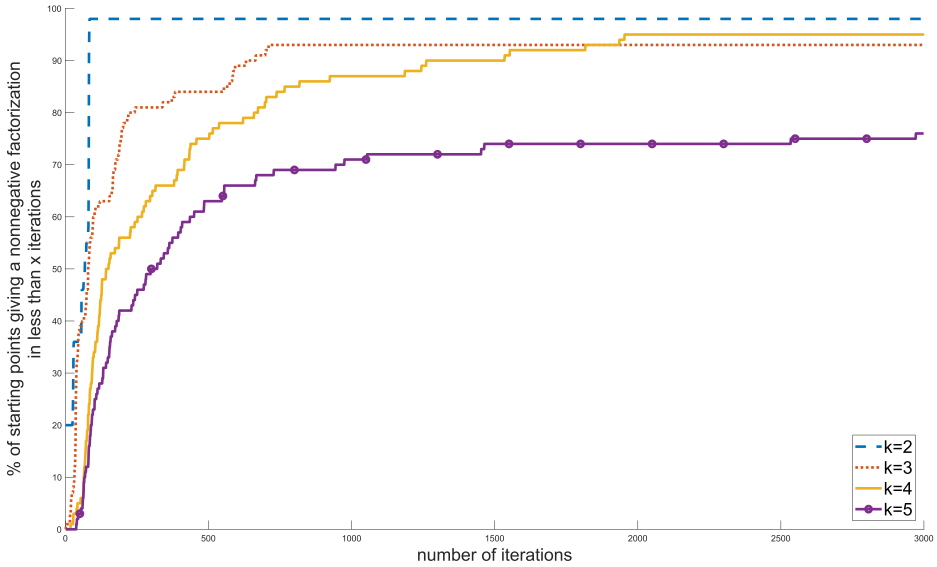

FIGURE 1. *Success rate of Algorithm 1 for a given matrix of order* 5×8 *and different values of* k.

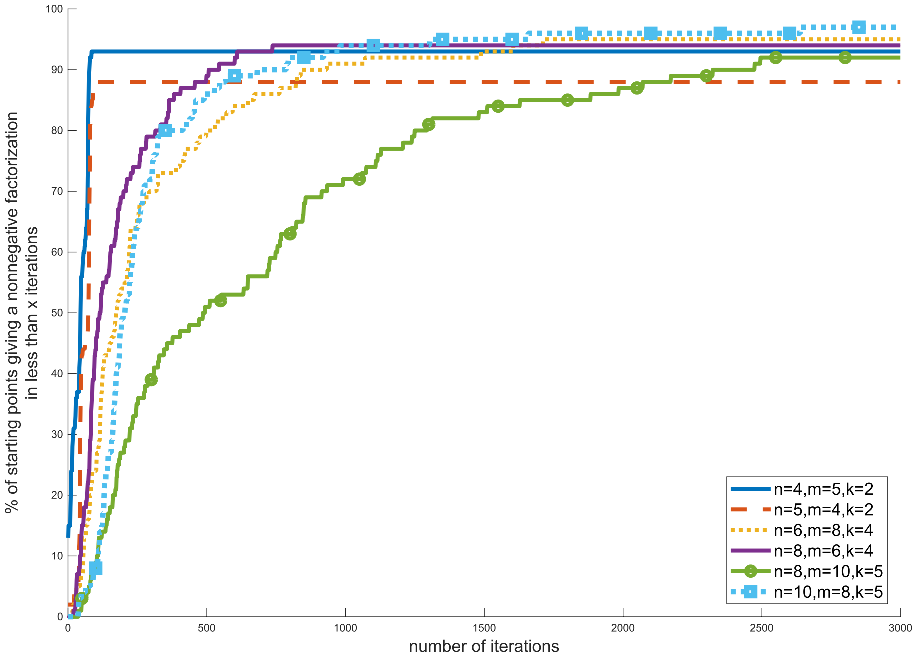

FIGURE 2. *Success rate of Algorithm 1 for matrices of order $n \times m$ and $k = \lfloor 0.7 \cdot \min\{n, m\} \rfloor$.*

values of n and m. The instances are generated as follows: Given the values n, m, we define $l := \min\{n, m\}$ and with this, we construct matrices $X \in \mathbb{R}^{n\times l}$ and $Y \in \mathbb{R}^{l\times m}$ using the Matlab command `randn`. Then we construct $|X|$ and $|Y|$ by taking the absolute values of the entries. Finally, we define $A := |X| \cdot |Y|$.

For each such generated matrix, we set $k = \lfloor 0.7l \rfloor$ and analyse the success rate of 100 randomly generated initial nonsingular matrices Q_0. For every Q_0, we allow at most 3000 iterations. The performance of Algorithm 1 in this setting is illustrated in Figure 2.

Note that Algorithm 1 terminates successfully in every case, such that the success in total is independent of n and m and it does not make a difference whether $m > n$ or $m < n$. Especially for small dimensions like $n = 4,\ m = 5$ and $n = 5,\ m = 4$, plotted as the solid blue and the dashed red line, it turns out that the algorithm terminates successfully for around 90% of the initial nonsingular matrices. Furthermore, if the algorithm terminates successfully for one of the initial nonsingular matrices, a NMF is provided in less than 100 iterations. Increasing values of m and n do not seem to influence the success rate in less then 3000 iterations, but it takes more iterations on average to return a NMF. Here especially the green line, representing $n = 8,\ m = 10$, illustrates this behaviour. Altogether, Figure 2 shows that the introduced method to derive NMF works well for matrices A, which are randomly generated. In the following, we will further analyze the performance of Algorithm 1 as a method to compute exact NMF in comparison to the heuristics introduced in [31]. For this experiment, we consider the benchmark instances summarized in [31, Table 1], containing so-called linear Euclidean distance matrices (LEDM6, LEDM8, LEDM12, LEDM16, LEDM32), slack matrices of well-known polytopes (6-G, 7-G, 8-G, 9-G, 12-G, 32-G, 20-D, 24-C), unique disjointness matrices (UDISJ4, UDISJ5, UDISJ6), and randomly generated instances (RND1, RND3). For more details, the reader is referred to [31]. We say that a method terminates successfully for given matrix $A \in \mathbb{R}^{m\times n}$ and $k \leq \min\{m, n\}$ if it return matrices $X \in \mathbb{R}^{m\times k}$ and $Y \in \mathbb{R}^{k\times m}$ which are entrywise greater

Table 1

Representative computation time in seconds for the individual algorithms to terminate successfully applied to the benchmark instances introduced in [31].

	Algorithm 1	MS2	SA	RBR	Hybrid
LEDM6	32.99	8.44	1.45	11.49	12.31
LEDM8	113.11	–	–	–	–
LEDM12	14.65	–	1.22	21.58	13.96
LEDM16	17,28	–	–	–	–
LEDM32	16,51	–	–	–	–
6-G	–	1.76	1.38	0.56	–
7-G	98.67	1.82	1.53	15.23	16.48
8-G	132.96	1.82	1.42	15.79	16.80
9-G	248.26	–	1.53	16.18	16.85
12-G	40.59	–	1.71	53.97	17.84
32-G	134.07	–	–	–	65.24
20-D	148.89	–	1.77	–	–
24-C	–	–	2.09	–	–
UDISJ4	0.07	1.91	1.72	–	0.78
UDISJ5	0.07	2.22	2.84	–	–
UDISJ6	–	–	–	0.07	–
RND1	0.89	2.12	2.03	–	1.02
RND3	0.02	2.22	2.02	–	0.94

or equal to -10^8 and $\|A - XY\|_F \leq 10^{-8}$ holds. Table 1 shows the computation time on a standard Laptop for each algorithm to terminate successfully on factorizing the benchmark instances taken from [31]. The ”–” entries indicate that the algorithm was not sucessfull. For the methods introduced in [31], the provided code as well as the recommended default parameters are used for fair comparisons. As can be seen, Algorithm 1 returns a nonnegative matrix factorization in nearly all of the cases, whereas for the other heuristics, only the method SA seems to be as reliable as the method presented in this paper. For some instances corresponding to slack matrices of well-known polytopes, Algorithm 1 takes more time as compared to the heuristics introduced in [31]. But especially for the unique disjointness matrices and the random instances, the method presented in this paper seems to be the fastest.

6. Conclusion.

In this paper, we introduced a new method to derive NMF for general rectangular input matrices, based on the alternating projection method, extending the results for the symmetric case as introduced in [16]. Unfortunately, since one of the subsets we need to project on is not closed, we could not apply straight forward alternating projections. Instead, we generalized the alternative approach to obtain an applicable heursitic approach for (exact) NMF. As numerical experiments substantiate, this algorithm still works well and hence gives a new approach to derive nonnegative matrix factorizations. Furthermore, as numerical experiments on benchmark instances show, the exact NMF method presented in this paper is at least as reliable as the heuristics introduced in [31] and even faster for some instances.

Acknowledgments. I highly appreciate the editor's and the referees' efforts, and I am especially grateful for their valuable comments and remarks. Moreover, I wish to thank Mirjam Dür for her highly useful comments and recommendations.

REFERENCES

[1] A. Berman and N. Shaked-Monderer. *Completely Positive Matrices.* World Scientific Publishing, 2003.

[2] A. Berman, M. Dür, and N. Shaked-Monderer. Open problems in the theory of completely positive and copositive matrices. *Electron. J. Linear Algebra,* 29:46–58, 2015.

[3] D.S. Bernstein. *Matrix Mathematics: Theory, Facts, and Formulas.* Princeton University Press, 2009.

[4] M.W. Berry, M. Browne, A.N. Langville, V.P. Pauca, and R.J. Plemmons. Algorithms and Applications for Approximate Nonnegative Matrix Factorization *Computat. Stat. Data Anal.,* 52:155–173, 2007.

[5] R. Borhani, J. Watt, and A. Katsaggelos. Fast and Effective Algorithms for Symmetric Nonnegative Matrix Factorization. *arXiv preprint arXiv:1609.05342*, 2016.

[6] J.-P. Brunet, P. Tamayo, T.R. Golub, and J.P. Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proc. Nat. Acad. Sci. U. S. A.* 101:4164–4169, 2004.

[7] E. Canhasi and I. Kononenko. Automatic extractive multi-document summarization based on archetypal analysis. In: G.R. Naik (editor), *Non-negative Matrix Factorization Techniques: Advances in Theory and Applications,* 75–88, 2016.

[8] G. Casalino, N. Del Buono, and C. Mencar. Nonnegative matrix factorizations for intelligent data analysis. In: G.R. Naik (editor), *Non-negative Matrix Factorization Techniques: Advances in Theory and Applications,* 49–74, 2016

[9] F. Deutsch. The method of alternating orthogonal projections. In: S.P. Singh (editor), *Approximation Theory, Spline Functions and Applications,* 356:105–121, 1992.

[10] C. Ding, X. He, and H.D. Simon. On the equivalence of nonnegative matrix factorization and spectral clustering. In: *Proceedings of the 2005 SIAM International Conference on Data Mining*, 606–610, 2005.

[11] K. Drakakis, S. Rickard, R. de Frein, and A. Cichocki. Analysis of financial data using non-negative matrix factorization. *Int. Math. Forum,* 3:1853–1870, 2008.

[12] C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika* 3:211–218, 1936.

[13] F. Li M.K. Ng, and R.J. Plemmons. Coupled segmentation and denoising/deblurring models for hyperspectral material identification. *Numer. Linear Algebra Appl.* 19, 153–173 , 2012.

[14] N. Gillis. Introduction to nonnegative matrix factorization. *SIAG/OPT Views and News* 1, 2017.

[15] N. Gillis. Nonnegative Matrix Factorization. *Society for Industrial and Applied Mathematics*, 2020.

[16] P. Groetzner and M. Dür. A factorization method for completely positive matrices. *Linear Algebra Appl.,* 591:1–24, 2020.

[17] Z. He, S. Xie, R. Zdunek, G. Zhou and A. Cichocki. Symmetric nonnegative matrix factorization: Algorithms and applications to probabilistic clustering. *IEEE Trans. Neural Netw.* 22:2117–2131, 2011.

[18] N.-D. Ho. Nonnegative matrix factorization algorithms and applications. *Phd Thesis, Université catholique de Louvain*, 2008.

[19] M. James. The generalised inverse. *Math. Gaz.,* 420:109–114, 1978.

[20] D. Kuang, C. Ding, and H. Park. Symmetric nonnegative matrix factorization for graph clustering. In: *Proceedings of the 2012 SIAM International Conference on Data Mining,* 106–117, 2012.

[21] T. Laudadio, A. C. Sava, Y. Li, N. Sauwen, D. Sima, and S. Van Huffel. NMF in MR spectroscopy. In: G.R. Naik (editor), *Non-negative Matrix Factorization Techniques: Advances in Theory and Applications,* 161–177, 2016.

[22] C. Lazar and A. Doncescu. Non negative matrix factorization clustering capabilities; application on multivariate image segmentation. In: L. Barolli, F. Xhafa, and H.-H. Hsu (editors), *International Conference on Complex, Intelligent and Software Intensive Systems 2009,* 924–929, 2009.

[23] D. Lee, H. Seung. Learning the parts of objects by nonnegative matrix factorization. *Nature,* 401:788791, 1999.

[24] D. Lee and H. Seung. Algorithms for non-negative matrix factorization. In: T. K. Leen, T. G. Dietterich, and V. Tresp (editors), *Advances in Neural Information Processing Systems,* 13:556–562, 2001.

[25] Y. Ma, X. Hu, T. He, and X. Jiang. A Robust Symmetric Nonnegative Matrix Factorization Framework for Clustering Multiple Heterogeneous Microbiome Data, 2017. *Preprint,* *https://www.preprints.org/manuscript/201704.0105/v1*

[26] S.K. Mohamad and Z. Tasir. Educational data mining: A review. *Proc.-Social Behav. Sci.* 97:320–324, 2013.

[27] L. Mirsky. Symmetric gauge functions and unitarily invariant norms. *Quart. J. Math. Oxford. Second Ser.,* 11:50–59, 1960.

[28] P. Paatero and U. Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics,* 5:111–126, 1994.

[29] M. Planitz. Inconsistent systems of linear equations. *Math. Gaz.,* 425:181–185, 1979.

[30] U. Schlink and A. Thiem. Non-negative matrix factorization for the identification of patterns of atmospheric pressure and geopotential for the northern hemisphere. *Int. J. Climatol.,* 30:909–925, 2010.

[31] A. Vandaele, N. Gillis, F. Glineur and D. Tuyttens. Heuristics for exact nonnegative matrix factorization. *J. Global Optim.,* 65(2):369–400, 2016.

[32] S. Vavasis. On the complexity of nonnegative matrix factorization. *SIAM J. Optim.,* 20(3):1364–1377, 2010.

[33] Z.-Y. Zhang. Nonnegative matrix factorization: Models, algorithms and applications. In: D.E. Holmes and L.C. Jain (editors), *Data Mining: Foundations and Intelligent Paradigms: Volume 2: Statistical, Bayesian, Time Series and other Theoretical Aspects,* 99–134, 2012.