Fifty-fourth Annual Allerton Conference
Allerton House, UIUC, Illinois, USA
September 27 - 30, 2016

Nonnegative Matrix Factorization by Optimization on the Stiefel Manifold with SVD Initialization

Ali Koochakzadeh* Sina Miran* Pouya Samangouei* Michael C. Rotkowitz

***Abstract*— We consider the problem of Nonnegative Matrix Factorization (NMF) which is a non-convex optimization problem with many applications in machine learning, computer vision, and topic modeling. General existing methods for finding a solution to the problem include additive or multiplicative update rules for doing alternate minimizations and ADMM, which find a locally optimal point for NMF. We propose a new method for finding a solution of NMF which considers transformations of initial factors derived from the singular value decomposition (SVD). This problem is shown to be equivalent to the NMF problem in a sense, and is then restricted to optimize over the set of orthonormal matrices known as the Stiefel manifold. We then utilize a method developed for optimization over this manifold to find solutions to the NMF problem. Application to synthetic data shows that the method exhibits promising characteristics, as it outperforms some traditional methods both in terms of reconstruction error and running time. Using the solution of this restriction as a starting point for the broader problem shows a further rapid decline in the error.**

I. INTRODUCTION

Nonnegative Matrix Factorization (NMF) is a non-convex optimization problem which arises in many areas such as topic modeling, recommender systems, and clustering [1]–[3]. The reason for the popularity of NMF is the interpretability of its results. Problems (1) and (2) express the exact and the approximate versions of the problem. The exact version in (1) has been written in the form of a feasibility problem. The objective function in the approximate version is most often written in Frobenius norm, as in (2); however, it can be replaced by any desired distance measure for matrices. The least square formulation resembles dealing with noisy observations as Y is mostly noisy, and we are looking for a close approximation of it that can be factored into elementwise non-negative components. The observation matrix Y is in $\mathbb{R}^{m\times n}$, and vec(.) represents the vectorizing operator for matrices. Thus, the condition $\text{vec}(.) \geq 0$ implies the elementwise nonnegativity of the argument matrix. The desired approximation rank r is mostly well below m and n, i.e. $r \ll \min\{m, n\}$.

$$\begin{aligned} \underset{A\in\mathbb{R}^{m\times r},\, B\in\mathbb{R}^{r\times n}}{\text{find}} \quad & (A, B) \\ \text{subject to} \quad & Y = AB,\ \text{vec}(A) \succeq 0,\ \text{vec}(B) \succeq 0 \end{aligned} \tag{1}$$

$$\begin{aligned} \underset{A\in\mathbb{R}^{m\times r},\, B\in\mathbb{R}^{r\times n}}{\text{minimize}} \quad & \|Y - AB\|_F^2 \\ \text{subject to} \quad & \text{vec}(A) \succeq 0,\ \text{vec}(B) \succeq 0 \end{aligned} \tag{2}$$

The power of NMF comes from the interpretability of its solutions A and B. For instance, in the context of topic modeling, a topic is defined as a (sparse) distribution on words, and we are trying to extract $r \ll \min\{m, n\}$ topics from a corpus while labeling each document with a few of the topics. In this case, the observation matrix $Y_{m\times n}$ is the normalized count of the m words in each of the n documents in the corpus. Consequently, the solutions $A_{m\times r}$ and $B_{r\times n}$ can be interpreted respectively as the word by topic matrix, expressing the distribution of each topic over words, and the topic by document matrix, expressing the distribution of each document over the topics [4]. Similarly, in the movie rating application, the observation matrix is a user by movie matrix consisting of ratings given by each user to each movie. Therefore, the exact same decomposition can be done to extract a total of r genres together with the distribution of each person's interest on the genres (solution $A^*_{m\times r}$) and the distribution of each movie on different genres (solution $B^*_{r\times n}$). It is worth noting that without the elementwise nonnegativity constraints, the approximation could be done by using matrix decompositions such as SVD [4].

More generally, NMF can be applied to reveal latent structures in the data as the elementwise nonnegativity of the solutions A and B allows them to be interpreted as distributions of the latent variables over observations. Both formulations of the problem are nonconvex due to the bilinear term, and are generally difficult to solve.

Many of the attempts for finding a solution for NMF rely on the bi-convexity of the problem in terms of A and B; i.e., the problem becomes convex if either A or B is fixed [3] [5]. Consequently, gradient descent based update rules have been used to do alternate minimizations over A and B until a convergence condition is satisfied. Although these approaches have been shown to work well in certain scenarios, they do not have any provable guarantees on finding a solution close to the global solution and are likely to end up in local optima based on the initialization points $A^{(0)}$ and $B^{(0)}$. Thus, SVD-based initializations have also been

* denotes equal contribution

A. Koochakzadeh is with the Department of Electrical and Computer Engineering, University of California, San Diego, CA 92093 USA, alik@eng.ucsd.edu

S. Miran is with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742 USA, smiran@umd.edu

P. Samangouei is with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742 USA, pouya@umd.edu

M. C. Rotkowitz is with the Institute for Systems Research and the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742 USA, mcrotk@umd.edu

978-1-5090-4550-1/16/$31.00 ©2016 IEEE

proposed to improve the performance of NMF algorithms [6].

Recently, there have been successful attempts on designing algorithms with provable guarantees under certain conditions. For instance, in the context of topic modeling, a word can have a non-zero probability in the distribution of a single topic and almost zero probabilities in the distribution of all the other topics within a corpus, i.e. the words (terms) "S&P 500" or "401k" which can be thought of as exclusively belonging to the finance and investment topic. The words with this property are called anchor words. It has been shown that if an anchor word exists for each topic, then the NMF problem in topic modeling can be solved by a very efficient algorithm with provable guarantees [7]. Therefore, some of the recent work on NMF has focused on reducing the complexity of the problem by making domain-specific assumptions on the structure of the solutions (A^*,B^*) and solving the problem with provable guarantees afterwards.

In this paper, we have introduced a new algorithm for finding a solution to the NMF problem called the Q-method. Starting from the SVD, we transform the search space over the elementwise nonnegative matrices $A \in \mathbb{R}^{m\times r}$ and $B \in \mathbb{R}^{r\times n}$ to a new space parameterized by matrices $Q_1 \in \mathbb{R}^{r\times r}$ and $Q_2 \in \mathbb{R}^{r\times r}$ satisfying certain conditions such as $Q_1Q_2^T = I$. Then, we introduce a new bi-convex optimization problem for finding a feasible solution for Q_1 and Q_2, which can be solved using similar NMF alternating minimization methods. Finally, taking advantage of the rich literature on Stiefel manifold optimization [8], we restrict ourselves to the case of $Q_1 = Q_2 = Q$ which modifies the previous constraint to the orthogonality constraint $QQ^T = I$. Simulations on randomly generated matrices show that Q-method has a better performance comparing to other general NMF methods such as additive and multiplicative update rules for alternate minimization and ADMM.

The rest of the paper is organized as following: Section II, reviews the general existing methods for obtaining a solution for NMF which have been compared to our Q-method. Section III explains the proposed Q-method in detail. Simulation results on random matrices are shown in Section IV. Finally, Section V includes our concluding remarks and discussion on the proposed method.

II. Existing Methods

A. Additive Update Rules for Alternating Minimization (A-AM)

A popular approach to solve (2) is to use alternating minimization on variables A and B. For the additive update rule, we follow the projected gradient method discussed in [9] which is summarized in Algorithm 1. To find $A^{(k+1)}$, $f = \|Y - AB^{(k)}\|_F^2$ is minimized by projecting the gradient descent solution to the non-negative orthant:

$$A^{(k+1)} = P\left[A^{(k)} - \alpha\nabla_A f\right] \tag{3}$$

where $P[.]$ zeros out the negative values, and α is updated according to Armijo step search condition. The same iterations are then repeated by fixing $A = A^{(k)}$ to find $B^{(k)}$. The discussed projected gradient method would be the same as solving the alternate minimizations by Forward-Backward Splitting [10]. As mentioned before, doing alternative minimization is sensitive to initialization of the variables.

Algorithm 1 A-AM

1: Initialize $B^{(0)}$, $k = 0$
2: **while** not Converged **do**
3: $A^{(k+1)} = P\left[A^{(k)} + 2\alpha_k(Y - A^{(k)}B^{(k)})(B^{(k)})^T\right]$, where α_k is chosen through backtracking
4: $B^{(k+1)} = P\left[B^{(k)} + 2\beta_k(A^{(k+1)})^T(Y - A^{(k+1)}B^{(k)})\right]$, where β_k is chosen through backtracking
5: $k = k + 1$
6: **end while**

B. Multiplicative Update Rules for Alternating Minimization (M-AM)

Another popular alternating minimization algorithm for general NMF is given in [5] and [11]. This algorithm uses multiplicative update rules as opposed to additive update rules in most gradient descent algorithms. The update rules are given in Algorithm 2. One advantage of having such update rules is that the updated A and B matrices would stay elementwise positive at each step without performing an extra projection step. Also, there is no backtracking step in Algorithm 2 which improves its running time. If the M-AM method reaches a global minimum, i.e. $Y = A^{(n)}B^{(n)}$ at some step n, then the update rations become unity making this a stationary point for the algorithm.

Algorithm 2 M-AM

1: Initialize $A^{(0)}$, $B^{(0)}$, $k = 0$
2: **while** not Converged **do**
3: $A_{i,j}^{(k+1)} = A_{(i,j)}^{(k)} \frac{\left(Y(B^{(k)})^T\right)_{i,j}}{\left(A^{(k)}B^{(k)}(B^{(k)})^T\right)_{i,j}}, \quad \forall i,j$
4: $B_{i,j}^{(k+1)} = B_{(i,j)}^{(k)} \frac{\left((A^{(k+1)})^TY\right)_{i,j}}{\left((A^{(k+1)})^TA^{(k+1)}B^{(k)}\right)_{i,j}}, \quad \forall i,j$
5: $k = k + 1$
6: **end while**

C. Alternating Direction Method of Multipliers (ADMM)

A more sophisticated alternate minimization scheme is to apply ADMM on (2) making use of the dual variables as well [12]. Although the initial version of ADMM is defined for optimization problems with convex separable objective function and affine constraints, neither of which holds here, we can calculate the ADMM iterations in NMF due to the bi-convexity of the problem [12]. A reformulation of (2) is shown in (4) below for ADMM:

$$\begin{aligned} \underset{A\in\mathbb{R}^{m\times r},\, B\in\mathbb{R}^{r\times n},\, X\in\mathbb{R}^{m\times n}}{\text{minimize}} \quad & \frac{1}{2}\|Y - X\|_F^2 + I_+(A) + I_+(B) \\ \text{subject to} \quad & X - AB = 0 \end{aligned} \tag{4}$$

where $I_+(.)$ is the $0/\infty$ indicator function of elementwise nonnegative matrices. In this reformulation we have a convex objective function in terms of X, A, and B and a bi-affine constraint in terms of A and B. Thus, if we apply ADMM on (4), each iteration will be a convex optimization problem. However, we do not have any of the convergence guarantees of ADMM as our original problem is non-convex. Generally, when ADMM is applied to a non-convex optimization problem as a local optimization method, it will possibly have better convergence properties, in terms of speed and objective value, than other local optimization methods on the primal problem such as A-AM and M-AM. Algorithm 3 summarizes the application of ADMM on (4). For each of the minimizations we have used a projected gradient approach similar to the one in A-AM.

Algorithm 3 ADMM

Initialize $B^{(0)}$, $U^{(0)}$, ρ, $k=0$
while not Converged **do**
  $(X^{(k+1)}, A^{(k+1)}) = \underset{A\geq 0, X}{\operatorname{argmin}} \left(\|X-Y\|_F^2 + \frac{\rho}{2}\|X - AB^{(k)} + U^{(k)}\|_F^2\right)$
  $B^{(k+1)} = \underset{B\geq 0}{\operatorname{argmin}} \left(\|X^{(k+1)} - A^{(k+1)}B + U^{(k)}\|_F^2\right)$
  $U^{(k+1)} = U^{(k)} + X^{(k+1)} - A^{(k+1)}B^{(k+1)}$
  $k = k+1$
end while

III. Proposed Method

Let $U\Sigma V^T$ be the compact SVD of the observation matrix Y, where $U \in \mathbb{R}^{m\times r}$, $\Sigma \in \mathbb{R}^{r\times r}$, and $V \in \mathbb{R}^{n\times r}$. Let $\tilde{U} = U\Sigma^p$, and $\tilde{V} = V\Sigma^{1-p}$ for some $0 \leq p \leq 1$. Clearly, we have $Y = \tilde{U}\tilde{V}^T$, where $\tilde{U}$ and $\tilde{V}$ are full column rank matrices. For the purpose of theoretical analysis suppose that the nonnegative rank of Y is equal to its rank r. Thus, we know that there exists a full column rank elementwise nonnegative matrix $A \in \mathbb{R}^{m\times r}$ and a full row rank elementwise nonnegative matrix $B \in \mathbb{R}^{r\times n}$ such that $Y = AB$. The following theorem helps to transform the NMF search space from elementwise nonnegative matrices $A \in \mathbb{R}^{m\times r}$ and $B \in \mathbb{R}^{r\times n}$ to a new search space parameterized by $Q_1 \in \mathbb{R}^{r\times r}$ and $Q_2 \in \mathbb{R}^{r\times r}$.

Theorem 1: There exist full column rank matrices $A \in \mathbb{R}^{m\times r}$ and $B^T \in \mathbb{R}^{n\times r}$ such that $Y = AB$ if and only if there exist unique $Q_1, Q_2 \in \mathbb{R}^{r\times r}$ such that $Q_1Q_2^T = I$, $A = \tilde{U}Q_1$, and $B^T = \tilde{V}Q_2$.

Proof: Let $\mathcal{R}(X)$ denote the range space of matrix X. First assume that the second condition holds. Clearly if such Q_1 and Q_2 exist then we have $AB = \tilde{U}Q_1Q_2^T\tilde{V}^T = \tilde{U}\tilde{V} = Y$ based on the definitions of $\tilde{U}$ and $\tilde{V}$ from the SVD of Y.

Now we assume that the first condition holds. If $Y = AB$ with A and B^T being full rank matrices, then $\mathcal{R}(Y) = \mathcal{R}(A)$ since B being full row rank results in $\mathcal{R}(B) = \mathbb{R}^r$. Similarly, we can argue that $\mathcal{R}(Y) = \mathcal{R}(\tilde{U})$, and hence, $\mathcal{R}(A) = \mathcal{R}(\tilde{U})$.

Let a_i denote the i^{th} column of A. Thus for each a_i there exists a unique $q_i \in \mathbb{R}^r$ such that $a_i = \tilde{U}q_i$ since $\tilde{U}$ is a full column rank matrix. Defining $Q_1 = [q_1, q_2, \cdots, q_r]$, we deduce that there exists a unique $Q_1 \in \mathbb{R}^{r\times r}$ such that $A = \tilde{U}Q_1$. Similarly, we can argue that there exists a unique $Q_2 \in \mathbb{R}^{r\times r}$ such that $B^T = \tilde{V}Q_2$.

Substituting into the original $Y = AB$ equation we have $Y = \tilde{U}Q_1Q_2^T\tilde{V}^T = \tilde{U}\tilde{V}^T$ and thus $\tilde{U}(Q_1Q_2^T - I)\tilde{V}^T = 0$. Again, as $\tilde{U}$ and $\tilde{V}$ are full column rank matrices, we obtain $Q_1Q_2^T = I$ which concludes the proof. ■

Therefore, we have shown that there is a one to one mapping between every solution pair (A, B) and (Q_1, Q_2) satisfying $Q_1Q_2^T = I$. Based on the mentioned facts and assumptions, the nonnegative matrix factorization problem can be written as

$$(\mathbf{P1}) \quad \begin{aligned} \underset{Q_1,Q_2\in\mathbb{R}^{r\times r}}{\text{find}} \quad & (Q_1, Q_2) \\ \text{subject to} \quad & Q_1Q_2^T = I, \\ & (\tilde{U}Q_1)_{ij} \geq 0, (\tilde{V}Q_2)_{ij} \geq 0, \forall i, j \end{aligned}$$

which is a feasibility search problem, and the set is nonconvex due to the constraint $Q_1Q_2^T = I$. We propose rewriting this problem as the following optimization problem:

$$(\mathbf{P2}) \quad \begin{aligned} \underset{Q_1,Q_2\in\mathbb{R}^{r\times r}}{\text{minimize}} \quad & \|(\tilde{U}Q_1)_-\|_F^2 + \|(\tilde{V}Q_2)_-\|_F^2 \\ \text{subject to} \quad & Q_1Q_2^T = I \end{aligned}$$

where $(X)_- = \min(X, \mathbf{0})$, i.e, it keeps the negative elements of X unchanged, and sets the positive elements to zero. The objective function of $(\mathbf{P2})$ is convex; however, due to the nonconvex constraint $Q_1Q_2^T = I$, it is a nonconvex problem. If the problem $(\mathbf{P2})$ attains zero as its optimal value, it means that we were able to find a nonnegative factorization. Hence, the feasible set of problem $(\mathbf{P1})$ is nonempty, and the primary solutions are $\hat{A} = \tilde{U}\hat{Q}_1$ and $\hat{B}^T = \tilde{V}\hat{Q}_2$, where $\hat{Q}_1, \hat{Q}_2$ are the solutions of $(\mathbf{P2})$. If the optimal value of $(\mathbf{P2})$ is a small positive value rather than zero, which is the case in most high dimensional problems, approximate solutions of A and B are calculated by truncating the negative values to zero, i.e. $\hat{A} = (\tilde{U}\hat{Q}_1)_+$ and $\hat{B}^T = (\tilde{V}\hat{Q}_2)_+$ where $(X)_+ = \max(X, \mathbf{0})$.

In the following, we solve this problem for two different cases. In the first case, we assume that Q_1 and Q_2 are not equal, and use an alternating minimization approach. In the second case, we assume $Q = Q_1 = Q_2$; therefore, $QQ^T = I$ and Q is an orthonormal matrix. Then, we solve $(\mathbf{P2})$ over the set of orthonormal matrices, known as Stiefel manifold. Although the latter seems to be a restrictive assumption, our simulation results show that by making this assumption, we are still able to find valid nonnegative matrix factorizations, and find feasible solutions to $(\mathbf{P1})$.

A. Alternating Minimization (Q-AM Method)

In this section, we introduce a method for solving $(\mathbf{P2})$ via an alternating minimization approach. Our approach consists

of two main steps. In the first step, we keep Q_2 fixed, and solve the optimization problem for Q_1. In the second step, we keep Q_1 fixed, and minimize the objective function with respect to Q_2. We repeat these operations until a convergence criterion is met. Algorithm 4 summarizes this approach.

Algorithm 4 Q-AM

while not Converged **do**
$Q_1^{(k+1)} = \underset{Q_1 \in \mathbb{R}^{r\times r}}{\arg\min} \|(\tilde{U}Q_1)_-\|_F^2 + \tau \|Q_1 (Q_2^{(k)})^T - I\|_F^2$
$Q_2^{(k+1)} = \underset{Q_2 \in \mathbb{R}^{r\times r}}{\arg\min} \|(\tilde{V}Q_2)_-\|_F^2 + \tau \|Q_1^{(k+1)} Q_2^T - I\|_F^2$
end while

$\tau > 0$ is a regularization parameter. The reason why we did not consider constraints like $Q_1 \left(Q_2^{(k)}\right)^T = I$ and instead added a penalty term is that adding the constraint can make the feasible set shrink to a single point $Q_1 = \left(Q_2^{(k)}\right)^{-1}$ (assuming that the inverse exists). This would prevent us to make any progress in each iteration.

We also note that the objective function for the optimization problems in 3 is convex. $|\min(X_{ij}, 0)|^2$ is convex for each i, j; hence, the Frobenius norm is convex. Moreover, the penalty term is Frobenius norm of an affine function which is convex. If this iterative approach finds zero as the optimal value of the optimization problems, we have found a feasible solution for nonnegative matrix factorization.

B. Optimization on Stiefel Manifold (Q-Method)

In this section, we add a more restrictive condition that $Q_1 = Q_2$. We will later elaborate on this assumption. This assumption reduces the problem into the following:

$$\begin{aligned} \underset{Q\in\mathbb{R}^{r\times r}}{\text{minimize}} \quad & \|(\tilde{U}Q)_-\|_F^2 + \|(\tilde{V}Q)_-\|_F^2 \\ \text{subject to} \quad & QQ^T = I \end{aligned}$$

This is a non-convex optimization problem as the feasible set is the highly non-convex set of orthonormal matrices, also known as Stiefel manifold. We denote this manifold with $\mathcal{M}_n$. Stiefel manifold optimization has been popular in the context of orthogonal matrix factorization [13]; however, it has not been used in the general NMF to the best of our knowledge. We follow the algorithm in [8] and propose to solve this problem via a projected gradient descent approach. We perform the iterations as shown in Algorithm 5.

Algorithm 5 Q-method

while not Converged **do**
$G = \partial_Q F(Q^{(k)})$
$H = G(Q^{(k)})^T - Q^{(k)} G^T$
Choose some τ via curvilinear search
$R(\tau) = (I + \frac{\tau}{2} H)^{-1} (I - \frac{\tau}{2} H)$ (Cayley Transform)
$Q^{(k+1)} = R(\tau) Q^{(k)}$
end while

We have $F(Q) = \|(\tilde{U}Q)_-\|_F^2 + \|(\tilde{V}Q)_-\|_F^2$, and it is easy to verify that the gradient with respect to Q is $G = 2\tilde{U}^T(\tilde{U}Q)_- + 2\tilde{V}^T(\tilde{V}Q)_-$. Since H is an anti-symmetric matrix, i.e., $H + H^T = 0$, it can be shown that $R(\tau) \in \mathcal{M}_n$, i.e, $R(\tau)R(\tau)^T = I$, for every $\tau > 0$ [8]. Moreover, $\frac{d}{d\tau} R(\tau) Q^{(k)} \Big|_{\tau=0}$ equals the projection of $-G$ onto the tangent space of $\mathcal{M}_n$. Therefore, $R(\tau)Q^{(k)}$ is a descent path.

Here τ is a descent parameter that can either be a fixed number, or be chosen through a curvilinear search in each iteration. In practice, a curvilinear search would make the algorithm much faster, and converge in fewer number of iterations.

We use a backtracking search on the curve $R(\tau)$ in order to find an appropriate τ value. The curvilinear search algorithm can be summarized as Algorithm 6, where $F'(R(0)Q^{(k)})$ is the derivative of $F(R(\tau)Q^{(k)})$ with respect to τ at $\tau = 0$, and is given by $F'(R(0)Q^{(k)}) = \mathrm{tr}(-G^T H Q^{(k)})$. τ_0 is a positive value used as the initialization point of τ, and $0 < \rho < 1$ is constant.

Algorithm 6 Curvilinear Search

$\tau = \tau_0$
while $F(R(\tau)Q^{(k)}) > F(R(0)Q^{(k)}) + \rho\tau F'(R(0)Q^{(k)})$ **do**
$\tau = \tau/2$
end while

These iterations can run very quickly, and also ensure that the gradient descent algorithm always decays in each step, and eventually it will converge to a minimum point.

Why $Q_1 = Q_2$ is a good assumption? Putting aside the different inverse scalings of A and B and the permutation of their respective columns and rows, the solution to NMF need not be unique in general. Necessary and sufficient conditions for the uniqueness of NMF solution are discussed in [14]. Consequently, solely assuming the positivity of A and B may not lead to unique solutions. Therefore, there is no way to recover the original A and B matrices in general. This could be justified by the following handwaving argument. Recall that we can always write $A = \tilde{U}Q_1$, and $B^T = \tilde{V}Q_2$, so that $Q_1 Q_2^T = I$. Now consider $\tilde{Q}_1 = Q_1 + \Delta Q_1$, and $\tilde{Q}_2 = Q_2 + \Delta Q_2$, so that we still have $\tilde{Q}_1 \tilde{Q}_2^T = I$. Assuming that ΔQ_1, and ΔQ_2 are small we have

$$\begin{aligned} Q_1 \Delta Q_2^T + \Delta Q_1 Q_2^T &= 0 \\ \tilde{U}(Q_1 + \Delta Q_1) &\geq 0 \\ \tilde{V}(Q_2 + \Delta Q_2) &\geq 0 \end{aligned}$$

where the first equality is derived directly from $\tilde{Q}_1 \tilde{Q}_2^T = I$ by ignoring the quadratic terms. Any ΔQ_1 and ΔQ_2 that satisfies the mentioned criteria can generate a feasible solution to the NMF problem, and there is no way to guarantee that the set of feasible nonzero solutions for ΔQ_1 and ΔQ_2 is empty in general.

Since the solutions are not unique in general, it is likely to find a solution pair (A, B) for which $Q_1 = Q_2 = Q$.

Therefore, by limiting ourselves to the Stiefel manifold, we are limiting the search to such solutions. According to our simulations in Section IV, this assumption works well for the randomly generated A and B, and Algorithm 5 results in the smallest residuals compared to the other algorithms discussed while having a reasonable running time. It is worth noting that in the case of symmetric nonnegative matrix factorization, i.e. $A = B^T$, the $Q_1 = Q_2 = Q$ assumption is correct and we have $p = 0.5$ in our formulation [14]. In fact, [14] has proposed an approach for *symmetric* nonnegative matrix factorization which is similar to ours in terms of changing the search space to Q. However, they perform alternate minimizations on the original NMF problem rather than using algorithms for Stiefel manifold optimization.

IV. Simulation Results

In Figure 1, we look at the comparison of how the mentioned methods reduce the normalized NMF residual $\frac{\|Y-\hat{A}\hat{B}\|_F^2}{\|Y\|_F^2}$. For this simulation we have set $m = 200$, $n = 150$, $r = 15$ and have considered a total of 100 runs. In each run, the matrices $A \in \mathbb{R}^{m\times r}$ and $B \in \mathbb{R}^{r\times n}$ are generated with uniformly random elements in $[0, 1]$, and $Y = AB$ is calculated after normalizing the columns of A and B. Then, the normalized NMF residual is calculated for each of the methods over 200 iterations to make sure all methods almost converge. Figure 1 shows the logarithm of the average of the residuals over all 100 runs. All the alternate minimization methods A-AM, M-AM, and ADMM are initialized with SVD based initialization method in [6] to make a fair comparison with our Q-Method which also utilizes the SVD as the starting point. The hyperparameters of methods, such as ρ in ADMM, τ in Q-AM, and p in Q-Method are tuned to the problem at hand; so that, all methods exhibit good convergence properties.

As we observe, the Q-Method decreases the NMF residual better than the other methods throughout the whole 200 iterations although it is not directly minimizing $\|Y - AB\|_F^2$ like ADMM, A-AM, and M-AM methods. This suggests that as the solution to NMF is not unique in general, we might be better off limiting ourselves to a particular set of solutions, i.e. $Q_1 = Q_2$, and benefit from Stiefel manifold optimization literature. As ADMM, A-AM and M-AM methods are using some kind of alternate minimizations on the original NMF problem, they are expected to perform better than Q-AM in reducing $\|Y - AB\|_F^2$. This is due to the fact that Q-AM uses alternate minimizations on another problem rather than the exact original NMF in Eq. 2.

Figure 2 compares the recovered $\hat{A}$ and $\hat{B}$ matrices at each iteration to the original A and B used to create Y over the total runs. The vertical axis shows the logarithm of the residual $\frac{\|A-\hat{A}\|_F^2}{\|A\|_F^2} + \frac{\|B-\hat{B}\|_F^2}{\|B\|_F^2}$. As we mentioned earlier, our primary concern is to reduce the NMF residual $\|Y - AB\|_F^2$, and looking for the original A and B is not a well defined problem since the solutions need not be unique. However, we have included Figure 2 here just to show that even the $Q_1 = Q_2 = Q$ restricted solutions found by Q-Method can get very

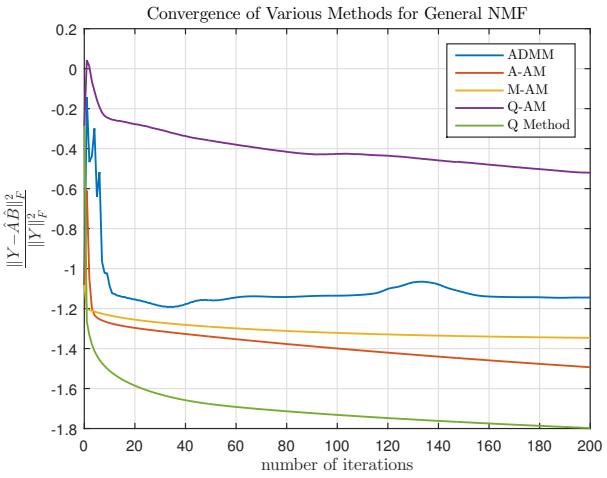

Fig. 1: Convergence of different methods on randomly generated A and B

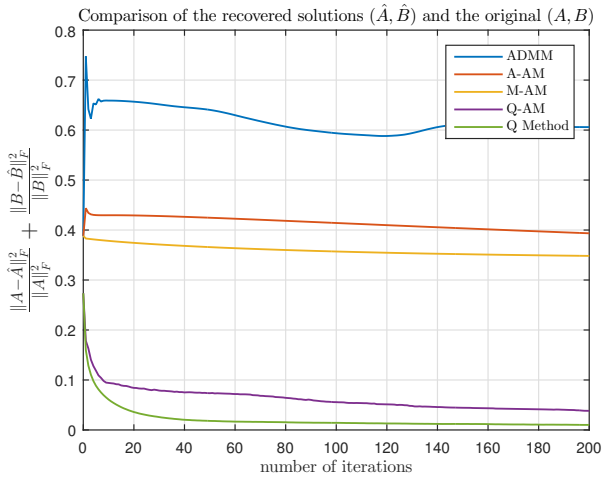

Fig. 2: Comparison of recovered solutions $\hat{A}$ and $\hat{B}$ with the original randomly generated A and B

close to the original random matrices A and B compared to other methods. We have considered the optimal scaling and column/row permutations of $\hat{A}$ and $\hat{B}$ with respect to A and B at each iteration to solve the scaling and permutation issues while comparing (A, B) with $(\hat{A}, \hat{B})$.

As the considered existing approaches A-AM, M-AM, and ADMM all directly do some kind of descent on the original residual $\|Y - AB\|_F^2$, unlike the Q-AM and Q-Method, we can initialize these methods with the final output of the Q-Method to further decrease the original residual. As an example, Figure 3 shows how this initialization boosts the performance of the A-AM method for the same $m = 200$, $n = 150$, and $r = 15$ setting in other simulations.

Running Time: Among the discussed algorithms, the M-AM method is the fastest as it does not require any kind of backtracking and is just a multiplicative update rule. The most computationally demanding step in Q-Method is the

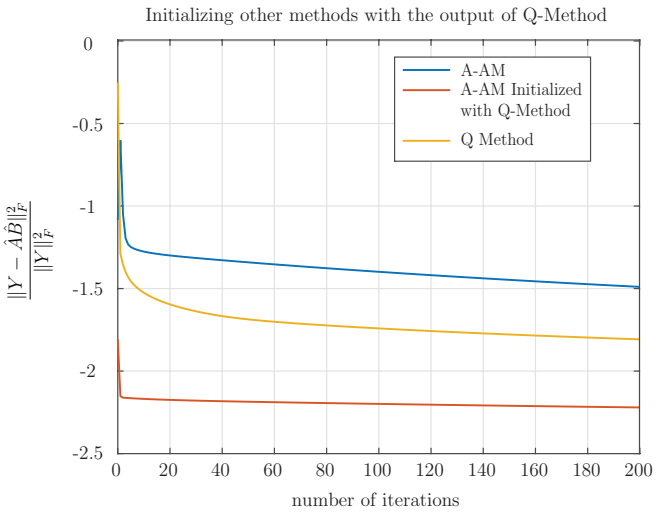

Fig. 3: Comparison of recovered solutions $\hat{A}$ and $\hat{B}$ with the original randomly generated A and B

calculation of $R(\tau)$ in Algorithm 5 which requires a matrix inversion. As in most of the applications as well as our simulations r is set to be a small value ($r \ll \min\{m, n\}$), this step does not cause any run time issues, unless r is set to a large value. According to our simulations for the $m = 200$, $n = 150$, and $r = 15$ setting, the Q-Method is actually faster than the other alternate minimization methods which require some sort of backtracking (A-AM, ADMM, Q-AM), while its final residual $\|Y - AB\|_F^2$ is smaller according to Figure 1.

V. Discussion

We have proposed a new algorithm, the Q-Method, for the basic nonnegative matrix factorization. This method transforms the search space from (A, B) to (Q_1, Q_2), and makes the simplifying assumption that there exists a solution for which $Q_1 = Q_2 = Q$. As mentioned earlier, the solution to NMF is not unique in general. Making this assumption results in a non-convex optimization problem in which a convex objective function has to be minimized over the set of orthonormal matrices known as the Stiefel manifold. Using Stiefel manifold optimization algorithms we obtain a solution for the original problem as $A = \tilde{U}Q^*$ and $B^T = \tilde{V}Q^*$, where Q^* is the output of the Stiefel manifold optimization algorithm and $\tilde{U}$ and $\tilde{V}$ are calculated from the compact SVD of the observation matrix as discussed.

We have analyzed the algorithm for the case in which the nonnegative rank of the observation matrix is in fact equal to the rank of the matrix, i.e. r. However, this is not the case in real applications since Y is usually a noisy matrix and r is mostly determined by the user regardless of the rank of Y. Considering the best rank r approximation of Y as Y_r given by SVD, we can implement the Q-Method to find a nonnegative factorization for Y_r. As Y_r has the minimum distance to Y among the rank r matrices and factorizations, the factorization calculated by the Q-Method should still be close to Y, i.e. the Frobenius norm of their difference should be small.

Our simulations on randomly generated matrices indicate that the Q-Method has a superior performance over the other popular NMF methods discussed. As many other NMF methods directly minimize the objective function $\|Y - AB\|_F^2$ on A and B, unlike the Q-Method, the output of the Q-Method can serve as an initialization point for these algorithms to further bring down the residual $\|Y - \hat{A}\hat{B}\|_F^2$.

The Stiefel manifold optimization algorithm in Q-Method requires performing a $r \times r$ matrix inversion at each iteration. If r is set to a large value, this step might increase the running time of the algorithm significantly. However, if r is small, as with most NMF applications, Q-Method should have a comparable running time to other discussed methods which use backtracking. In fact, for the dimensions in our simulations, Q-Method took less time to converge compared to A-AM, ADMM, and Q-AM.

References

[1] Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[2] Andrzej Cichocki, Rafal Zdunek, Anh Huy Phan, and Shun-ichi Amari. *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons, 2009.

[3] Yu-Xiong Wang and Yu-Jin Zhang. Nonnegative matrix factorization: A comprehensive review. *IEEE Transactions on Knowledge and Data Engineering*, 25(6):1336–1353, 2013.

[4] Sanjeev Arora, Rong Ge, and Ankur Moitra. Learning topic models–going beyond svd. In *IEEE 53rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1–10, 2012.

[5] Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562, 2001.

[6] Christos Boutsidis and Efstratios Gallopoulos. Svd based initialization: A head start for nonnegative matrix factorization. *Pattern Recognition*, 41(4):1350–1362, 2008.

[7] Sanjeev Arora, Rong Ge, Yonatan Halpern, David M Mimno, Ankur Moitra, David Sontag, Yichen Wu, and Michael Zhu. A practical algorithm for topic modeling with provable guarantees. In *International Conference on Machine Learning*, pages 280–288, 2013.

[8] Zaiwen Wen and Wotao Yin. A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142(1-2):397–434, 2013.

[9] Chih-Jen Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19(10):2756–2779, 2007.

[10] Patrick L Combettes and Valérie R Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling & Simulation*, 4(4):1168–1200, 2005.

[11] Jiho Yoo and Seungjin Choi. Orthogonal nonnegative matrix factorization: Multiplicative updates on stiefel manifolds. In *International Conference on Intelligent Data Engineering and Automated Learning*, pages 140–147. Springer, 2008.

[12] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, January 2011.

[13] Seungjin Choi. Algorithms for orthogonal nonnegative matrix factorization. In *IEEE International Joint Conference on Neural Networks*, pages 1828–1832, 2008.

[14] Kejun Huang, Nicholas D Sidiropoulos, and Ananthram Swami. Nonnegative matrix factorization revisited: Uniqueness and algorithm for symmetric decomposition. *IEEE Transactions on Signal Processing*, 62(1):211–224, 2014.