FAST GRADIENT METHODS FOR SYMMETRIC NONNEGATIVE MATRIX FACTORIZATION

RADU-ALEXANDRU DRAGOMIR, JÉRÔME BOLTE, AND ALEXANDRE D'ASPREMONT

ABSTRACT. We describe fast gradient methods for solving the symmetric nonnegative matrix factorization problem (SymNMF). We use recent results on non-Euclidean gradient methods and show that the SymNMF problem is smooth relatively to a well-chosen Bregman divergence. This approach provides a simple hyper-parameter-free method which comes with theoretical convergence guarantees. We also discuss accelerated variants. Numerical experiments on clustering problems show that our algorithm scales well and reaches both state of the art convergence speed and clustering accuracy for SymNMF methods.

1. INTRODUCTION

Nonnegative Matrix Factorization (NMF) is a popular unsupervised learning method. Similar to factor analysis, it is tailored for data sets that contain a natural nonnegativity constraint, e.g. word frequencies in text analysis, pixel brightness in imaging problems. Given a nonnegative matrix $M \in \mathbb{R}_+^{n\times m}$ and a target rank $r \leq \min(n,m)$, NMF seeks a decomposition into the product of two terms

$$M \approx XY^T$$

such that $X \in \mathbb{R}_+^{n\times r}$ and $Y \in \mathbb{R}_+^{m\times r}$ are nonnegative matrices. NMF has found a broad range of applications, from image processing, text mining [Lee & Seung, 1999], music analysis [Févotte & Idier, 2011], biology [Brunet et al., 2004], to astronomy [Berne et al., 2007], to cite only a few examples. NMF is commonly performed by solving the following optimization problem

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|M - XY^T\|_F^2 \\ \text{subject to} & X, Y \geq 0 \end{array} \tag{NMF}$$

in the variables $X \in \mathbb{R}_+^{n\times r}$ and $Y \in \mathbb{R}_+^{m\times r}$, where the inequality constraint is meant componentwise, $M \in \mathbf{S}_n$ is a given symmetric nonnegative matrix and r is the target rank. Other losses such as Kullback-Leibler divergence are possible in (NMF), but we focus here on the quadratic loss which is the most widely used.

Symmetric Nonnegative Matrix Factorization (SymNMF) is a variant of NMF where the two factors are constrained to be identical. This amounts to solving

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|M - XX^T\|_F^2 \\ \text{subject to} & X \geq 0 \end{array} \tag{SymNMF}$$

in the variable $X \in \mathbb{R}^{n\times r}$, where the inequality constraint is meant componentwise, $M \in \mathbf{S}_n$ is a given symmetric nonnegative matrix and r is the target rank.

SymNMF is used as a probabilistic clustering or graph clustering technique [He et al., 2011, Kuang et al., 2015]. In particular, it has been shown by Ding et al. [2005] that it can be interpreted as a nonnegative relaxation of the kernel k-means algorithm. In these applications, M is a pairwise similarity matrix between data points $\{x_1, \ldots, x_n\}$, r is the desired number of clusters, and a matrix X provided by SymNMF can be interpreted as a soft clustering assignment: X_{ik} is the likelihood that the point x_i belongs to the cluster k. Numerical experiments by Kuang et al. [2015] have shown that SymNMF achieves the state of the art clustering accuracy on several documents and image data sets.

Date: January 31, 2019.

Solving SymNMF. Both NMF and SymNMF are nonconvex minimization problems for which there is no hope to find a global solution. Exact NMF has been shown to be NP-hard [Vavasis, 2008]. Even though some recent results prove exact recovery of the factorized matrix [Arora et al., 2011], they rely on stringent and somewhat unrealistic "anchor words" conditions. Empirically, different solvers typically produce markedly different local solutions. Hence, current solvers are *local*, in the sense that they only seek a stationary point of the objective function. However, quite surprisingly, these local solutions are often satisfying enough for practical purposes. Therefore, the current challenge is to develop simple fast local solvers that scale well with the growing volume of data available.

Although the two problems look similar, NMF is currently easier to solve than SymNMF. This is because NMF has a favorable block structure that allows the application of efficient alternating algorithms. Indeed, if we fix one of the two matrices in NMF, then the minimization problem simply becomes a convex Nonnegative Least Squares (NLS) problem, for which several efficient algorithms are available, such as Block Principal Pivoting [Kim & Park, 2013]. On the other hand, if we consider an alternating minimization method on the columns of X and Y, then the iterates can be computed in closed form. The resulting algorithm is the well-known HALS [Cichocki & Phan, 2009] which is arguably the state-of-the-art of NMF solvers.

SymNMF, however, does not enjoy the same block structure, which makes it a harder problem. The current solution methods can be split into two categories.

Direct solvers. There have been several attempts at solving the original problem. He et al. [2011] propose multiplicative update rules that are guaranteed to monotonically decrease the objective. Kuang et al. [2015] apply two schemes: a projected gradient descent method, and Newton-like algorithm. The projected gradient method suffers from extremely slow convergence, while the Newton one is efficient on small problems but scales poorly because of the $O(n^3)$ complexity per iteration. Vandaele et al. [2016] propose a coordinate descent method, using the fact that the objective function is a fourth-order polynomial whose minimization in one coordinate can be computed in closed form. However, it does not enjoy the same efficiency on large-scale problems as the HALS algorithm for NMF as the minimization has to be carried on each separate entry of the matrix X.

Nonsymmetric relaxations. Another idea is to use a mere penalty method [Kuang et al., 2015], relaxing thus SymNMF into the following penalized nonsymmetric problem.

$$\begin{array}{ll} \text{minimize} & \|M - XY^T\|_F^2 + \mu\|X - Y\|_F^2 \\ \text{subject to} & X, Y \geq 0 \end{array} \tag{P-NMF}$$

in the variables $X, Y \in \mathbb{R}^{n\times r}$, with parameter $\mu \geq 0$. This formulation is very similar to NMF and can be solved by the same fast alternating algorithms that exploit the block structure. The authors argue that if the penalization parameter μ is large enough, then X and Y are very close and the solution is similar to the one found by SymNMF. Lu et al. [2017] propose an alternating direction method of multipliers (ADMM) for tackling this problem, and prove convergence of a subsequence to a stationary point of SymNMF for μ large enough. Recently, Zhu et al. [2018] proved that for μ greater than a certain threshold, any critical point (X^*, Y^*) of P-NMF is such that X^* is a critical point of SymNMF, so the problems become equivalent in a certain sense. They also prove, in the same setting, the convergence of two NMF algorithms adapted to P-NMF: SymANLS and SymHALS.

Numerical experiments in [Zhu et al., 2018] seem to show that algorithms of the the second category have much faster convergence. However, they require a delicate tuning of the penalization coefficient μ, because it not only affects convergence speed, but also the solution quality. Overall, Kuang et al. [2015] provide evidence that methods solving SymNMF directly show better clustering accuracy on text and image datasets for example.

Contributions. In this work, we introduce a new approach for solving SymNMF, based on a recent line of work on non-Euclidean gradient methods [Bauschke et al., 2017, Bolte et al., 2018] and subsequent work [Lu et al., 2018].

A critical issue with gradient algorithms is the choice of step sizes which can impact dramatically the efficiency of the method. When a function f has a Lipschitz continuous gradient with constant L, i.e.,

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\| \tag{1}$$

this implies

$$f(x) \leq f(y) + \langle \nabla f(y), x - y\rangle + \frac{L}{2}\|x - y\|^2 \tag{2}$$

for $x, y \in \mathbf{dom}\, f$, then gradient methods work for any fixed step size in the interval $(0, \frac{1}{L})$.

This smoothness assumption is used in the broad majority of the theoretical analyses of gradient algorithms, yet there are many cases where it is not satisfied [Bauschke et al., 2017, Bolte et al., 2018]. In particular, it does not hold for the objective function of SymNMF.

Of course, there is a way to circumvent this issue to apply classical Euclidean methods. It suffices to use an Armijo line search [Lin, 2007, Kuang et al., 2015], arguing that the objective function is Lipschitz continuous on bounded subsets. However, in some cases, line search procedures may generate very small step sizes which in turn involves costly subroutines. This explains why projected gradient algorithms for NMF and SymNMF are generally inefficient [Zhu et al., 2018].

The NoLips algorithm, introduced by Bauschke et al. [2017] for convex functions and extended to the nonconvex setting in Bolte et al. [2018], is a gradient method designed for minimizing functions that satisfy a more general condition than Lipschitz continuity of the gradient in (2), also known as *relative smoothness* [Lu et al., 2018]:

$$f(x) \leq f(y) + \langle \nabla f(y), x - y\rangle + LD_h(x, y), \tag{3}$$

for $x, y \in \mathbf{dom}\, f$, where the Euclidean distance has been replaced by the Bregman "distance" $D_h(x, y) = h(x) - h(y) - \langle \nabla h(y), x - y\rangle$, with h a properly chosen convex function. If $h(x) = \frac{1}{2}\|x\|^2$, we recover the Euclidean case in (2). Choosing other types of functions h may adapt better to the geometry of the objective. In the case of SymNMF, the loss function appears to satisfy the generalized Lipschitz condition (3) for some degree four polynomial kernel h.

Outline. Section 2 sets the framework for the general NoLips algorithm. In Section 3, this methodology is used to derive a gradient algorithm for solving SymNMF. Convergence towards a stationary point is established. We introduce in Section 4 two fast variants, the first one based on a dynamical step size strategy, and the second one inspired from a Nesterov-type acceleration scheme. Section 5 reports numerical results on synthetic and real-world datasets. Finally, we discuss other possible improvements in Section 6.

Notations. $\langle \cdot, \cdot \rangle$ and $\| \cdot \|$ denote respectively the standard Euclidean inner product and norm, on any Euclidean space that will be clear from context. For a matrix $X \in \mathbb{R}^{n\times r}$ we denote by X_i its i-th row and by $X_{:,k}$ its k-th column. We define the $\| \cdot \|_{1,\infty}$ norm of a square symmetric matrix $M \in \mathbf{S}_n$ as

$$\|M\|_{1,\infty} = \max_{1\leq i\leq n} \|M_i\|_1 = \max_{1\leq i\leq n} \sum_{j=1}^{n} |M_{ij}| \tag{4}$$

For $M \in \mathbf{S}_n$, we denote by $\|M\|_2$ its spectral norm, defined as

$$\|M\|_2 = \max_{x\in\mathbb{R}^n, x\neq 0} \frac{\|Mx\|}{\|x\|} \tag{5}$$

2. Generic NoLips Algorithm

We first briefly recall the general nonconvex NoLips algorithm of Bolte et al. [2018]. Let E be a Euclidean vector space endowed with an inner product $\langle \cdot, \cdot \rangle$. Our goal is to solve the minimization problem

$$\min_{x\in E} \Psi(x) \triangleq f(x) + g(x) \tag{6}$$

where f is a C^1 function, and g is a proper lower semicontinuous function (in our case, it will be the indicator of the nonnegative orthant).

Let $h : E \to \mathbb{R}$ be a differentiable strictly convex function, which is called the *distance kernel*. It generates the *Bregman distance*

$$D_h(x, y) = h(x) - h(y) - \langle \nabla h(y), x - y \rangle \tag{7}$$

Note that D_h is not a proper distance, it is sometimes referred to as a *Bregman divergence*. However D_h enjoys a distance-like separation property: $D_h(x, x) = 0$ and $D_h(x, y) > 0$ for $x \neq y$. We are now ready to define the notion of relative smoothness, also called generalized Lipschitz property.

Definition 2.1 (Relative smoothness)**.** *We say that a differentiable function $f : E \to \mathbb{R}$ is L-smooth relatively to the distance kernel h if there exists $L > 0$ such that*

$$f(x) \leq f(y) + \langle \nabla f(y), x - y \rangle + L D_h(x, y) \tag{RelSmooth}$$

for all $x, y \in E$.

For twice differentiable functions, relative smoothness has an elementary characterization: f is L-smooth relatively to h if and only if

$$\nabla^2 f(x) \preceq L \nabla^2 h(x), \quad x \in E \tag{8}$$

Notice that if $h(x) = \frac{1}{2}\|x\|^2$, then $D_h(x, y) = \frac{1}{2}\|x - y\|^2$ and we recover the Euclidean case (2). By using different kernels, such as logarithms or power functions, it is possible to show that (RelSmooth) holds for functions that do not have a Lipschitz continuous gradient (see Bauschke et al. [2017], Bolte et al. [2018] for examples).

NoLips algorithm. Now that we are equipped with a non-Euclidean geometry generated by h, we can define the Bregman proximal gradient map with step size λ as follows.

$$T_\lambda(x) = \operatorname*{argmin}_{u \in E} \left\{ g(u) + \langle \nabla f(x), u - x \rangle + \frac{1}{\lambda} D_h(u, x) \right\} \tag{9}$$

NoLips then simply means iterating this mapping as in Algorithm 1.

Algorithm 1 NoLips

```
Input: A function $h$ such that (RelSmooth) holds with relative Lipschitz constant $L$, and step size $0 < \lambda \leq \frac{1}{L}$.
  Initialize $x^0 \in E$ such that $\Psi(x^0) < \infty$.
  for k = 1,2,... do
      $x^k \in T_\lambda(x^{k-1})$
  end for
```

We implicitly assume that the Bregman proximal map $T_\lambda(x)$ is nonempty and simple to compute (see Bolte et al. [2018] and references therein). When h is the squared Euclidean norm, we recover the proximal gradient algorithm.

It can be easily be proved that, under the relative smoothness condition and with $\lambda \in (0, \frac{1}{L})$, the sequence $\{\Psi(x^k)\}_{k \geq 0}$ is nonincreasing. Convergence towards a critical point of the problem (6) is established in Bolte et al. [2018] under additional assumptions (boundedness of the sequence and Kurdyka-Lojasiewicz property) which will hold in our case.

3. NoLips algorithm for SymNMF

We now apply the NoLips algorithm to the SymNMF problem, defined by

$$\begin{array}{ll} \text{minimize} & f(X) \triangleq \frac{1}{2}\|M - XX^T\|^2 \\ \text{subject to} & X \geq 0 \end{array} \tag{SymNMF}$$

where X ranges over $\mathbb{R}^{n \times r}$.

As we will see below, f does not admit a globally Lipschitz continuous gradient, since its Hessian is unbounded as $\|X\| \to +\infty$. Hence to apply NoLips we need to identify an alternative distance kernel that fits the geometry of f. Define, for $X \in \mathbb{R}^{n\times r}$ and $\alpha \geq 0$, the distance kernel h by

$$h(X) \triangleq \frac{1}{4}\|X\|^4 + \frac{\alpha}{2}\|X\|^2. \tag{10}$$

We now show that, with the proper choice of the coefficient α, h is a kernel adapted to f.

Proposition 3.1. *Suppose $\alpha \geq \frac{1}{3}\min(\|M\|_2, \|M\|_{1,\infty})$, then the function f is 6-smooth relatively to h.*

Proof. The quadratic form induced by the Hessian of f if given by

$$\begin{aligned}\langle U, \nabla^2 f(X)U\rangle &= \|UX^T + XU^T\|^2 + 2\langle UU^T, XX^T\rangle - 2\langle UU^T, M\rangle \\ &\leq 2(\|XU^T\|^2 + \|UX^T\|^2) + 2\|UU^T\|\,\|XX^T\| - 2\langle UU^T, M\rangle \\ &\leq 6\|X\|^2\|U\|^2 - 2\langle UU^T, M\rangle\end{aligned} \tag{11}$$

for $X, U \in \mathbb{R}^{n\times r}$, where we used the Cauchy-Schwarz inequality and the submultiplicative property of the Frobenius norm. Now, there are two ways we can bound the second term. First, if we write $U_{:,k}$ the k-th column of U, we have

$$\begin{aligned}-2\langle UU^T, M\rangle &= -2\langle U, MU\rangle \\ &= -2\sum_{k=1}^{r}\langle MU_{:,k}, U_{:,k}\rangle \\ &\leq 2\sum_{k=1}^{r}\|M\|_2\|U_{:,k}\|^2 \\ &= 2\|M\|_2\|U\|^2\end{aligned} \tag{12}$$

On the other side, rembembering that U_i denotes the i-th row of U and that M is a symmetric matrix we also get

$$\begin{aligned}-2\langle UU^T, M\rangle &= -2\sum_{1\leq i,j\leq n} M_{i,j}U_iU_j^T \\ &\leq 2\sum_{1\leq i,j\leq n} M_{i,j}\|U_i\|\|U_j\| \\ &\leq \sum_{1\leq i,j\leq n} M_{i,j}\big(\|U_i\|^2 + \|U_j\|^2\big) \\ &= 2\sum_{1\leq i\leq n}\|M_i\|_1\|U_i\|^2 \\ &\leq 2\|M\|_{1,\infty}\|U\|^2\end{aligned} \tag{13}$$

Combining the two bounds (12) and (13), along with our choice for the constant α yields

$$-2\langle UU^T, M\rangle \leq 2\min\left(\|M\|_2, \|M\|_{1,\infty}\right)\|U\|^2 \leq 6\alpha\|U\|^2$$

Together with (11), we get

$$\langle U, \nabla^2 f(X)U\rangle \leq 6\big(\|X^2\| + \alpha\big)\|U\|^2 \tag{14}$$

On the other hand, the Hessian of h is such that

$$\begin{aligned}\langle U, \nabla^2 h(X)U\rangle &= \|X\|^2\|U\|^2 + 2\langle X, U\rangle^2 + \alpha\|U\|^2 \\ &\geq \big(\|X\|^2 + \alpha\big)\|U\|^2\end{aligned} \tag{15}$$

Combining (14) and (15) shows that for every matrix $X \in \mathbb{R}^{n\times r}$ we have $\nabla^2 f(X) \preceq 6\nabla^2 h(X)$, hence f is 6-smooth relatively to h [Bauschke et al., 2017]. ■

Remark. Usually, the spectral norm $\|M\|_2$ is smaller thant $\|M\|_{1,\infty}$ an thus provides a better estimate for the constant α, but we evoke both as the latter can be much faster to compute for large matrices.

In order to make the algorithm applicable, we need to compute the Bregman proximal gradient map efficiently.

$$T_\lambda(X) = \underset{U \geq 0}{\operatorname{argmin}} \left\{ f(X) + \langle \nabla f(X), U - X \rangle + \frac{1}{\lambda} D_h(U, X) \right\} \tag{16}$$

Writing $\Pi_+(X) = \max(X, 0)$ (entrywise), the projection on the nonnegative orthant, we have the following result.

Proposition 3.2. *The Bregman proximal gradient step $T_\lambda(X)$ is given by*

$$T_\lambda(X) = \frac{1}{z^*}\Pi_+(Q)$$

where

$$\begin{aligned} Q &= \nabla h(X) - \lambda \nabla f(X) \\ &= (\|X\|^2 + \alpha)X - 2\lambda(XX^T - M)X \end{aligned}$$

and z^* is the unique real solution to the cubic equation

$$z^2(z - \alpha) = \|\Pi_+(Q)\|^2. \tag{17}$$

Proof. Omitting constant terms, the minimization problem (16) reduces to

$$\begin{aligned} T_\lambda(X) &= \underset{U \geq 0}{\operatorname{argmin}} \left\{ \langle \lambda \nabla f(X), U \rangle + h(U) - \langle \nabla h(X), U \rangle \right\} \\ &= \underset{U \geq 0}{\operatorname{argmin}} \left\{ h(U) - \langle Q, U \rangle \right\} \end{aligned}$$

where $Q = \nabla h(X) - \lambda \nabla f(X)$. This is a strictly convex coercive minimization problem, hence it has a unique solution $T_\lambda(X) = U^\star$ that satisfies the KKT optimality conditions

$$U^\star \geq 0 \tag{18}$$

$$\nabla h(U^\star) - Q \geq 0 \tag{19}$$

$$\langle \nabla h(U^\star) - Q, U^\star \rangle = 0 \tag{20}$$

Recall that the gradient of h is $\nabla h(U) = (\|U\|^2 + \alpha)U$, and therefore it is positively correlated to U, so it is also nonnegative. It follows that, for $1 \leq i \leq n$ and $1 \leq k \leq r$,

- If $Q_{ik} < 0$, then $\nabla h(U^\star)_{ik} - Q_{ik} > 0$, so (19)-(20) impose that $U^\star_{ik} = 0$
- If $Q_{ik} > 0$, then by (19), $\nabla h(U^\star)_{ik} > 0$ hence $U^\star_{ik} > 0$ so it must hold that $\nabla h(U^\star)_{ik} - Q_{ik} = 0$
- If $Q_{ik} = 0$, we must have from (19)-(20) that $\nabla h(U^\star)_{ik} = 0$

All these conditions can be summarized as

$$(\|U^\star\|^2 + \alpha)U^\star = \Pi_+(Q) \tag{21}$$

where Π_+ is the projection on the nonnegative orthant. Now, the solution is determined up to the scalar value $z^* := \|U^\star\|^2 + \alpha$. By applying the squared Frobenius norm to (21), we find that z^* is the unique real solution of the cubic equation

$$z^{*2}(z^* - \alpha) = \|\Pi_+(Q)\|^2 \tag{22}$$

Hence, after solving for z^*, it is easy to see that $U^\star = \frac{1}{z^*}\Pi_+(Q)$ is the unique solution to the KKT optimality conditions. ■

Note that the real solution z^* of the cubic equation

$$z^2(z - \alpha) = c$$

can be computed in closed form using Cardano's method, which yields

$$\begin{aligned} z^* &= \frac{\alpha}{3} + \sqrt[3]{\frac{c+\sqrt{\Delta}}{2} + \frac{\alpha^3}{27}} + \sqrt[3]{\frac{c-\sqrt{\Delta}}{2} + \frac{\alpha^3}{27}} \\ \Delta &= c^2 + \frac{4}{27}c\alpha^3. \end{aligned} \tag{23}$$

Algorithm 2 describes the NoLips scheme for solving SymNMF.

Algorithm 2 NoLips for SymNMF

```
Input: Nonnegative matrix $M \in \mathbf{S}_n$. Rank $r$, step size $\lambda > 0$.
Initialize $X \in \mathbb{R}_+^{n\times r}$ randomly.
Set $\alpha = \frac{1}{3}\min(\|M\|_2, \|M\|_{1,\infty})$
Set $z^* = \|X\|^2 + \alpha$
repeat
    Compute $\nabla f = 2(XX^T - M)X$.
    Form $Q_+ = \max(0, z^*X - \lambda\nabla f)$.
    Set $z^*$ to be the unique real solution to
        $z^2(z-\alpha) = \|Q_+\|^2$
    (see (23) for closed form).
    Set $X \leftarrow Q_+/z^*$
until Convergence criterion is satisfied.
Output: Solution matrix $X$.
```

Complexity. Let $p \leq n^2$ be the number of stored entries in the matrix M. In many cases M is sparse so that the storage is cheap $p << n^2$ which results in a significant speedup of the minimization methods we consider.

The most expensive step of Algorithm 2 is the gradient computation, which costs $O(nr^2 + pr)$ flops when carried out in the proper order, $X(X^TX) - MX$. The other steps are linear in the size of the matrix X which is nr, so the overall complexity per iteration is $O(nr^2 + pr)$.

Convergence. Let us show that the NoLips-SymNMF algorithm has the desired theoretical convergence properties.

Theorem 3.3. *Let $\{X^k\}_{k\geq 0}$ be the sequence generated by Algorithm 2 with a step size $0 < \lambda < 1/6$, then*

(1) *The sequence $\{f(X^k)\}_{k\geq 0}$ is nonincreasing,*
(2) *The sequence $\{X^k\}_{k\geq 0}$ converges towards a critical point X^* of problem* (SymNMF).
(3) *There exist $C > 0$ and $\omega > 0$ such that we have the convergence rate $\|X^k - X^*\| \leq Ck^{-\omega}$.*

Proof. As shown in Proposition 3.1, the objective function f is 6-smooth relatively to h, with $\alpha = \|M\|_{1,\infty}/3$. Therefore, (1) is a direct consequence of [Bolte et al., 2018, Prop. 4.1]. Global convergence towards a critical point is more delicate to prove, and requires more assumptions which are fortunately true in our case. The kernel h is defined over the entire space $\mathbb{R}^{n\times r}$, it is strongly convex (because of the quadratic term $\frac{\alpha}{2}\|X\|^2$), ∇h and ∇f are Lipschitz continuous on bounded subsets of $\mathbb{R}^{n\times r}$. We also need to show that the sequence $\{X^k\}_{k\geq 0}$ is bounded, which is a consequence of (1) and the fact that the function f is coercive (we can easily see that $\|M - XX^T\|^2 \to +\infty$ when $\|X\| \to +\infty$).

The last requirement is that $f + I_{\mathbb{R}_+^{n\times r}}$ satisfies the nonsmooth Lojasiewicz property (see Bolte et al. [2007] for a definition), which has been shown to hold for semialgebraic functions [Bolte et al., 2007]. Since f is polynomial and $I_{\mathbb{R}_+^{n\times r}}$ is the indicator function of a semialgebraic set, their sum is a semialgebraic function which therefore verifies the Lojasiewicz property.

Since all these assumptions hold, [Bolte et al., 2018, Th. 4.1] applies and $\{X^k\}_{k\geq 0}$ converges towards a critical point of problem (SymNMF), with in addition a convergence rate as provided in [Bolte et al., 2018, Th. 6.3.]. ■

4. Fast variants

Although NoLips has good theoretical convergence properties, it turns out to be relatively slow in practice. We propose two different variants for improving the convergence speed: an adaptive step size strategy and a Nesterov-type acceleration scheme [Auslender & Teboulle, 2006].

4.1. **Dynamical step size strategy.** While the choice of a constant step size $\lambda < \frac{1}{6}$ guarantees convergence, it can be overly conservative, especially as X^k gets close to a sparse solution (see Section 6 for a discussion). Therefore, we propose in Algorithm 3 a more agressive dynamical step size strategy, following the scheme described in Nesterov [2007]: at each iteration, we ensure that the chosen step λ and the new iterate $X_{new} = T_\lambda(X)$ satisfy the decrease condition

$$f(X_{\text{new}}) \leq f(X) + \langle \nabla f(X), X_{\text{new}} - X \rangle + \frac{1}{\lambda} D_h(X_{\text{new}}, X) \tag{24}$$

where we recall that $D_h(\cdot,\cdot)$ is given by

$$D_h(X,Y) = h(X) - h(Y) - \langle \nabla h(Y), X - Y \rangle \tag{25}$$

and that $h(X) = \frac{1}{4}\|X\|^4 + \frac{\alpha}{2}\|X\|^2$ and $\nabla h(X) = (\|X\|^2 + \alpha)X$.

Algorithm 3 NoLips for SymNMF with dynamical step size

```
Input: Nonnegative matrix $M \in \mathbf{S}_n$. Rank $r$.
Initialize $X \in \mathbb{R}_+^{n\times r}$ randomly.
Set $\alpha = \frac{1}{3}\min(\|M\|_2, \|M\|_{1,\infty})$
Set $z^* = \|X\|^2 + \alpha$
Set $\lambda = 0.9/6$ and $\lambda_{\max} = 4r$
repeat
  Compute $\nabla f = 2(XX^T - M)X$.
  repeat
    Form $Q_+ = \max(0, z^*X - \lambda\nabla f)$.
    Set $z^*$ to be the unique real solution to
        $z^2(z-\alpha) = \|Q_+\|^2$
    (see (23) for closed form).
    Form $X_{\text{new}} = Q_+/z^*$
    Check if
        $f(X_{\text{new}}) \leq f(X) + \langle \nabla f(X), X_{\text{new}} - X \rangle + \frac{1}{\lambda} D_h(X_{\text{new}}, X)$    (26)
    if the decrease condition (26) is not satisfied then
      Set $\lambda \leftarrow \lambda/2$
    end if
  until decrease condition (26) is satisfied
  Set $X \leftarrow X_{\text{new}}$
  Set $\lambda \leftarrow \min(2\lambda, \lambda_{\max})$
until Convergence criterion is satisfied.
Output: Solution matrix $X$.
```

Note that in order to check the decrease condition, we need to do several evaluations of the objective function per iteration. As pointed out by Vandaele et al. [2016], the most efficient way to compute it is by using

$$\|M - XX^T\|^2 = \|M\|^2 - 2\langle MX, X \rangle + \|X^TX\|^2 \tag{27}$$

which has a $O(pr + nr^2)$ complexity (remember that p is the number of entries of M, if M is sparse), instead of $O(n^2r)$ for the naive way.

Note that we need to impose a maximal value for the step size, we choose the heuristic value $\lambda_{\max} = 4r$. This value is never attained in our experiments, but serves as a safeguard for theoretical guarantees. Now, the $\lambda_{\max}$ value along with the relative smoothness property allow us to bound the values of the dynamical step size:

Lemma 4.1. *Let $\{\lambda_k\}_{k\geq 0}$ be the sequence of step sizes chosen at each iteration of Algorithm 3. Then for every $k \geq 0$ we have*

$$\frac{1}{2L} \leq \lambda_k \leq \lambda_{\max} \tag{28}$$

Proof. The upper bound holds by construction of the algorithm. The lower bound comes from the the relative smoothness property: the condition (26) is true for every $\lambda \in (0, \frac{1}{L}]$, so the inner loop will stop whenever λ gets below $\frac{1}{L}$. ■

Complexity per iteration. From Lemma 4.1, we conclude that at each iteration, the number of inner loops is bounded by $\lfloor \log_2(L\lambda_{\max})\rfloor + 1 = \lfloor \log_2(24r)\rfloor + 1$. Since computing the Bregman proximal map and evaluating the decrease condition (26) can be done in $O(pr + nr^2)$ operations, we conclude that the overall complexity of an iteration is $O\big[(pr + nr^2)\log_2(r)\big]$. Therefore, we only lose a logarithmic factor in r compared to the fixed step size strategy.

This algorithm enjoys the same theoretical convergence properties as Algorithm 2 (we defer the proof to Appendix A).

Theorem 4.2. *Let $\{X^k\}_{k\geq 0}$ be the sequence generated by Algorithm 3.*

(1) *The sequence $\{f(X^k)\}_{k\geq 0}$ is nonincreasing.*
(2) *The sequence $\{X^k\}_{k\geq 0}$ converges towards a critical point X^* of Ψ.*
(3) *There exist $\omega > 0$ and $\theta > 0$ such that we have the convergence rate $\|X^k - X^*\| \leq Ck^{-\omega}$.*

Proof. See Appendix A. ■

4.2. **Nesterov acceleration.** We now consider a faster variant of NoLips which is based on an empirical adaptation of the Improved Interior Gradient Algorithm (IGA) of Auslender & Teboulle [2006], combined with a restart scheme [Roulet & D'Aspremont, 2017].

In his seminal work Nesterov [Nesterov, 1983] designed an accelerated gradient method with $O(1/k^2)$ complexity for convex functions with Lipschitz continuous gradient. This improved the worst case bound $O(1/k)$ of standard one step gradient descent at the price of a small extra computational cost. It is then natural to wonder whether NoLips possesses accelerated versions that would show the same improved rate for convex relatively smooth functions. As a matter of fact, an accelerated version of the Bregman gradient scheme has been proposed by Auslender & Teboulle [2006]. It is called Improved Interior Gradient Algorithm (IGA). For convex functions with a Lipschitz continuous gradient, IGA is proven to have a $O(1/k^2)$ convergence rate provided the kernel is elliptic. Whether or not this result can be extended to the general relatively smooth setting is a major open question. The work of Hanzely et al. [2018] started to address this issue for distance kernels h satisfying some crucial triangle scaling property.

Nonetheless, despite this major open question and the nonconvex features of our present problem, we propose to test the IGA scheme on SymNMF. We observe a significant numerical speedup over standard NoLips. This accelerated scheme involves two auxiliary variables Z and Y that are defined with linear interpolation steps. Using this along with the dynamical step size strategy yields Algorithm 4.

Finally, as accelerated methods can greatly benefit from restarting, we apply the restart scheme described in Roulet & D'Aspremont [2017]. It consists in setting a schedule $\{T_k\}_{k\geq 0}$ of number of iterations, and restarting Algorithm 4 according to this schedule (see Algorithm 5). This means that the algorithm's memory (in our case, the variables t and Z) is erased periodically.

We use the restart schedule advised for algorithms with sublinear convergence

$$T_k = \lfloor C\rho^k \rfloor \tag{29}$$

In our experiments, we set the parameters to $C = 10$ and $\rho = 5$.

Algorithm 4 Fast-NoLips for SymNMF

```
Input: A nonnegative matrix $M \in \mathbf{S}_n$, rank $r$.
Initialize $X \in \mathbb{R}_+^{n\times r}$ randomly.
Set $\alpha = \frac{1}{3}\min(\|M\|_2, \|M\|_{1,\infty})$
Set $z^* = \|M\|^2 + \alpha$
Set $\lambda = 0.9/6$
Set $t = 1$ and $Z = X$.
repeat
  Form $Y = (1 - t^{-1})Z + t^{-1}X$
  Compute $\nabla f = 2(YY^T - M)Y$
  Update the step size by doing:
  repeat
    Form $Q_+ = \max(0, z^*Y - \lambda\nabla f)$.
    Set $z^*$ to be the unique real solution to
        $z^2(z - \alpha) = \|Q_+\|^2$
    Form $Y_{\text{new}} = Q_+/z^*$
    if the decrease condition (24) is not satisfied with $(Y_{\text{new}}, Y)$ then
      Set $\lambda \leftarrow \lambda/2$
    end if
  until decrease condition (24) is satisfied with $(Y_{\text{new}}, Y)$
  Form $Q_+ = \max(0, z^*X - \lambda t\nabla f)$.
  Set $z^*$ to be the unique real solution to
      $z^2(z - \alpha) = \|Q_+\|^2$
  Set $X \leftarrow Q_+/z^*$
  Set $Z \leftarrow (1 - t^{-1})Z + t^{-1}X$
  Set $t \leftarrow \frac{1}{2}\left(1 + \sqrt{1 + 4t^2}\right)$
  Set $\lambda \leftarrow \min(2\lambda, \lambda_{\max})$
until Convergence criterion is satisfied.
Output: Solution matrix $X$.
```

Algorithm 5 Restart scheme for Fast-NoLips

```
Input: A nonnegative matrix $M \in \mathbf{S}_n$, rank $r$, and a schedule $\{T_k\}_{k\geq 0}$.
Initialize $X^0 \in \mathbb{R}_+^{n\times r}$ randomly.
for k = 0,1,... do
  Run Algorithm 4 with initial value $X^k$ for $T_k$ iterations.
  Set $X^{k+1}$ to be the output.
end for
```

5. Numerical experiments

5.1. **Algorithms.** We implemented the following algorithms for SymNMF.

- **Dyn-NoLips**: Algorithm 3 with a dynamical step size strategy.
- **Fast-NoLips**: the combination of a dynamical step size strategy, and acceleration scheme with restart (Algorithm 5).
- β**-SNMF** from [He et al., 2011], where we set $\beta = 0.99$ as advised by the authors.

- **PG.** Projected gradient algorithm with Armijo line search Kuang et al. [2015]. We used the standard values for the line search parameters $\beta = 0.1$ and $\sigma = 0.01$. Note that in order to be fair with the **Dyn-NoLips** algorithm, we used the same efficient way of computing the objective function and also an improved Armijo line search strategy [Lin, 2007, Algorithm 4.]
- **CD.** The coordinate descent scheme as in Vandaele et al. [2016].
- **SymANLS.** The algorithm in [Zhu et al., 2018] for solving the relaxed problem (P-NMF), with a block principal pivoting method [Kim & Park, 2013] for solving the NLS subproblems.
- **SymHALS** from [Zhu et al., 2018], also for solving (P-NMF). We used the efficient FastHALS implementation in [Cichocki & Phan, 2009].

For the SymANLS and SymHALS algorithms, we tuned the μ penalization parameter for best performance. We left out the Quasi-Newton algorithm from Kuang et al. [2015] because of its prohibitive $O(n^3)$ complexity for large datasets.

All algorithms were implemented in Julia [Jeff Bezanson, Alan Edelman & Shah, 2017] which is a highly-optimized numerical computing language. Since our algorithms have different complexity per iteration, it is essential to compare them in terms of running time, and Julia provides a fairly accurate way to do so as there is little interpreter overhead in loops, unlike Python for instance. Tests were run on a PC Intel CORE i7-4910MQ CPU @ 2.90 GHz x 8 with 32 Go RAM.

5.2. **Datasets.** Various synthetic and real-world datasets were used.

- **Synthetic sparse data.** In order to mimic a sparse similarity matrix, we generate $\widetilde{X} \in \mathbb{R}^{n\times r}$ with entries sampled from a Bernoulli distribution of probability $2/r$, and a noise matrix $N \in \mathbb{R}^{n\times n}$ with entries sampled from a standard Gaussian distribution. To ensure symmetry and nonnegativity, matrix M is then constructed as $M = \max(\widetilde{X}\widetilde{X}^T + \frac{1}{2}(N + N^T), 0)$.
- **Image.**
  - **CBCL**[1]: 2,429 images of faces of size 19×19
  - **ORL**[2]: 400 images of size 92×112 representing 40 subjects under different angles and facial expressions.
  - **Coil-20**[3]: 1440 images of size 128×128 representing 20 objects under various angles.
- **Text.**
  - **TDT2**[4]: dataset of 11,201 news articles classified in 96 semantic categories. We used the version provided by Cai et al. [2009, 2008, 2007, 2005], which has been restricted to the largest 30 categories, leaving a total of 9,394 documents.
  - **Reuters**[4]: dataset of news articles, which we restricted to the largest 25 categories, leaving a total of 7,963 documents.

For all image and text datasets, we construct a sparse similarity matrix M following the procedure described in Kuang et al. [2015]. We begin by computing the similarity graph between data points, using cosine similarity on term frequency vectors for text, and a Gaussian kernel for image (with the self-tuning method for the scale). The graph obtained is *sparsified* by keeping only the edges connecting the k-nearest neighbors, with $k = \lfloor \log_2 n \rfloor + 1$. Then, M is taken as a normalized version of the graph adjacency matrix.

5.3. **Convergence speed.** X^0 is initialized with entries drawn uniformly at random in $[0, 2\sqrt{\frac{m}{r}}]$ where m is the mean value of M. For each dataset, we run all the algorithms over 20 random initializations. We present the averaged value of $f(X^k) - f_{\min}$, where $f_{\min}$ is the minimal objective value found over all runs.

Figure 1 reports the results. Overall, the algorithms that show the best convergence speed are **Fast-NoLips**, **SymHALS**, and **Dyn-NoLips**. Even if **SymHALS** performs better on some datasets, it has the disadvantage of requiring tuning of the penalization parameter μ for an optimal performance. In our experiments, $\mu = 10$ for synthetic datasets, $\mu = 10^{-2}$ for image and $\mu = 10^{-1}$ for text showed the best

[1] http://cbcl.mit.edu/software-datasets/FaceData2.html

[2] https://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html

[3] http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php

[4] http://www.cad.zju.edu.cn/home/dengcai/Data/TextData.html

TABLE 1. Clustering accuracy for SymNMF algorithms on text and image datasets after 20 seconds of running time, averaged over 20 random initializations.

	ORL	Coil-20	TDT2	Reuters
Dyn-NoLips	**0.855**	0.753	0.829	0.458
Fast-NoLips	0.843	0.741	**0.850**	0.470
β-SNMF	0.847	0.688	0.800	**0.472**
PG	0.849	0.734	0.801	0.446
CD	0.834	0.724	0.399	0.269
SymHALS	0.850	**0.778**	0.832	0.439
SymANLS	0.845	0.734	0.520	0.324

convergence speed; the smallness of these values contrast with a penalization theory advising for large values.

Our algorithms seem to scale well with dimension, as their best performance relatively to **SymHALS** is on large text datasets (Figures 1(e) and 1(f)). Surprisingly, **PG** does not perform as bad as in experiments from previous work [Zhu et al., 2018, Kuang et al., 2015]. One reason that might explain this is the fact that we used the efficient method (27) for evaluating the objective function and the improved backtracking strategy Lin [2007]. Nonetheless, its speed is inferior to the one of **Dyn-NoLips**, demonstrating the benefit of using a non-Euclidean geometry in this problem.

We also note that **SymANLS**, which was found to be one of the fastest methods in previous work [Zhu et al., 2018, Kuang et al., 2015], is quite slow in our experiments despite a careful tuning of μ. Perhaps the solver we used in Julia for the NLS subproblems is not as efficient as Matlab's one.

5.4. **Application to clustering.** Most of the tested algorithms are proven to converge to stationary points of problem (SymNMF). However, because of the nonconvexity, they may converge to solutions of different quality. We evaluate this quality on clustering tasks, which is the principal application of SymNMF. If $\hat{X}$ is the solution matrix returned by an algorithm, then the data point i can be assigned to the label $\hat{l}_i = \operatorname{argmax}_{1\leq k\leq r} \hat{X}_{ik}$.

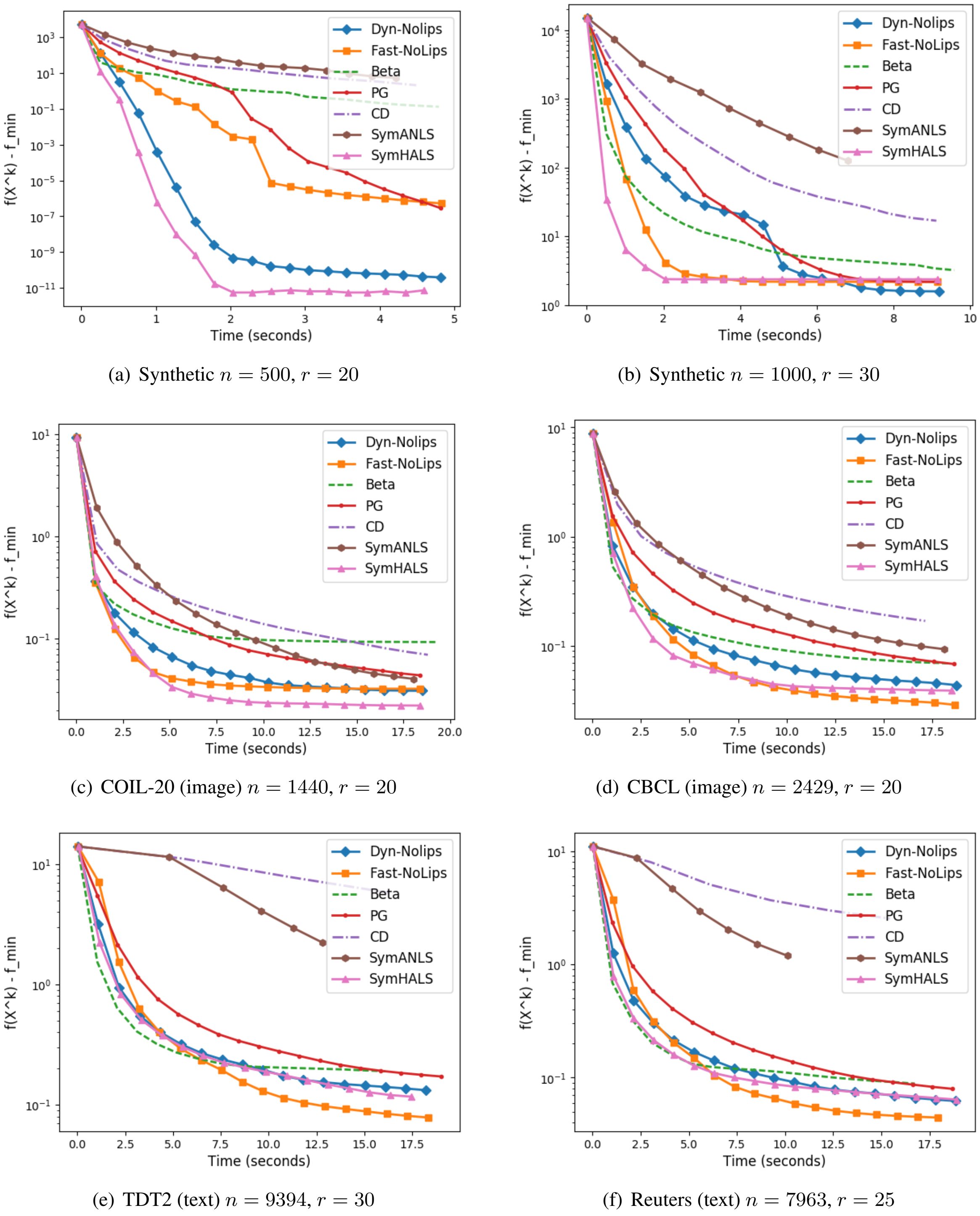
If the dataset has ground truth labels $\{l_i\}_{i=1\ldots n}$ (which is the case for **ORL**, **Coil-20**, **TDT2** and **Reuters**), then we can compute clustering accuracy, which is the fraction of correct labels, minimized over all possible permutations of indices Li et al. [2014]. Table 1 reports the clustering results for these 4 datasets after 20 seconds of running time. Overall, our algorithms show competitive clustering performance with regards to the other SymNMF methods.

6. DISCUSSION

In Proposition 3.1 we proved that the SymNMF objective function f is 6-smooth relatively to the kernel $h(X) = \frac{1}{4}\|X\|^4 + \alpha\|X\|^2$. Can we improve this constant? Recall that, in the proof of the relative smoothness inequality, we bound the Hessian of f in the following way:

$$\begin{aligned}\langle U, \nabla^2 f(X)U\rangle &= \|UX^T + XU^T\|^2 + 2\langle UU^T, XX^T\rangle - 2\langle UU^T, M\rangle \\ &\leq 6\|X^TX\|\|U\|^2 - 2\langle UU^T, M\rangle \\ &\leq 6\|X\|^2\|U\|^2 + 2\min\left(\|M\|_2, \|M\|_{1,\infty}\right)\|U\|^2\end{aligned} \tag{30}$$

The bound on the second term is quite tight and cannot be improved much. However, going from the second to the third line, the inequality $\|X^TX\| \leq \|X\|^2$ can be overly conservative in our problem. Indeed, in the special case where the columns of X are orthonormal, we have $X^TX = I_{r\times r}$. Therefore $\|X^TX\| = \sqrt{r}$ and $\|X\|^2 = r$, meaning that the relative smoothness constant could be improved up to the factor $\sqrt{r}$.



(a) Synthetic $n = 500$, $r = 20$

(b) Synthetic $n = 1000$, $r = 30$

(c) COIL-20 (image) $n = 1440$, $r = 20$

(d) CBCL (image) $n = 2429$, $r = 20$

(e) TDT2 (text) $n = 9394$, $r = 30$

(f) Reuters (text) $n = 7963$, $r = 25$

FIGURE 1. SymNMF objective gap $f(X^k) - f_{\min}$ averaged over 20 random initializations, for various sparse similarity matrices $M \in \mathbb{R}^{n\times n}$. Hyperparameters for SymHALS, SymANLS were tuned for best performance.

Yet, in the SymNMF problem, it often happens that the solution matrices have quasi-orthogonal columns (see Ding et al. [2005] for an informal argument). This means that, as the iterates X^k converge to the solution, the relative smoothness constant can be improved, and therefore we could take larger steps. This was the motivation behind the variants of NoLips with dynamical step size strategies.

7. Conclusion

We proposed a novel approach for solving the SymNMF problem revolving around the NoLips algorithm. We showed that the geometry induced by a fourth degree polynomial kernel adapts well to the SymNMF loss function.

We then derived two fast algorithms. **Dyn-NoLips** is a variant where we try at each iteration to increase aggressively the step size for which we proved a full convergence theory. The second one, **Fast-NoLips**, allows to increase further the convergence speed by adding a Nesterov-type acceleration scheme, but lacks of theoretical guarantees at this time.

Both variants showed very competitive speed and clustering accuracy with regards to the other SymNMF methods. Besides we do not need the delicate tuning a penalty parameter. In particular, the good performance of **Fast-NoLips** should motivate future attempts at establishing convergence guarantees for accelerated algorithms in the relatively smooth nonconvex setting.

Acknowledgements

Radu-Alexandru Dragomir would like to acknowledge support from an AMX fellowship as well as from the Air Force Office of Scientific Research, Air Force Material Command, USAF, under grant number FA9550-18-1-0226

Jérôme Bolte was partially supported by Air Force Office of Scientific Research, Air Force Material Command, USAF, under grant number FA9550-18-1-0226 and ANR OMS (ANR-17-CE23-0013-01).

Alexandre d'Aspremont is at CNRS & département d'informatique, École normale supérieure, UMR CNRS 8548, 45 rue d'Ulm 75005 Paris, France, INRIA and PSL Research University. The authors would like to acknowledge support from the *ML & Optimisation* joint research initiative with the *fonds AXA pour la recherche* and Kamet Ventures, as well as a Google focused award.

References

Arora, S., Ge, R., Kannan, R., and Moitra, A. Computing a nonnegative matrix factorization–provably. *Arxiv preprint arXiv:1111.0952*, 2011.

Auslender, A. and Teboulle, M. Interior gradient and proximal methods for convex and conic optimization. *SIAM Journal on Optimization*, 16(3):697–725, 2006.

Bauschke, H. H., Bolte, J., and Teboulle, M. A Descent Lemma Beyond Lipschitz Gradient Continuity: First-Order Methods Revisited and Applications. *Mathematics of Operations Research*, 42(2):330–348, 2017.

Berne, O., Tielens, A. G. G. M., Pilleri, P., and Joblin, C. Non-negative matrix factorization pansharpening : an application to mid-infrared astronomy. *ArXiv preprint arXiv:1003.0805v4*, 2007.

Bolte, J., Daniilidis, A., and Lewis, A. The Łojasiewicz Inequality for Nonsmooth Subanalytic Functions with Applications to Subgradient Dynamical Systems. *SIAM Journal on Optimization*, 17(4):1205–1223, 2007.

Bolte, J., Sabach, S., Teboulle, M., and Vaisbourd, Y. First order methods beyond convexity and lipschitz gradient continuity with applications to quadratic inverse problems. *SIAM Journal on Optimization*, 28(3):2131–2151, 2018.

Brunet, J. P., Tamayo, P., Golub, T. R., and Mesirov, J. P. Metagenes and molecular pattern discovery using matrix factorization. *Proc Natl Acad Sci U S A*, 101(12):4164–4169, 2004.

Cai, D., He, X., and Han, J. Document clustering using locality preserving indexing. *IEEE Transactions on Knowledge and Data Engineering*, 17(12):1624–1637, December 2005.

Cai, D., He, X., Zhang, W. V., and Han, J. Regularized locality preserving indexing via spectral regression. In *Proceedings of the 16th ACM conference on Conference on information and knowledge management (CIKM'07)*, pp. 741–750, 2007.

Cai, D., Mei, Q., Han, J., and Zhai, C. Modeling hidden topics on document manifold. In *Proceeding of the 17th ACM conference on Information and knowledge management (CIKM'08)*, pp. 911–920, 2008.

Cai, D., Wang, X., and He, X. Probabilistic dyadic data analysis with local and global consistency. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML'09)*, pp. 105–112, 2009.

Cichocki, A. and Phan, A. H. Fast local algorithms for large scale nonnegative matrix and tensor factorizations. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 2009.

Ding, C., He, X., and Simon, H. On the Equivalence of Nonnegative Matrix Factorization and Spectral Clustering. *Proceedings of the 2005 SIAM ICDM*, (4):126–135, 2005.

Févotte, C. and Idier, J. Algorithms for nonnegative matrix factorization with the beta-divergence. *Neural Computation*, 23(9), 2011.

Hanzely, F., Richt, P., and Xiao, L. Accelerated Bregman proximal gradient methods for relatively smooth convex optimization. *ArXiv preprint arXiv:1808.03045v1*, 2018.

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

Jeff Bezanson, Alan Edelman, S. K. and Shah, V. B. Julia : A Fresh Approach to Numerical Computing. *SIAM Review*, 59(1):65–98, 2017.

Kim, J. and Park, H. Fast Nonnegative Matrix Factorization: An Active-set-like Method and Comparisons. *SIAM Journal on Scientific Computing*, 33(6):3261–3281, 2013.

Kuang, D., Yun, S., and Park, H. SymNMF: nonnegative low-rank approximation of a similarity matrix for graph clustering. *Journal of Global Optimization*, 62(3):545–574, 2015.

Lee, D. D. and Seung, H. S. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401, 1999.

Li, L., Yang, J., Xu, Y., Qin, Z., and Zhang, H. Documents clustering based on max-correntropy nonnegative matrix factorization. *Proceedings - International Conference on Machine Learning and Cybernetics*, 2:850–855, 2014.

Lin, C.-J. Projected Gradient Methods for Nonnegative Matrix Factorization. *Neural Computation*, 2007.

Lu, H., Freund, R. M., and Nesterov, Y. Relatively-Smooth Convex Optimization by First-Order Methods, and Applications. *SIAM Journal on Optimization*, 28(1):333–354, 2018.

Lu, S., Hong, M., and Wang, Z. A Nonconvex Splitting Method for Symmetric Nonnegative Matrix Factorization : Convergence Analysis and Optimality. *IEEE Transactions on Signal Processing*, 65(12):2572–2576, 2017.

Nesterov, Y. A Method of Solving A Convex Programming Problem With Convergence rate O(1/k^2), 1983.

Nesterov, Y. Gradient methods for minimizing composite objective function. *CORE Report*, 2007.

Roulet, V. and D'Aspremont, A. Sharpness, Restart and Acceleration. *NIPS*, 2017.

Vandaele, A., Gillis, N., Lei, Q., Zhong, K., and Dhillon, I. Efficient and non-convex coordinate descent for symmetric nonnegative matrix factorization. *IEEE Transactions on Signal Processing*, 64(21):5571–5584, 2016.

Vavasis, S. A. On the complexity of nonnegative matrix factorization Nonnegative matrix factorization. *SIAM Journal on Optimization*, 20(3):1364–1377, 2008.

Zhu, Z., Li, X., Liu, K., and Li, Q. Dropping Symmetry for Fast Symmetric Nonnegative Matrix Factorization. *NIPS*, 2018.

Appendix A. Convergence proof for Dyn-NoLips

In this appendix, we analyze the variant of NoLips with a dynamical step size strategy, and establish the same convergence properties as for the standard version.

Setup. Recall that the minimization problem is

$$\min_{X \in \mathbb{R}^{n \times r}} \Psi(X) \triangleq f(X) + g(X) \tag{31}$$

where $f(X) = \frac{1}{2}\|M - XX^T\|^2$ and $g(X) = I(\{X \geq 0\})$ is the indicator of the nonnegative orthant $\mathbb{R}_+^{n \times r}$.

We proved that f is relatively smooth with respect to a distance kernel $h(X) = \frac{1}{4}\|X\|^4 + \alpha\|X\|^2$ with constant $L = 1/6$ (and $\alpha = \|M\|_{1,\infty}/3$):

$$f(X) \leq f(Y) + \langle \nabla f(Y), X - Y \rangle + LD_h(X, Y) \tag{32}$$

For every $X, Y \in \mathbb{R}^{n \times r}$. We now prove the following Lemma, that establishes a crucial inequality for proving the convergence.

Lemma A.1. *Let $\{X^k\}_{k \geq 0}$ the sequence generated by Algorithm 3, and $\{\lambda_k\}_{k \geq 0}$ the corresponding sequence of step size values. Then for every $k \geq 0$,*

$$\Psi(X^{k+1}) - \Psi(X^k) \leq -\frac{1}{\lambda_k} D_h(X^k, X^{k+1}) \tag{33}$$

Proof. Since the condition (24) holds at each iteration k, we can write

$$f(X^{k+1}) \leq f(X^k) + \langle \nabla f(X^k), X^{k+1} - X^k \rangle + \frac{1}{\lambda_k} D_h(X^{k+1}, X^k) \tag{34}$$

On the other hand, the optimality condition characterizing $X^{k+1} = T_{\lambda_k}(X^k)$ writes

$$0 \in \lambda_k \left(\partial g(X^{k+1}) + \nabla f(X^k) \right) + \nabla h(X^{k+1}) - \nabla h(X^k) \tag{35}$$

Where ∂g denotes the subdifferential of the convex function g. Combining (35) with the subgradient inequality for g yields

$$g(X^{k+1}) \leq g(X^k) + \frac{1}{\lambda_k} \langle \nabla h(X^k) - \nabla h(X^{k+1}), X^{k+1} - X^k \rangle - \langle \nabla f(X^k), X^{k+1} - X^k \rangle \tag{36}$$

Summing (34) and (36) gives

$$\begin{aligned}\Psi(X^{k+1}) &\leq \Psi(X^k) + \frac{1}{\lambda_k} \left[D_h(X^{k+1}, X^k) + \langle \nabla h(X^k) - \nabla h(X^{k+1}), X^{k+1} - X^k \rangle \right] \\ &= \Psi(X^k) - \frac{1}{\lambda_k} D_h(X^k, X^{k+1})\end{aligned}$$

■

We can now prove the same convergence properties as for the standard NoLips scheme:

Theorem A.2. *Let $\{X^k\}_{k \geq 0}$ be the sequence generated by Algorithm 3.*

(1) *The sequence $\{f(X^k)\}_{k \geq 0}$ is nonincreasing.*
(2) *The sequence $\{X^k\}_{k \geq 0}$ converges towards a critical point X^* of Ψ.*
(3) *There exist $\omega > 0$ and $\theta > 0$ such that we have the convergence rate $\|X^k - X^*\| \leq Ck^{-\omega}$.*

Proof. The first point is a direct consequence of Lemma A.1 and the fact that $D_h(\cdot, \cdot)$ is nonnegative. Since $\lambda_k \leq \lambda_{\max}$ by construction of the algorithm, it follows that at every iteration $k \geq 0$,

$$\Psi(X^k) - \Psi(X^{k+1}) \geq \frac{1}{\lambda_{\max}} D_h(X^k, X^{k+1}) \tag{37}$$

Now that we have this inequality, the same arguments as in the case of the fixed step size hold; thus [Bolte et al., 2018, Th. 4.1] shows the global convergence of $\{X^k\}_{k\geq 0}$ towards a critical point and [Bolte et al., 2018, Th. 6.3] establishes the generic rate of convergence. ■

UNIVERSITÉ TOULOUSE I CAPITOLE,
TOULOUSE, FRANCE.
E-mail address: radu-alexandru.dragomir@ut-capitole.fr

TSE (UNIVERSITÉ TOULOUSE 1 CAPITOLE),
TOULOUSE, FRANCE.
E-mail address: jbolte@ut-capitole.fr

CNRS & D.I., UMR 8548,
ÉCOLE NORMALE SUPÉRIEURE, PARIS, FRANCE.
E-mail address: aspremon@ens.fr