http://pubsonline.informs.org/journal/moor/

MATHEMATICS OF OPERATIONS RESEARCH
Articles in Advance, pp. 1–13
ISSN 0364-765X (print), ISSN 1526-5471 (online)

Downloaded from informs.org by [130.113.111.210] on 14 June 2018, at 09:10 . For personal use only, all rights reserved.

On the Complexity of Robust PCA and ℓ_1-Norm Low-Rank Matrix Approximation

Nicolas Gillis,[a] Stephen A. Vavasis[b]

[a] Department of Mathematics and Operational Research, University of Mons, 7000 Mons, Belgium; [b] Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada

Contact: nicolas.gillis@umons.ac.be, http://orcid.org/0000-0001-6423-6897 (NG); vavasis@uwaterloo.ca (SAV)

Received: August 11, 2016
Revised: April 18, 2017
Accepted: July 25, 2017
Published Online in Articles in Advance: June 13, 2018

MSC2010 Subject Classification: 15A23; 90C60
OR/MS Subject Classification: Analysis of algorithms: computational complexity

https://doi.org/10.1287/moor.2017.0895

Copyright: © 2018 INFORMS

Abstract. The low-rank matrix approximation problem with respect to the component-wise ℓ_1-norm (ℓ_1-LRA), which is closely related to robust principal component analysis (PCA), has become a very popular tool in data mining and machine learning. Robust PCA aims to recover a low-rank matrix that was perturbed with sparse noise, with applications for example in foreground-background video separation. Although ℓ_1-LRA is strongly believed to be NP-hard, there is, to our knowledge, no formal proof of this fact. In this paper, we prove that ℓ_1-LRA is NP-hard, already in the rank-one case, using a reduction from MAX CUT. Our derivations draw interesting connections between ℓ_1-LRA and several other well-known problems, i.e., robust PCA, ℓ_0-LRA, binary matrix factorization, a particular densest bipartite subgraph problem, the computation of the cut norm of $\{-1, +1\}$ matrices, and the discrete basis problem, all of which we prove to be NP-hard.

Funding: The first author acknowledges the support by the F.R.S.-FNRS (incentive grant for scientific research no F.4501.16) and by the ERC (starting grant no 679515). The second author's work is supported in part by a Discovery Grant from NSERC (Natural Sciences and Engineering Research Council) of Canada and the U.S. Air Force Office of Scientific Research.

Keywords: robust PCA • low-rank matrix approximations • binary matrix factorization • cut norm • computational complexity

1. Introduction

Low-rank matrix approximation is a key problem in data analysis and machine learning. It is equivalent to linear dimensionality reduction that approximates a set of data points via a low-dimensional linear subspace. Given a matrix $M \in \mathbb{R}^{m\times n}$ and a factorization rank $r \le \min(m, n)$, the problem can be stated as follows:

$$\min_{X\in\mathbb{R}^{m\times n}} \|M - X\| \quad \text{such that} \quad \operatorname{rank}(X) \le r,$$

where $\|\cdot\|$ is a matrix norm used to measure the error of the approximation. Equivalently, the matrix X can be written as the outer product of two matrices and we have

$$\min_{U\in\mathbb{R}^{m\times r},\, V\in\mathbb{R}^{r\times n}} \|M - UV\|.$$

Typically, the columns of the matrix M represent n data points in a m-dimensional space. The above decomposition gives $M(:, j) \approx \sum_{k=1}^{r} U(:, k)V(k, j)$ for all j, and hence is a linear and low-dimensional model for the data: The columns of U are the basis of the linear subspace while each column of V gives the coordinates in the basis U to approximate each data point.

The choice of the norm $\|\cdot\|$ usually depends on the problem at hand and the noise model that is assumed on the input data. The most widely used norm is the Frobenius norm:

$$\min_{U\in\mathbb{R}^{m\times r},\, V\in\mathbb{R}^{r\times n}} \|M - UV\|_F^2 = \sum_{i,j} (M - UV)_{ij}^2, \tag{1}$$

which assumes Gaussian noise. The problem (1) can be solved via the singular value decomposition (SVD); see Golub and Van Loan [19] and the references therein. It is closely related to principal component analysis (PCA) as both problems are essentially equivalent (in PCA, the data is usually assumed to be mean centered). However, in some applications, other metrics might have to be used; here are two important examples:

- *Weights and missing data.* Adding weights in the objective function, that is, minimizing $\sum_{i,j} W_{ij}(M - UV)_{ij}^2$ for some nonnegative weight matrix $W \in \mathbb{R}_+^{m\times n}$, allows us to take into account different confidence levels among

RIGHTSLINK

Downloaded from informs.org by [130.113.111.210] on 14 June 2018, at 09:10 . For personal use only, all rights reserved.

the entries of the input data M (Gabriel and Zamir [15]), or to take into account missing entries (corresponding to zero entries of W). This has applications in machine learning for recommender systems (Koren et al. [26]), in computer vision to recover structure from motion (Shum et al. [36]), and in control for system identification (Markovsky and Usevich [28], Usevich and Markovsky [38]). However, the problem is NP-hard for any fixed factorization rank (Gillis and Glineur [17]), even in the rank-one case (that is, for $r = 1$).

• *Sparse input matrix.* If the input matrix is sparse, which is typical for example in applications involving large graphs and networks or document data sets, Gaussian noise is not a good model; it makes more sense to minimize for example the (generalized) Kullback-Leibler divergence

$$D(M\|UV) = \sum_{i,j}\left(M_{ij}\log\left(\frac{M_{ij}}{(UV)_{ij}}\right) - M_{ij} + (UV)_{ij}\right);$$

see, e.g., the discussion in Chi and Kolda [9] and the references therein.

Another important example that has been extensively studied is when the noise is sparse. In that case, the following problem is often considered:

$$\min_{X\in\mathbb{R}^{m\times n},\, S\in\mathbb{R}^{m\times n}} \operatorname{rank}(X) + \lambda\|S\|_0 \quad \text{such that} \quad M = X + S, \tag{2}$$

where $\|\cdot\|_0$ is the ℓ_0 "norm" defined as

$$\|S\|_0 = |\{(i,j) \mid S_{ij} \neq 0\}|$$

that counts the number of nonzero entries in the matrix S, and $\lambda > 0$ is a penalty parameter; see, e.g., Wright et al. [40] and the references therein. Note that the equality constraint $M = X + S$ can be replaced with $\|M - X - S\| \leq \epsilon$ in case some other type of noise is present (e.g., using the Frobenius norm allows us to model both Gaussian and sparse noise). This problem is sometimes called *robust PCA* (Candès et al. [7]). Equivalently, if the rank of X is fixed to r, the problem (2) can be written as

$$\min_{U\in\mathbb{R}^{m\times r},\, V\in\mathbb{R}^{r\times n}} \|M - UV\|_0, \tag{3}$$

which we will call ℓ_0 low-rank matrix approximation (ℓ_0-LRA).

Several heuristic algorithms have been proposed for this problem. The two main families are the following:

1. *Non-linear optimization-based algorithms*. Using the formulation (3) and replacing the ℓ_0 norm by its well known convex surrogate, the ℓ_1 norm, we have

$$\min_{U\in\mathbb{R}^{m\times r},\, V\in\mathbb{R}^{r\times n}} \|M - UV\|_1 = \sum_{i,j} |M_{ij} - (UV)_{ij}|, \tag{4}$$

which we will call ℓ_1-LRA. One can apply standard non-linear optimization schemes to (4), e.g., sequential rank-one updates (Ke and Kanade [22]), alternating optimization (Ke and Kanade [23]) (a.k.a. coordinate descent), the Wiberg algorithm (Eriksson and Van Den Hengel [12]), augmented Lagrangian approaches (Zheng et al. [43]), successive projections on hyperplanes and linear programming (Brooks et al. [5]), to cite a few. The main drawback of this class of methods is that it does not guarantee recovery of the global optimum of (4) and is in general sensitive to initialization. (In fact, we will show that this problem is NP-hard; see Theorem 3.)

2. *Convexification*. Starting from the formulation (2), the standard convexification approach is to use the ℓ_1 norm as a proxy for sparsity and the nuclear norm $\|\cdot\|_*$ as a proxy for the rank function (Candès et al. [7]). The nuclear norm $\|X\|_* = \sum_i \sigma_i(X)$ is the sum of the singular values of X. Denoting $\sigma(X)$ the vector containing the singular values of X, we have $\operatorname{rank}(X) = \|\sigma(X)\|_0$ and $\|X\|_* = \|\sigma(X)\|_1$ which explains this choice of the nuclear norm: It is the ℓ_1 norm of the vector of singular values. Finally, the difficult combinatorial problem (2) is replaced with the following SDP-representable optimization problem (hence tractable) (Chandrasekaran et al. [8]):

$$\min_{X,L} \|X\|_* + \lambda\|L\|_1 \quad \text{such that} \quad M = X + L. \tag{5}$$

Given that the input matrix M has the sought structure (that is, sparse + low-rank) and satisfies some additional conditions (e.g., the non-zero entries of the sparse noise are not too numerous and appear randomly among the entries of M), solving (5) guarantees recovery of the sought solution (Chandrasekaran et al. [8], Candès et al. [7], Xu et al. [41]). This model has attracted a lot of attention lately, for its theoretical, algorithmic, and application-oriented aspects.

RIGHTSLINK

Downloaded from informs.org by [130.113.111.210] on 14 June 2018, at 09:10 . For personal use only, all rights reserved.

The two main drawbacks of this approach are that,

(a) if the input matrix is far from being a low-rank matrix plus sparse noise, the solution of (5) can be very poor, and

(b) one must solve an optimization problem in nm variables (for example, in foreground-background video separation, n is the number of pixels and m the number of frames, which can both be high).

More recently, a simple algorithm based on alternating projections (alternatively project onto the set of low-rank matrices and sparse matrices) was proved to recover the sought solutions under reasonable conditions (Netrapalli et al. [31]) (similar to that of the convexification-based approaches).

Problems (3), (4), (5) and variants have been used for many applications, e.g., foreground-background video separation, face recognition, latent semantic indexing, graphical modeling with latent variables, matrix rigidity and composite system identification; see discussions in Candès et al. [7], Chandrasekaran et al. [8], Qiu et al. [33] and the references therein. It can also be used to identify large and dense subgraphs in bipartite graphs. Let M be the biadjacency matrix of a graph representing the relationships between two groups of objects, e.g., movies versus users, documents versus words, or papers versus authors. Let us focus on the movies versus users example: Each row of M corresponds to a movie, each column to a user, and $M_{ij} = 1$ if and only if user j has watched movie i. Finding a subset of movies and a subset of users that is fully connected (referred to as a biclique) amounts to finding a community (a group of users watching the same movies). In the unweighted case, the matrix M is binary and it can be easily checked that a community corresponds to a rank-one binary matrix (a rectangle of ones). Moreover, in practice, some edges are often missing inside a community (all users have not watched all movies from their community) or some edges between communities might be present (some users might belong to several communities or watch movies from other communities). An important problem in this setting is to find the largest community. This can be cast as a rank-one robust PCA problem (see also Section 2). We illustrate this with a simple example.

Example 1. Assume we have a single community represented by the following matrix (the first three movies have been watched by the first four users):

$$M = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 1 & 0 \end{pmatrix}.$$

In real-world problems, some sparse noise is added to the matrix M (see above). Under such perturbations, the optimal rank-one solution of ℓ_2-LRA (1) loses the underlying structure very quickly, even when only a few entries in M are modified. For example, adding three edges to M to obtain $\tilde{M}$ gives the following optimal rank-one approximation $u^* v^{*T}$ for ℓ_2-LRA:

$$\tilde{M} = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix} \approx u^* v^{*T} = \begin{pmatrix} 1.03 & 0.92 & 0.92 & 0.92 & 0.44 \\ 1.15 & 1.02 & 1.02 & 1.02 & 0.50 \\ 1.03 & 0.92 & 0.92 & 0.92 & 0.44 \\ 0.40 & 0.36 & 0.36 & 0.36 & 0.17 \end{pmatrix}.$$

Conversely, the optimal rank-one solutions[1] of ℓ_0-LRA (3) and ℓ_1-LRA (4) for $\tilde{M}$ are the same and given by the original unperturbed M. Ames and Vavasis [2], and Doan and Vavasis [11] studied this particular variant of robust PCA (although they did not call it that) and showed that the convexification approach based on the nuclear norm can recover the largest community given that sufficiently few edges are perturbed (randomly or by an adversary).

Another closely related class of low-rank matrix approximation problems has also attracted much attention lately, i.e.,

$$\min_{X,\,\mathrm{rank}(X)=r} \sum_{i=1}^{n} (\|M(:,i) - X(:,i)\|_2)^p.$$

For $p = 2$, this is ℓ_2-LRA (1). For $1 \le p \ne 2$, the problem has been shown to be NP-hard (Guruswami et al. [20], Clarkson and Woodruff [10]), and approximation algorithms have been proposed; see Clarkson and Woodruff [10] and the references therein.

RIGHTSLINK

Downloaded from informs.org by [130.113.111.210] on 14 June 2018, at 09:10 . For personal use only, all rights reserved.

1.1. Contribution and Outline of the Paper

Although robust PCA and its variants are widely believed to be NP-hard (see, e.g., Kwak [27], Wright et al. [40]), it has, to our knowledge, never been rigorously proved. In this paper, we prove that ℓ_0-LRA and ℓ_1-LRA are NP-hard, already in the rank-one case, that is, for $r = 1$. This solves the first part of the open question 2 in Woodruff [39].

In Section 2, we focus on ℓ_0-LRA (3) of a binary matrix, which we show is equivalent to rank-one binary matrix factorization (BMF). We also show the connection with the problem of finding a large and dense subgraph in a bipartite graph. In Section 3, we prove that rank-one BMF is equivalent to the cut norm computation of $\{-1,+1\}$ matrices, which we prove to be NP-hard using an equivalence with the computation of the norm $\|\cdot\|_{\infty\to 1}$ and a reduction from MAX CUT (Theorem 1). This implies that ℓ_0-LRA and rank-one BMF are both NP-hard. In Section 4, we prove that, for a $\{-1,+1\}$ input matrix, any optimal solution of rank-one ℓ_1-LRA (4) can be transformed into a rank-one solution with entries in $\{-1,+1\}$ (Theorem 2). We also show that, for $\{-1,+1\}$ matrices, rank-one ℓ_1-LRA is equivalent to the computation of the norm $\|\cdot\|_{\infty\to 1}$ which implies NP-hardness of ℓ_1-LRA (Theorem 3). In Section 5, we briefly describe how the complexity results in the rank-one case can be generalized to higher ranks.

2. Binary Matrix Factorization and Densest Bipartite Subgraph

Let $M \in \{0,1\}^{m\times n}$ be a binary matrix. Rank-one BMF is the problem

$$\min_{u\in\{0,1\}^m,\, v\in\{0,1\}^n} \|M - uv^T\|. \tag{6}$$

BMF was successfully used to mine discrete patterns with applications for example to analyze gene expression data (Shen et al. [35], Zhang et al. [42], Mirisaee et al. [30]). See the tutorial http://people.mpi-inf.mpg.de/~pmiettin/bmf_tutorial and the references therein for more details. Although BMF is conjectured to be NP-hard (Shen et al. [35], Mirisaee et al. [30]), there is, to our knowledge, no formal proof of this fact. We will prove in this paper that it is in fact NP-hard.

For BMF, all component-wise norms, that is, all norms of the form $\|M - uv^T\| = \sum_{i,j} f(M_{i,j}, u_i v_j)$ for some function f with $f(z,z) = 0$ and $f(z,z') > 0$ for $z' \neq z$, are equivalent since both M and uv^T are binary matrices. For such norms, $\|M - uv^T\|$ amounts to count the number of mismatches between M and uv^T and hence rank-one BMF can be formulated as follows:

$$\min_{u\in\{0,1\}^m,\, v\in\{0,1\}^n} \|M - uv^T\|_0. \tag{7}$$

The matrix $M \in \{0,1\}^{m\times n}$ can be interpreted as the biadjacency of a bipartite graph $G = (S\times T, E)$ with $S = \{s_1, s_2, \ldots, s_m\}$, $T = \{t_1, t_2, \ldots, t_n\}$, and $E \subset S\times T$ where $M_{ij} = 1 \iff (s_i, t_j) \in E$. Let us denote $E(S', T')$ the number of edges in G in the subgraph induced by $S'\times T'$, and denote $\bar{S}' = S\backslash S'$. Then (7) is the problem of finding two subsets $S' \subseteq S$ and $T' \subseteq T$ (with $s_i \in S' \iff u_i = 1$ and $t_j \in V_1' \iff v_j = 1$), such that the subsets of vertices S' and T' maximize the following quantity

$$\underbrace{E(S',T')}_{\#\text{ edges in } S'\times T'} - \underbrace{(|S'||T'| - E(S',T'))}_{\#\text{ non-edges in } S'\times T'} - \underbrace{(|E| - E(S',T'))}_{\#\text{ edges outside } S'\times T'} = 3E(S',T') - |S'||T'| - |E|.$$

This problem is a particular variant of the general problem of finding large dense subgraphs in bipartite graphs; see, e.g., Asahiro et al. [3], Khot [24], Khuller and Saha [25] and the references therein. If the size of the subgraph is fixed a priori, finding the densest subgraph is NP-hard (Asahiro et al. [3]), even to approximate (Khot [24]). However, finding a partition $S'\times T'$ that maximizes $E(S',T')/\sqrt{|S'||T'|}$ can be done in polynomial time (Kannan and Vinay [21]). The problem above is slightly different because we do not fix the size nor try to find the densest subgraph: It looks for a subgraph that is at the same time large and relatively dense. Hence, as far as we know, the complexity results for the densest subgraph do not apply to our problem (at least we could not find a reduction from these problems to ours).

In the following we prove that rank-one ℓ_0-LRA is equivalent to rank-one BMF. Let us show the following straightforward lemma.

Lemma 1. *Let $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$, and let M be a binary matrix. Applying the following simple transformation to x and y*

$$\Phi(x)_i = \begin{cases} 0 & \text{if } x_i = 0, \\ 1 & \text{otherwise,} \end{cases}$$

RIGHTSLINK

Downloaded from informs.org by [130.113.111.210] on 14 June 2018, at 09:10 . For personal use only, all rights reserved.

gives

$$\|M - \Phi(x)\Phi(y)^T\|_0 \le \|M - xy^T\|_0.$$

Proof. There are two cases

1. If $x_i y_j = 0$, then $\Phi(x_i)\Phi(y_j) = 0$ hence the transformation does not affect the approximation.
2. If $x_i y_j \neq 0$, then $\Phi(x_i)\Phi(y_j) = 1$. If $M_{ij} = 0$ then $\|M_{ij} - x_i y_j\|_0 = \|M - \Phi(x_i)\Phi(y_j)\|_0 = 1$ while, if $M_{ij} = 1$, $\|M_{ij} - x_i y_j\|_0 \ge \|M - \Phi(x_i)\Phi(y_j)\|_0 = 0$. □

Corollary 1. *For a binary input matrix M, rank-one ℓ_0-LRA is equivalent to rank-one BMF.*

In the next section, we prove that rank-one ℓ_0-LRA of a binary matrix is NP-hard, showing it is equivalent to the computation of the cut norm of a $\{-1,+1\}$ matrix which we show is NP-hard using a reduction from MAX CUT.

3. Cut Norm of $\{-1,+1\}$ Matrices and Rank-One ℓ_0-LRA

Given a matrix M, its cut norm is defined as Frieze and Kannan [14]

$$\|M\|_C = \max_{u\in\{0,1\}^m,\, v\in\{0,1\}^n} |u^T M v|. \tag{8}$$

The fact that this is a norm on $\mathbb{R}^{m\times n}$ (regarded as a vector space isomorphic to $\mathbb{R}^{mn}$) can be easily checked. In Frieze and Kannan [14], Frieze and Kannan study the low-rank matrix approximation problem with respect to the cut norm and design an algorithm that provides a solution which is the sum of $O(1/\epsilon^2)$ rank-one matrices with error at most ϵmn.

Alon and Naor [1] proved NP-hardness of the problem of computing the cut norm using a reduction from MAX CUT. The reduction uses matrices M with entries in $\{-1,0,+1\}$.

Let us show how the cut norm computation is related to low-rank matrix approximations. Let M be a binary matrix and let us derive some equivalent forms of rank-one BMF or, equivalently, to rank-one ℓ_0-LRA (Corollary 1). First note that for any (u,v), we have

$$\|M - uv^T\|_F^2 = \|M\|_F^2 - 2\sum_{i,j} M_{ij}u_i v_j + \sum_{i,j}(u_i v_j)^2. \tag{9}$$

Hence, we obtain

$$\begin{aligned}\min_{u\in\{0,1\}^m,\, v\in\{0,1\}^n} \|M - uv^T\|_F^2 &= \|M\|_F^2 + \min_{u\in\{0,1\}^m,\, v\in\{0,1\}^n} \sum_{i,j} u_i v_j - 2\sum_{i,j} M_{ij}u_i v_j \\ &= \|M\|_F^2 + \min_{u\in\{0,1\}^m,\, v\in\{0,1\}^n} \sum_{i,j}(1-2M_{ij})u_i v_j \\ &= \|M\|_F^2 + \max_{u\in\{0,1\}^m,\, v\in\{0,1\}^n} \sum_{i,j}(2M_{ij}-1)u_i v_j.\end{aligned}$$

The last problem is closely related to the cut norm of the $\{-1,+1\}$ matrix $A = 2M - 1$ and $-A$. In fact,

$$\|A\|_C = \max\left(\max_{u\in\{0,1\}^m,\, v\in\{0,1\}^n} \sum_{i,j}(2M_{ij}-1)u_i v_j,\ \max_{u\in\{0,1\}^m,\, v\in\{0,1\}^n} \sum_{i,j}(1-2M_{ij})u_i v_j\right).$$

Hence, if we could solve $\max_{u\in\{0,1\}^m,\, v\in\{0,1\}^n} u^T X v$ for any $X \in \{-1,+1\}^{m\times n}$, we could compute the cut norm of A. However, the cut norm problem was shown to be NP-hard only for $A \in \{-1,0,+1\}^{m\times n}$ using a reduction from MAX CUT. It turns out that the reduction no longer holds for $A \in \{-1,+1\}^{m\times n}$.

In the following, we prove that computing the cut norm of $\{-1,+1\}^{m\times n}$ matrices is NP-hard hence rank-one BMF and rank-one ℓ_0-LRA are also NP-hard.

First, let us consider the following norm introduced in Alon and Naor [1]: For $A \in \mathbb{R}^{m\times n}$

$$\|A\|_{\infty\to 1} = \max_{u\in\{-1,+1\}^m,\, v\in\{-1,+1\}^n} u^T A v. \tag{10}$$

To our knowledge, solving (10) was first shown to be NP-hard in Poljak and Rohn [32] (for matrices in $\{0,1,p\}$ for some positive real p); see also Rohn [34].

This norm is closely related to the cut norm; in fact, it can be easily shown that Alon and Naor [1]

$$\|A\|_C \le \|A\|_{\infty\to 1} \le 4\|A\|_C.$$

Moreover,

RIGHTSLINK

Downloaded from informs.org by [130.113.111.210] on 14 June 2018, at 09:10 . For personal use only, all rights reserved.

Lemma 2. *Given $A \in \mathbb{R}^{m\times n}$, we have*

$$\|A\|_{\infty\to 1} = \left\| \begin{pmatrix} A & -A \\ -A & A \end{pmatrix} \right\|_C.$$

Proof. Alon and Naor [1] proved that a matrix B whose rows and columns sum to zero satisfies

$$\|B\|_{\infty\to 1} = 4\|B\|_C.$$

In fact, let e be the vector of all ones, and $u = 2x - e$ and $v = 2y - e$ have entries in $\{-1, +1\}$ where x and y have binary entries. Then,

$$u^T Bv = (2x-e)^T B(2y-e) = 4x^T By - 2x^T Be - 2e^T By + e^T Be = 4x^T By$$

since $Be = 0$ and $B^T e = 0$ by assumption.

Since the rows and columns of the matrix

$$B = \begin{pmatrix} A & -A \\ -A & A \end{pmatrix}$$

sum to zero, we have $\|B\|_{\infty\to 1} = 4\|B\|_C$. To conclude the proof, we show that $\|B\|_{\infty\to 1} = 4\|A\|_{\infty\to 1}$. Let $u = [u_1; u_2]$ and $v = [v_1; v_2]$ be a solution of (10) for B. We have

$$\|B\|_{\infty\to 1} = u^T Bv = u_1^T Av_1 - u_1^T Av_2 + u_2^T Av_2 - u_2^T Av_1.$$

Clearly, each term must be smaller than $\|A\|_{\infty\to 1}$ since u and v have entries in $\{-1, +1\}$ and $\|A\|_{\infty\to 1} = \|-A\|_{\infty\to 1}$. However, taking u_1 and v_1 such that

$$\|A\|_{\infty\to 1} = u_1^T Av_1,$$

and $u_2 = -u_1$, $v_2 = -v_1$ gives $\|B\|_{\infty\to 1} = 4\|A\|_{\infty\to 1}$. □

Let us show that computing the norm $\|\cdot\|_{\infty\to 1}$ is NP-hard for $\{-1, +1\}$ matrices, which will imply, by Lemma 2, that it is NP-hard to compute the cut norm of $\{-1, +1\}$ matrices. The reduction in our proof is similar to that of Alon and Naor [1] except that we replace the zeros in A by Hadamard matrices, while ones (resp. minus ones) are replaced with matrices containing all ones (resp. minus ones).

Theorem 1. *It is NP-hard to compute the norm $\|\cdot\|_{\infty\to 1}$ for $\{-1, +1\}$-matrices.*

Proof. The problem under consideration is, given an $m \times n$ matrix A all of whose entries are ± 1, find $u \in \{-1, 1\}^m$ and $v \in \{-1, 1\}^n$ to maximize $u^T Av$. Let us define P1 the decision version of this problem:

(P1) Given A and an integer d^*, does there exist $u \in \{-1, 1\}^m$ and $v \in \{-1, 1\}^n$ such that $u^T Av \geq d^*$?

We prove that this problem is NP-hard by a reduction from unweighted MAX CUT.

(MAX CUT) Given a graph $G = (V, E)$ and a positive integer c^*, is there a cut containing at least c^* edges?

Given an instance (G, c^*) of MAX CUT, let us produce an instance (A, d^*) of P1 as follows: Let $p \geq 1$ be an integer to be determined later. The matrix A has dimension $m \times n$ with $m = p|E|$ and $n = p|V|$. It will be constructed via $p \times p$ blocks as follows. Suppose edge $q \in E$, $1 \leq q \leq |E|$, has endpoints (i, j), $1 \leq i < j \leq |V|$. Then the (q, i) block of A is the $p \times p$ block of all 1's; the (q, j) block of A is the $p \times p$ block of all -1's, and the (q, l) block of A for any $l \in \{1, \ldots, |V|\} - \{i, j\}$ is the $p \times p$ Hadamard matrix H.

Recall that a $p \times p$ *Hadamard matrix* H is a matrix all of whose entries are ± 1 and such that the columns are mutually orthogonal. In the case that p is a power of 2, there is a straightforward recursive construction of a Hadamard matrix: The 1×1 Hadamard matrix is $[1]$, and the $p \times p$ Hadamard matrix is $[H_0, H_0; -H_0, H_0]$, where H_0 is the $(p/2) \times (p/2)$ Hadamard matrix. For the remainder of this proof, assume that p is a power of 2 so that we can rely on this simple construction.

Suppose $(S, \bar{S})$ is a partition of V (i.e., a cut). We can associate vectors (u, v) with S as follows: For $i \in S$, let the ith block of v contain p 1's. For $i \in \bar{S}$, let the ith block of v contain p -1's. For $q = (i, j) \in E$, $i < j$, such that $i \in S$, $j \in \bar{S}$, let the qth block of u contain p 1's. For $(i, j) \in E$, $i < j$, such that $i \in \bar{S}$, $j \in S$, let the qth block of u contain p -1's. Finally, for $q = (i, j)$ such that $\{i, j\} \subset S$ or $\{i, j\} \subset \bar{S}$, the qth block of u may be selected arbitrarily.

For the choice of (u, v) in the last paragraph, let us obtain a lower bound on $u^T Av$. Observe that the signs have been chosen such that for each $q = (i, j)$ that crosses the cut, the blocks of A indexed (q, i) and (q, j) contribute $2p^2$ to $u^T Av$. Blocks of A indexed (q, i) and (q, j) such that $q = (i, j)$ does not cross the cut contribute 0. Finally,

RIGHTSLINK

Downloaded from informs.org by [130.113.111.210] on 14 June 2018, at 09:10 . For personal use only, all rights reserved.

we must account for the blocks of the form (q, l) where l is not an endpoint of q. For this, we make the following observation: If u_0, v_0 are two ± 1 p-dimensional vectors and H is a $p \times p$ Hadamard matrix, then $|u_0^T H v_0| \le p^{3/2}$ (Brown and Spencer [6]). This follows because $\|u_0\|_2 = \|v_0\|_2 = \|H\|_2 = \sqrt{p}$. (The last equation follows because the ℓ_2-norm of an orthogonal matrix is exactly 1, and a Hadamard matrix is an orthogonal matrix scaled by $\sqrt{p}$.)

Thus, a lower bound on the objective $u^T A v$ for the cut $(S, \bar{S})$ is $2p^2 c - |E||V|p^{3/2}$, where c is the size of the cut induced by $(S, \bar{S})$ because there are at most $|E||V|$ blocks of the form (q, l) where l is not an endpoint of q. So the decision problem posed for P1 is: Given A constructed above, is the objective function for this A at least $d^* = 2p^2 c^* - |E||V|p^{3/2}$? We have already shown that if there is a cut of size c^* in the graph, then there is a solution of size d^* for P1.

The last thing to prove is that if the max cut of G has fewer than c^* edges, then P1 is a no-instance, i.e., for any $u \in \{-1, 1\}^{|E|p}$ and $v \in \{-1, 1\}^{|V|p}$, $u^T A v < d^*$. Choose any $u \in \{-1, 1\}^{|E|p}$ and $v \in \{-1, 1\}^{|V|p}$. Let s_q denote the number of 1's in block q of u, $q = 1, \ldots |E|$ (so that $p - s_q$ is the number of -1's in the block). Let t_i denote the number of 1's in block i of v. It is straightforward to show that contribution to the objective function from the $2|E|$ blocks of A that correspond to edges (i.e., the blocks numbered (q, i) where i is an endpoint of q) is precisely T_1, where

$$T_1 = \sum_{q=(i,j)\in E} 2(t_i - t_j)(2s_q - p).$$

Now observe that

$$T_1 \le \sum_{q=(i,j)\in E} 2|t_i - t_j|p$$

since the second factor in the previous summation has absolute value at most p. Next, note that this latter summation, regarded as a function of $t \in [0, p]^{|V|}$, is maximized at an extreme point because it is a convex function. Therefore, there exists a vector $\tilde{t} \in \{0, p\}^{|V|}$ such that

$$T_1 \le 2p \sum_{q=(i,j)\in E} |\tilde{t}_i - \tilde{t}_j|.$$

The latter is exactly $2p^2$ multiplied by the size of cut induced by $\tilde{t}$ (i.e., the cut $(S, \bar{S})$ with $i \in S$ if and only if $\tilde{t}_i = p$). Since we are considering the case that all cuts have fewer than c^* edges,

$$T_1 \le 2p^2(c^* - 1).$$

This accounts for the $2|E|$ blocks that correspond to edges. For the $|E||V| - 2|E|$ blocks that do not correspond to edges, the contribution to the objective function is at most $p^{3/2}$ for the same reason as above. Therefore,

$$u^T A v \le 2p^2(c^* - 1) + |E||V|p^{3/2} = 2p^2 c^* - (2p^2 - |E||V|p^{3/2}).$$

We see that the right-hand side (RHS) is less than $d^* = 2p^2 c^* - |E||V|p^{3/2}$ provided that $2|E||V|p^{3/2} < 2p^2$, i.e., $\sqrt{p} > |E||V|$. Therefore, we choose $p > |E|^2|V|^2$ and also p a power of 2. □

Corollary 2. *It is NP-hard to compute the cut norm* (8) *of* $\{-1, +1\}$*-matrices.*

Corollary 3. *Rank-one* ℓ_0*-LRA, that is, problem* (3) *with* $r = 1$*, is NP-hard.*

Corollary 4. *Rank-one BMF* (6) *is NP-hard.*

Remark 1 (Discrete Basis Problem and Boolean Matrix Factorization)**.** The discrete basis problem (DBP) (Miettinen et al. [29]), also known as Boolean matrix factorization, is similar to BMF and can be formulated a follows: Given a binary matrix $M \in \{0, 1\}^{m\times n}$ and a rank r, solve

$$\min_{U\in\{0,1\}^{m\times r},\, V\in\{0,1\}^{r\times n}} \|M - U \circ V\| \tag{11}$$

where $(U \circ V)_{ij} = \bigoplus_k U_{ik} V_{kj}$ where $0 \oplus 0 = 0$, $0 \oplus 1 = 1$ and $1 \oplus 1 = 1$. For $r = 1$, BMF and DBP coincide, therefore our result also implies that rank-one DBP is NP-hard.

Note that DBP is closely related to the rectangle covering problem which is equivalent to the minimum biclique cover problem in bipartite graph; see Fiorini et al. [13].

RIGHTSLINK

Downloaded from informs.org by [130.113.111.210] on 14 June 2018, at 09:10 . For personal use only, all rights reserved.

4. The Component-Wise ℓ_1 Low-Rank Matrix Approximation Problem

In this section, we prove that for a $\{-1,+1\}$ matrix, any optimal solution of rank-one ℓ_1-LRA (4) can be assumed to have entries in $\{-1,+1\}$ (Theorem 2). Moreover, we prove that computing the norm $\|\cdot\|_{\infty\to 1}$ of a $\{-1,+1\}$ matrix is equivalent to solving rank-one ℓ_1-LRA for that matrix (Lemma 3). This will imply that rank-one ℓ_1-LRA is NP-hard (Theorem 3).

Lemma 3. *For $A \in \{-1,+1\}^{m\times n}$, computing $\|A\|_{\infty\to 1}$ is equivalent to solving*

$$\min_{x\in\{-1,+1\}^m,\, y\in\{-1,+1\}^n} \|A - xy^T\|_1.$$

Proof. We have

$$\begin{aligned} 2 \min_{x\in\{-1,+1\}^m,\, y\in\{-1,+1\}^n} \|A - xy^T\|_1 &= \min_{x\in\{-1,+1\}^m,\, y\in\{-1,+1\}^n} \|A - xy^T\|_F^2 \\ &= \|A\|_F^2 + \min_{x\in\{-1,+1\}^m,\, y\in\{-1,+1\}^n} -2x^TAy + \|xy^T\|_F^2 \\ &= \|A\|_F^2 + mn + 2\|A\|_{\infty\to 1}. \end{aligned}$$

The first equality follows from the fact that $A - uv^T$ has entries in $\{-2,0,+2\}$, the second from (9), and the third by definition of $\|A\|_{\infty\to 1}$ and since xy^T has entries in $\{-1,+1\}$. □

Theorem 2. *Let A be a $\{-1,+1\}$ matrix, then any optimal solution of rank-one ℓ_1-LRA (4) for input matrix A can be transformed into an optimal solution whose entries are in $\{-1,+1\}$. This implies that*

$$\min_{x\in\{-1,+1\}^m,\, y\in\{-1,+1\}^n} \|A - xy^T\|_1 = \min_{x\in\mathbb{R}^m,\, y\in\mathbb{R}^n} \|A - xy^T\|_1.$$

Proof. Let (u,v) be an optimal solution of rank-one ℓ_1-LRA for matrix A, that is, of

$$\min_{x\in\mathbb{R}^m,\, y\in\mathbb{R}^n} \|A - xy^T\|_1 = \sum_{i,j} |A_{ij} - u_i v_j|.$$

First note that $u \neq 0$ and $v \neq 0$ since we can approximate exactly at least one entry of A, e.g., take $u_1 = 1$, $v_1 = A_{11}$ and all other entries of u and v equal to zero.

By optimality, we have for all $1 \le i \le m$ that

$$u_i = \arg\min_{x_i\in\mathbb{R}} \sum_j |A_{ij} - x_i v_j| = \arg\min_{x_i\in\mathbb{R}} \sum_{j,\, v_j\neq 0} |A_{ij} - x_i v_j|.$$

The objective function of this problem is piece-wise linear hence we can assume without loss of generality (w.l.o.g.) that u_i is equal to one of the break points (otherwise we can easily modify u so that this property holds), that is, there exists j such that $v_j \neq 0$ and

$$u_i = \frac{A_{ij}}{v_j} \in \left\{\frac{1}{v_j}, \frac{-1}{v_j}\right\}.$$

By symmetry, the same holds of v. Note that this implies that we can assume w.l.o.g. that u and v have all their entries different from zero.

Let the entries of u take k different values in absolute value $0 < \alpha_1 < \alpha_2 < \cdots < \alpha_k$. Note that the entries of v also take k different values in absolute value, i.e., $1/\alpha_p$ for $1 \le p \le k$.

We partition the solution (u,v) into k blocks (K_p^u, K_p^v) for $1 \le p \le k$ such that for all $i \in K_p^u$ and $j \in K_p^v$

$$u_i = \pm\alpha_p \qquad \text{and} \qquad v_j = \pm\frac{1}{\alpha_p}.$$

Let us assume, w.l.o.g., that

- u and v are scaled such that $\max_i |u_i| = 1$ hence $\min_j |v_j| = 1$. This implies $\alpha_k = 1$.
- we permute the entries of u (and the corresponding rows of A) such that the entries of $|u|$ are in nondecreasing order, and we permute the entries of v (and the corresponding columns of A) such that the entries of $|v|$ are in nonincreasing order.

RIGHTSLINK

Downloaded from informs.org by [130.113.111.210] on 14 June 2018, at 09:10 . For personal use only, all rights reserved.

Therefore, after suitable permutations and scaling, the rank-one solution can be put, w.l.o.g., in the following form

$$u = (\pm\alpha_1, \ldots, \pm\alpha_1, \pm\alpha_2, \ldots, \pm\alpha_2, \ldots, \pm\alpha_{k-1}, \ldots, \pm\alpha_{k-1}, \pm 1, \ldots, \pm 1)^T,$$

and

$$v = (\pm\alpha_1^{-1}, \ldots, \pm\alpha_1^{-1}, \pm\alpha_2^{-1}, \ldots, \pm\alpha_2^{-1}, \ldots, \pm\alpha_{k-1}^{-1}, \ldots, \pm\alpha_{k-1}^{-1}, \pm 1, \ldots, \pm 1)^T,$$

which gives the following rank-one solution:

	$\pm\alpha_1^{-1}$	$\cdots$	$\pm\alpha_1^{-1}$	$\cdots$	$\pm\alpha_{k-1}^{-1}$	$\cdots$	$\pm\alpha_{k-1}^{-1}$	± 1	$\cdots$	± 1
$\pm\alpha_1$										
$\vdots$		$\pm\mathbf{1}$		$\ldots$		$\pm\frac{\alpha_1}{\alpha_{k-1}}\mathbf{1}$			$\pm\alpha_1\mathbf{1}$	
$\pm\alpha_1$										
$\vdots$		$\vdots$				$\vdots$			$\vdots$	
$\pm\alpha_{k-1}$										
$\vdots$		$\pm\frac{\alpha_{k-1}}{\alpha_1}\mathbf{1}$		$\ldots$		$\pm\mathbf{1}$			$\pm\alpha_{k-1}\mathbf{1}$	
$\pm\alpha_{k-1}$										
± 1										
$\vdots$		$\pm\frac{1}{\alpha_1}\mathbf{1}$		$\ldots$		$\pm\frac{1}{\alpha_{k-1}}\mathbf{1}$			$\pm\mathbf{1}$	
± 1										

where $\mathbf{1}$ is the matrix of all ones of appropriate dimensions. The rank-one matrix uv^T is a block matrix with k^2 blocks at positions (p, q) for $1 \le p, q \le k$. Note that the diagonal blocks are not necessarily square, that is, $|u|$ may contain more (or less) entries equal to α_p than $|v|$ contains entries equal to α_p^{-1}. Note also that the blocks above (resp. below) the diagonal have entries strictly smaller (resp. larger) than one in absolute value since $\alpha_1 < \alpha_2 < \cdots < \alpha_k = 1$.

Now, let us consider two modifications of this rank-one solution and see how the objective function of rank-one ℓ_1-LRA (4) changes with these modifications.

***Move* 1.** We divide the entries in u different from ± 1 by α_{k-1} and multiply the entries in v different from ± 1 by α_{k-1} from which we get the following solution (u', v'):

$$u' = \left(\pm\frac{\alpha_1}{\alpha_{k-1}}, \ldots, \pm\frac{\alpha_1}{\alpha_{k-1}}, \pm\frac{\alpha_2}{\alpha_{k-1}}, \ldots, \pm\frac{\alpha_2}{\alpha_{k-1}}, \ldots, \frac{\alpha_{k-2}}{\alpha_{k-1}}, \ldots, \frac{\alpha_{k-2}}{\alpha_{k-1}}, \pm 1, \ldots, \pm 1\right)^T,$$

and

$$v' = \left(\pm\frac{\alpha_{k-1}}{\alpha_1}, \ldots, \pm\frac{\alpha_{k-1}}{\alpha_1}, \pm\frac{\alpha_{k-1}}{\alpha_2}, \ldots, \pm\frac{\alpha_{k-1}}{\alpha_2}, \ldots, \pm\frac{\alpha_{k-1}}{\alpha_{k-2}}, \ldots, \pm\frac{\alpha_{k-1}}{\alpha_{k-2}}, \pm 1, \ldots, \pm 1\right)^T.$$

Compared to uv^T, only the blocks at position (p, k) and (k, p) for $1 \le p \le k-1$ have been modified by the changes from (u, v) to (u', v'). Let us denote $\delta_p^{(1)}$ the modification of the objective function for the two blocks (p, k) and (k, p) (the 1 stands for move 1), that is,

$$\delta_p^{(1)} = \sum_{(i,j) \text{ in } (p,k),(k,p)\text{-blocks}} (|M_{ij} - u_i' v_j'| - |M_{ij} - u_i v_j|).$$

Let us denote a_p (resp. b_p) the number of entries in the (p, k)-block such that the sign of A does (resp. does not) match the sign in uv^T. Let us also denote c_p (resp. d_p) the number of entries in the block (k, p) such that the sign of A and the sign of uv^T does (resp. does not) match. To simplify notations, let us denote $\beta = \alpha_{k-1}$ where $1 > \beta \ge \alpha_p$ for all $1 \le p \le k-1$ by construction.

In the block (p, k) $(1 \le p \le k-1)$ the error of the solution uv^T is given by

$$e_{(p,k)} = a_p(1 - \alpha_p) + b_p(1 + \alpha_p),$$

RIGHTSLINK

Downloaded from informs.org by [130.113.111.210] on 14 June 2018, at 09:10 . For personal use only, all rights reserved.

while the error of the solution $u'v'^T$ is given by

$$e'_{(p,k)} = a_p\left(1-\frac{\alpha_p}{\beta}\right) + b_p\left(1+\frac{\alpha_p}{\beta}\right).$$

The difference is given by

$$e'_{(p,k)} - e_{(p,k)} = -\alpha_p\left(\frac{1}{\beta}-1\right)a_p + \alpha_p\left(\frac{1}{\beta}-1\right)b_p = \frac{\alpha_p(1-\beta)}{\beta}(-a_p+b_p).$$

Note that $1/\alpha_p > 1$ and $\alpha_p/\beta \le 1$ for all $1 \le p \le k-1$. Doing exactly the same for the block (k,p), we obtain the error of the solution uv^T,

$$e_{(k,p)} = c_p\left(\frac{1}{\alpha_p}-1\right) + d_p\left(\frac{1}{\alpha_p}+1\right),$$

and the error of the solution $u'v'^T$,

$$e'_{(k,p)} = c_p\left(\frac{\beta}{\alpha_p}-1\right) + d_p\left(\frac{\beta}{\alpha_p}+1\right),$$

so that

$$e'_{(k,p)} - e_{(k,p)} = -\frac{1}{\alpha_p}(1-\beta)c_p - \frac{1}{\alpha_p}(1-\beta)d_p = -\frac{1-\beta}{\alpha_p}(c_p+d_p).$$

Finally, the total difference between the objective function of (u,v) and (u',v') is given by

$$\delta^{(1)} = \|A-u'v'^T\|_1 - \|A-uv^T\|_1 = \sum_{p=1}^{k-1}\delta_p^{(1)} = (1-\beta)\sum_{p=1}^{k-1}\left(-\frac{\alpha_p}{\beta}a_p + \frac{\alpha_p}{\beta}b_p - \frac{1}{\alpha_p}c_p - \frac{1}{\alpha_p}d_p\right),$$

and is nonnegative since (u,v) is an optimal solution.

***Move* 2.** We divide the entries in u different from ± 1 by $-\alpha_{k-1}$ to get u'' and multiply the entries in v different from ± 1 by $-\alpha_{k-1}$ to get v''. Again, only the blocks at position (k,p) and (p,k) for $1 \le p \le k-1$ are affected by these modifications and we can, following exactly the same procedure as for the first move, compute the difference between the objective function value of (u'',v'') and (u,v). The error of the solution $u''v''^T$ for the block (p,k) is given by

$$e''_{(p,k)} = a_p\left(1+\frac{\alpha_p}{\beta}\right) + b_p\left(1-\frac{\alpha_p}{\beta}\right), \quad \text{so that} \quad e''_{(p,k)} - e_{(p,k)} = \alpha_p\left(\frac{1}{\beta}+1\right)a_p - \alpha_p\left(\frac{1}{\beta}+1\right)b_p.$$

For the block (k,p), we obtain the error of the solution $u''v''^T$,

$$e''_{(k,p)} = c_p\left(\frac{\beta}{\alpha_p}+1\right) + d_p\left(\frac{\beta}{\alpha_p}-1\right), \quad \text{so that} \quad e''_{(k,p)} - e_{(k,p)} = c_p\left(2+\frac{\beta}{\alpha_p}-\frac{1}{\alpha_p}\right) + d_p\left(\frac{\beta}{\alpha_p}-2-\frac{1}{\alpha_p}\right).$$

Finally,

$$\delta^{(2)} = \sum_{p=1}^{k-1}\left(\alpha_p\left(\frac{1+\beta}{\beta}\right)a_p - \alpha_p\left(\frac{1+\beta}{\beta}\right)b_p + \left(2+\frac{\beta}{\alpha_p}-\frac{1}{\alpha_p}\right)c_p - \left(2+\frac{1}{\alpha_p}-\frac{\beta}{\alpha_p}\right)d_p\right).$$

***Combining Move* 1 *and Move* 2.** Now, recall that, by optimality of (u,v), $\delta^{(1)} \ge 0$ and $\delta^{(2)} \ge 0$. Let us compute the following nonnegative linear combination of $\delta^{(1)}$ and $\delta^{(2)}$:

$$\begin{aligned}
0 \le \frac{1-\beta}{1+\beta}\delta^{(1)} + \delta^{(2)} &= \sum_{p=1}^{k-1}\left[\left(-\frac{\alpha_p(\beta+1)}{\beta} + \frac{\alpha_p(\beta+1)}{\beta}\right)a_p + \left(\frac{\alpha_p(\beta+1)}{\beta} - \frac{\alpha_p(1+\beta)}{\beta}\right)b_p\right.\\
&\quad \left. + \left(\frac{-(1+\beta)}{\alpha_p} + 2 + \frac{\beta}{\alpha_p} - \frac{1}{\alpha_p}\right)c_p + \left(\frac{-(1+\beta)}{\alpha_p} - 2 - \frac{1}{\alpha_p} + \frac{\beta}{\alpha_p}\right)d_p\right]\\
&= -\sum_{p=1}^{k-1}\left(\frac{2(1-\alpha_p)}{\alpha_p}c_p + \frac{2(1+\alpha_p)}{\alpha_p}d_p\right).
\end{aligned}$$

For all $1 \le p \le k-1$, the coefficients for c_p and d_p are positive, which implies that $c_p = d_p = 0$ for all $1 \le p \le k-1$. In other words, all the (k,p)-blocks for $1 \le p \le k-1$ are empty: This is only possible if v has entries only in $\{-1,+1\}$ hence u also has all its entries in $\{-1,+1\}$. □

RIGHTSLINK

Downloaded from informs.org by [130.113.111.210] on 14 June 2018, at 09:10 . For personal use only, all rights reserved.

Theorem 3. *Rank-one ℓ_1-LRA (4) is NP-hard.*

Proof. This follows from Theorem 1, Lemma 3 and Theorem 2. □

Remark 2 (Local Minima of ℓ_1-LRA)**.** Initially, we thought that any local minimum of ℓ_1-LRA (4) of a $\{-1,+1\}$ matrix can be assumed, w.l.o.g., to have entries in $\{-1,+1\}$. However, this is not always true. Here is a counter example: For

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & -1 & 1 & 1 \\ -1 & 1 & -1 & -1 & 1 & 1 \\ -1 & -1 & 1 & -1 & 1 & 1 \\ -1 & -1 & -1 & 1 & 1 & 1 \end{pmatrix}$$

the solution

$$u = [1,1,x,x,x,x]^T, \qquad v = [1,1,1,1,1/x,1/x]^T$$

is a stationary point[2] of ℓ_1-LRA for any $0.5 < x < 1$, and is a local minimum for $x = \sqrt{2}/2$ (there is a segment of stationary points with a local minimum in its interior) with error 23.3. Using the "Move 2" from the proof of Theorem 2, we obtain

$$u = [1,1,-1,-1,-1,-1]^T, \qquad v = [1,1,1,1,-1,-1]^T$$

which is an optimal solution with error 16 (with 8 mismatches).

Theorem 2 also has practical implications: In fact, the difficult combinatorial problem (10) of computing the norm $\|\cdot\|_{\infty\to 1}$ has a continuous characterization given by ℓ_1-LRA (4). Therefore, a simple heuristic for computing the norm $\|\cdot\|_{\infty\to 1}$ is to use any (iterative) nonlinear optimization scheme for ℓ_1-LRA (see the Introduction). Moreover, we have a good initial candidate, i.e., the solution of ℓ_2-LRA that can be efficiently computed via the truncated singular value decomposition. The same continuous characterization can be used to solve other closely related combinatorial problems such as the densest bipartite subgraph problem described in Section 2. We have implemented a simple cyclic coordinate descent method in Matlab for ℓ_1-LRA, as described in Ke and Kanade [22]. The code is available from https://sites.google.com/site/nicolasgillis/code, and also contains the matrices from Remark 2 and Example 1.

5. Higher-Rank Matrix Approximations

It is easy to generalize our rank-one NP-hardness results to higher ranks. In fact, the same construction as in Gillis and Glineur [16, Theorem 3] can be used. For example, for the rank-one BMF problem, instead of considering the binary input matrix M as in the rank-one case, we consider the input matrix

$$A = \begin{pmatrix} M & 0 & \cdots & 0 \\ 0 & M & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & M \end{pmatrix} = \operatorname{diag}(M, r).$$

The idea is that an optimal rank-r approximation of A is constituted of rank-one factors that do not have nonzero entries in more than one diagonal block (otherwise the solution can be improved). Hence an optimal solution for rank-r BMF of A leads to a combination of optimal rank-one approximations of M, one for each block; see the proof in Gillis and Glineur [16, Theorem 3] for more details.

Note however that using this construction, the ratio between the factorization rank r and the dimension of the input matrix remains unchanged. Moreover, ℓ_1-LRA with $r = \min(m,n) - 1$ can be solved in polynomial time using linear programming (Brooks and Dulá [4], Brooks et al. [5]). Therefore, it remains an open question whether ℓ_1-LRA is NP-hard for different values of r depending on m and n. For example, is ℓ_1-LRA a difficult problem for $r = \min(m,n) - 2$?

RIGHTSLINK

Downloaded from informs.org by [130.113.111.210] on 14 June 2018, at 09:10 . For personal use only, all rights reserved.

6. Conclusion and Future Work

The main results of this paper are

- The equivalence between robust PCA, rank-one ℓ_0 low-rank matrix approximation, rank-one binary matrix factorization, a particular densest bipartite subgraph problem, the computation of the cut norm and the norm $\|\cdot\|_{\infty\to 1}$ of $\{-1,+1\}$ matrices, and the rank-one discrete basis problem;
- The proof that the optimal rank-one solution of ℓ_1-LRA of $\{-1,+1\}$ matrices can be assumed, w.l.o.g., to be a $\{-1,+1\}$ matrix (Theorem 2);
- The NP-hardness of the computation of the norm $\|\cdot\|_{\infty\to 1}$ of $\{-1,+1\}$ matrices (Theorem 1) which allowed us to prove NP-hardness of all the problems listed above.

After the publication of a preprint of this paper (Gillis and Vavasis [18]), Song, Woodruff and Zhong proposed several approximation algorithms for ℓ_1-LRA (Song et al. [37]), addressing the second part of open question 2 in Woodruff [39]. In particular, they showed that it is possible to achieve an approximation factor $\alpha = (\log n)\cdot \mathrm{poly}(r)$ in $\mathrm{nnz}(M) + (m+n)\mathrm{poly}(r)$ time, where $\mathrm{nnz}(M)$ denotes the number of non-zero entries of M. If r is constant, they further improve the approximation ratio to $O(1)$ with a $\mathrm{poly}(mn)$-time algorithm. They also discuss the extension of their results to ℓ_p-LRA for $1 < p < 2$, and to other related problems, where ℓ_p-LRA is defined as

$$\min_{U\in\mathbb{R}^{p\times r},\, V\in\mathbb{R}^{r\times n}} \|M - UV\|_p = \sum_{i,j} |M_{ij} - (UV)_{ij}|^p.$$

Directions for further research include the study of the complexity of ℓ_p-LRA, for $p \notin \{0,1\}$, except for $p = 2$ that can be solved in polynomial time via the singular value decomposition. We expect that the techniques introduced in Section 4 can be extended to show NP-hardness of ℓ_p-LRA for p in $(0,1)$ (because optimal solutions have the same structure as for the ℓ_1-norm). It would be particularly interesting to know whether ℓ_∞-LRA is NP-hard, in particular in the rank-one case. In fact, it was shown to be NP-hard for $r = \min(m,n) - 1$ in Poljak and Rohn [32] (although the problem is stated in a slightly different way).

Acknowledgments

The authors thank the area editor, Jens Vygen, and two anonymous reviewers for their insightful comments that helped improve the paper.

Endnotes

[1] You can run this example with our code available from https://sites.google.com/site/nicolasgillis/code.

[2] We say that (u,v) is a stationary point of ℓ_1-LRA if $u = \arg\min_x \|M - xv^T\|_1$ and $v = \arg\min_y \|M - uy^T\|_1$.

References

[1] Alon N, Naor A (2006) Approximating the cut-norm via Grothendieck's inequality. *SIAM J. Comput.* 35(4):787–803.
[2] Ames B, Vavasis S (2011) Nuclear norm minimization for the planted clique and biclique problems. *Math. Programming* 129(1):69–89.
[3] Asahiro Y, Hassin R, Iwama K (2002) Complexity of finding dense subgraphs. *Discrete Appl. Math.* 121(1):15–26.
[4] Brooks J, Dulá J (2013) The L1-norm best-fit hyperplane problem. *Appl. Math. Lett.* 26(1):51–55.
[5] Brooks J, Dulá J, Boone E (2013) A pure L_1-norm principal component analysis. *Computational Statist. Data Anal.* 61:83–98.
[6] Brown T, Spencer J (1971) Minimization of ±1 matrices under line shifts. *Colloquium Mathematicae*, Vol. 23 (Institute of Mathematics Polish Academy of Sciences), 165–171.
[7] Candès E, Li X, Ma Y, Wright J (2011) Robust principal component analysis? *J. ACM* 58(3):Article 11.
[8] Chandrasekaran V, Sanghavi S, Parrilo P, Willsky A (2011) Rank-sparsity incoherence for matrix decomposition. *SIAM J. Optim.* 21(2): 572–596.
[9] Chi E, Kolda T (2012) On tensors, sparsity, and nonnegative factorizations. *SIAM J. Matrix Anal. Appl.* 33(4):1272–1299.
[10] Clarkson K, Woodruff D (2015) Input sparsity and hardness for robust subspace approximation. *56th Annual IEEE Sympos. Foundations Comput. Sci.* (*FOCS* 2015).
[11] Doan X, Vavasis S (2013) Finding approximately rank-one submatrices with the nuclear norm and ℓ_1-norm. *SIAM J. Optim.* 23(4): 2502–2540.
[12] Eriksson A, Van Den Hengel A (2010) Efficient computation of robust low-rank matrix approximations in the presence of missing data using the l_1 norm. *IEEE Conf. Comput. Vision and Pattern Recognition* (*CVPR '10*), 771–778.
[13] Fiorini S, Kaibel V, Pashkovich K, Theis D (2013) Combinatorial bounds on nonnegative rank and extended formulations. *Discrete Math.* 313(1):67–83.
[14] Frieze A, Kannan R (1999) Quick approximation to matrices and applications. *Combinatorica* 19(2):175–220.
[15] Gabriel K, Zamir S (1979) Lower rank approximation of matrices by least squares with any choice of weights. *Technometrics* 21(4): 489–498.
[16] Gillis N, Glineur F (2010) Using underapproximations for sparse nonnegative matrix factorization. *Pattern Recognition* 43(4):1676–1687.
[17] Gillis N, Glineur F (2011) Low-rank matrix approximation with weights or missing data is NP-hard. *SIAM J. Matrix Anal. Appl.* 32(4): 1149–1165.
[18] Gillis N, Vavasis S (2015) On the Complexity of Robust PCA and ℓ_1-norm Low-Rank Matrix Approximation. arXiv:1509.09236.

RIGHTSLINK

[19] Golub G, Van Loan C (1996) *Matrix Computation*, 3rd ed. (The Johns Hopkins University Press, Baltimore).
[20] Guruswami V, Raghavendra P, Saket R, Wu Y (2012) Bypassing UGC from some optimal geometric inapproximability results. *Proc. Twenty-Third Annual ACM-SIAM Sympos. Discrete Algorithms* (*SODA 2012*), 699–717.
[21] Kannan R, Vinay V (1999) Analyzing the structure of large graphs. Manuscript, http://www.cs.yale.edu/homes/kannan/Papers/webgraph.pdf.
[22] Ke Q, Kanade T (2003) Robust subspace computation using L1 norm. Technical Report CMU-CS-03-172, Carnegie Mellon University, http://www.cs.cmu.edu/afs/.cs.cmu.edu/Web/People/ke/publications/CMU-CS-03-172.pdf.
[23] Ke Q, Kanade T (2005) Robust l_1 norm factorization in the presence of outliers and missing data by alternative convex programming. *IEEE Comput. Society Conf. Comput. Vision and Pattern Recognition, CVPR '05* (IEEE Computer Society, Washington, DC), 739–746.
[24] Khot S (2006) Ruling out PTAS for graph min-bisection, dense k-subgraph, and bipartite clique. *SIAM J. Comput.* 36(4):1025–1071.
[25] Khuller S, Saha B (2009) On finding dense subgraphs. Albers S, Marchetti-Spaccamela A, Matias Y, Nikoletseas S, Thomas W, eds. *Proc. 36th Internat. Colloquium Automata, Languages and Programming, ICALP '09*, Lecture Notes in Computer Science, Vol. 5555 (Springer, Berlin), 597–608.
[26] Koren Y, Bell R, Volinsky C (2009) Matrix factorization techniques for recommender systems. *IEEE Comput.* 42(8):30–37.
[27] Kwak N (2008) Principal component analysis based on L1-norm maximization. *IEEE Trans. Pattern Anal. Machine Intelligence* 30(9): 1672–1680.
[28] Markovsky I, Usevich K (2013) Structured low-rank approximation with missing data. *SIAM J. Matrix Anal. Appl.* 34(2):814–830.
[29] Miettinen P, Mielikainen T, Gionis A, Das G, Mannila H (2008) The discrete basis problem. *IEEE Trans. Knowledge and Data Engrg.* 20(10):1348–1362.
[30] Mirisaee S, Gaussier E, Termier A (2015) Improved local search for binary matrix factorization. *AAAI Conf. Artificial Intelligence*, 1198–1204.
[31] Netrapalli P, Niranjan U, Sanghavi S, Anandkumar A (2014) Non-convex robust PCA. *Adv. Neural Inform. Processing Systems* (*NIPS*), 2080–2088.
[32] Poljak S, Rohn J (1993) Checking robust nonsingularity is NP-hard. *Math. Control, Signals and Systems* 6(1):1–9.
[33] Qiu C, Vaswani N, Lois B, Hogben L (2014) Recursive robust PCA or recursive sparse recovery in large but structured noise. *IEEE Trans. Inform. Theory* 60(8):5007–5039.
[34] Rohn J (2000) Computing the norm $||A||_{\infty,1}$ is NP-hard. *Linear and Multilinear Algebra* 47(3):195–204.
[35] Shen BH, Ji S, Ye J (2009) Mining discrete patterns via binary matrix factorization. Elder JF IV, Fogelman-Soulié F, Flach PA, Javeed Zaki M, eds. *Proc. 15th ACM SIGKDD Internat. Conf. Knowledge Discovery and Data Mining, KDD '09* (ACM, New York), 757–766.
[36] Shum H, Ikeuchi K, Reddy R (1995) Principal component analysis with missing data and its application to polyhedral object modeling. *IEEE Trans. Pattern Anal. Machine Intelligence* 17(9):854–867.
[37] Song Z, Woodruff D, Zhong P (2017) Low rank approximation with entrywise ℓ_1-norm error. Hatami H, McKenzie P, King V, eds. *Proc. 49th Annual ACM SIGACT Sympos. Theory Comput., STOC '17* (ACM, New York), 688–701.
[38] Usevich K, Markovsky I (2014) Optimization on a Grassmann manifold with application to system identification. *Automatica* 50(6): 1656–1662.
[39] Woodruff D (2014) Sketching as a tool for numerical linear algebra. *Foundations and Trends® Theoret. Comput. Sci.* 10(1–2):1–157.
[40] Wright J, Ganesh A, Rao S, Peng Y, Ma Y (2009) Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. *Adv. Neural Inform. Processing Systems* (*NIPS*), 2080–2088.
[41] Xu H, Caramanis C, Sanghavi S (2012) Robust PCA via outlier pursuit. *IEEE Trans. Inform. Theory* 58(5):3047–3064.
[42] Zhang ZY, Li T, Ding C, Ren XW, Zhang XS (2010) Binary matrix factorization for analyzing gene expression data. *Data Min. Knowl. Disc.* 20(1):28–52.
[43] Zheng Y, Liu G, Sugimoto S, Yan S, Okutomi M (2012) Practical low-rank matrix approximation under robust L1-norm. *IEEE Conf. Comput. Vision and Pattern Recognition, CVPR '12* (IEEE Computer Society, Washington, DC), 1410–1417.

Downloaded from informs.org by [130.113.111.210] on 14 June 2018, at 09:10 . For personal use only, all rights reserved.

RIGHTSLINK