

# SVD-free Convex-Concave Approaches for Nuclear Norm Regularization

Yichi Xiao<sup>1\*</sup>, Zhe Li<sup>2\*</sup>, Tianbao Yang<sup>2</sup>, Lijun Zhang<sup>1</sup>

<sup>1</sup>Nanjing University of Aeronautics and Astronautics, Nanjing, China  
<sup>2</sup>Department of Computer Science, University of California, Irvine, CA, USA  
 {yichixiao, zhe.li}@nuaa.edu.cn, {tianbao, lijun}@uci.edu

## Abstract

Minimizing the nuclear norm of a matrix is a non-smooth problem. In this paper, we propose a convex-concave approach for nuclear norm regularization. We first reformulate the nuclear norm as the sum of the singular values of a matrix. Then, we use the singular value decomposition (SVD) to decompose the matrix into three matrices. The nuclear norm is then expressed as the sum of the singular values of the middle matrix. This reformulation allows us to use the first SVD-free approach to solve the problem. We show that the proposed approach is more efficient than the first SVD-free approach. The proposed approach is based on the singular value decomposition (SVD) of the matrix. The nuclear norm is the sum of the singular values of the matrix. We use the SVD to decompose the matrix into three matrices. The nuclear norm is then expressed as the sum of the singular values of the middle matrix. This reformulation allows us to use the first SVD-free approach to solve the problem. We show that the proposed approach is more efficient than the first SVD-free approach.

## 1 Introduction

Low-rank matrix approximation is a fundamental problem in many applications. The nuclear norm is a common regularization for low-rank matrix approximation. However, minimizing the nuclear norm is a non-smooth problem. In this paper, we propose a convex-concave approach for nuclear norm regularization. We first reformulate the nuclear norm as the sum of the singular values of a matrix. Then, we use the singular value decomposition (SVD) to decompose the matrix into three matrices. The nuclear norm is then expressed as the sum of the singular values of the middle matrix. This reformulation allows us to use the first SVD-free approach to solve the problem. We show that the proposed approach is more efficient than the first SVD-free approach.

\*Equal Contribution

1

$$\min_{A \in \mathbb{R}^{m \times n}} f(A) = g(A) + \|A\|_* \quad (1)$$

where  $f(\cdot)$  is a convex function,  $g(\cdot)$  is a concave function, and  $\|\cdot\|_*$  is the nuclear norm. The nuclear norm is defined as the sum of the singular values of a matrix. We use the SVD to decompose the matrix into three matrices. The nuclear norm is then expressed as the sum of the singular values of the middle matrix. This reformulation allows us to use the first SVD-free approach to solve the problem. We show that the proposed approach is more efficient than the first SVD-free approach.

In this paper, we propose a convex-concave approach for nuclear norm regularization. We first reformulate the nuclear norm as the sum of the singular values of a matrix. Then, we use the singular value decomposition (SVD) to decompose the matrix into three matrices. The nuclear norm is then expressed as the sum of the singular values of the middle matrix. This reformulation allows us to use the first SVD-free approach to solve the problem. We show that the proposed approach is more efficient than the first SVD-free approach.

**Applications** A matrix  $A$  is often used to represent a graph, where the entries  $A_{ij}$  represent the weights of the edges between nodes  $i$  and  $j$ . In this case,  $A$  is symmetric and non-negative. The Laplacian matrix  $L$  is defined as  $L = D - A$ , where  $D$  is the degree matrix. The Laplacian matrix is used in many applications, such as spectral clustering and graph partitioning.

- **Robust Low-rank Matrix Approximation:** This problem involves finding a low-rank matrix  $A$  that approximates a given matrix  $B$ . The Frobenius norm is used to measure the error between  $A$  and  $B$ . The problem is often solved using the Alternating Least Squares (ALS) algorithm. The Frobenius norm is defined as  $\|A\|_F = \sqrt{\sum_{ij} |A_{ij}|^2}$ .

- **Sparse and Low-rank Link Prediction:** This problem involves predicting missing links in a network. The network is represented by a matrix  $A$ , and the goal is to find a low-rank matrix  $\hat{A}$  that approximates  $A$ . The L1 norm is used to regularize the matrix  $\hat{A}$ . The problem is often solved using the Alternating Least Squares (ALS) algorithm. The L1 norm is defined as  $\|A\|_1 = \sum_{ij} |A_{ij}|$ .

## 2 Related Work

In this paper, we propose a new method for solving the robust low-rank matrix approximation problem. Our method is based on the Alternating Least Squares (ALS) algorithm, but we use a different regularization function to improve the performance of the algorithm.

### 2.1 Nuclear-norm Regularized Problems

One of the most common methods for solving the robust low-rank matrix approximation problem is the Alternating Least Squares (ALS) algorithm. The ALS algorithm is based on the following optimization problem:

$$A_{t+1} = A_t - \eta_t (\nabla (A_t) + \|A_t\|_*)$$

where  $\|A_t\|_*$  is the nuclear norm of  $A_t$ . The nuclear norm is defined as the sum of the singular values of  $A_t$ . The ALS algorithm is often used to solve the robust low-rank matrix approximation problem. In this paper, we propose a new method for solving the robust low-rank matrix approximation problem. Our method is based on the Alternating Least Squares (ALS) algorithm, but we use a different regularization function to improve the performance of the algorithm.

---

**Algorithm 1** SVD-based CON-CVEA (SECONE)

---

1: **Initialize:**  $\mathbf{A}_1 = \mathbf{1} = \mathbf{0} \in \mathbb{R}^{m \times n}$   
 2: **for**  $t = 1$  **to**  $T$  **do**  
 3:    $\mathbf{U} \leftarrow \mathbf{A}_{t+1}$   
        $\mathbf{A}_{t+1} = \mathbf{A}_t - \eta_t (\mathbf{A}_t) + \mathbf{1}_t$   
 4:    $\mathbf{U} \leftarrow \mathbf{1}_{t+1}$   
        $\mathbf{1}_{t+1} = \mathbf{1}_t + \mathbf{1}(\|\mathbf{A}_t - \mathbf{1}_t\|_2 - 1)_+$   
 5: **end for**  
 6: **Output:**  $\hat{\mathbf{A}}_T = \sum_{t=1}^T \mathbf{A}_t / T$

---

Suppose  $\mathbf{v} \in \mathbb{R}^n$  is a unit vector,  $\mathbf{H} = \mathbf{v}\mathbf{v}^\top$ . For any  $\mathbf{A} \in \mathbb{R}^{m \times n}$ , we have  $\mathbf{A}^\top \mathbf{H} \mathbf{A} = (\mathbf{A}^\top \mathbf{v})(\mathbf{v}^\top \mathbf{A})$ . Let  $\mathbf{u} = \mathbf{A}^\top \mathbf{v} / \|\mathbf{A}^\top \mathbf{v}\|_2$  and  $\mathbf{1} = \mathbf{v}\mathbf{v}^\top$ . Then  $\mathbf{A}^\top \mathbf{H} \mathbf{A} = \|\mathbf{A}^\top \mathbf{v}\|_2 \mathbf{u}\mathbf{u}^\top$ . By SVD, we can write  $\mathbf{u}\mathbf{u}^\top = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^\top$ , where  $\mathbf{V} \in \mathbb{R}^n$  is an orthogonal matrix and  $\mathbf{\Lambda} \in \mathbb{R}^n$  is a diagonal matrix with non-negative entries. Let  $\mathbf{1}_t = \mathbf{V}\mathbf{\Lambda}_t\mathbf{V}^\top$ , where  $\mathbf{\Lambda}_t \in \mathbb{R}^n$  is a diagonal matrix with non-negative entries. Then  $\mathbf{1}_t \mathbf{1}_t = \mathbf{1}_t$  and  $\|\mathbf{1}_t\|_2 \leq 1$ . For any  $\mathbf{A} \in \mathbb{R}^{m \times n}$ , we have  $\mathbf{A}^\top \mathbf{1}_t \mathbf{A} = \mathbf{A}^\top \mathbf{V}\mathbf{\Lambda}_t\mathbf{V}^\top \mathbf{A} = (\mathbf{A}^\top \mathbf{V})\mathbf{\Lambda}_t(\mathbf{V}^\top \mathbf{A})$ .

$$\min_{\mathbf{A} \in \mathbb{R}^{m \times n}} \max_{\mathbf{U} \in \mathbb{R}^{m \times n}} (\mathbf{A}) + \mathbf{1}_t (\mathbf{A}^\top \mathbf{U}) - [\|\mathbf{U}\|_2 - 1]_+ \quad (3)$$

Let  $\mathbf{1}_t = \mathbf{V}\mathbf{\Lambda}_t\mathbf{V}^\top$ , where  $\mathbf{\Lambda}_t \in \mathbb{R}^n$  is a diagonal matrix with non-negative entries. Then  $\mathbf{1}_t \mathbf{1}_t = \mathbf{1}_t$  and  $\|\mathbf{1}_t\|_2 \leq 1$ . For any  $\mathbf{A} \in \mathbb{R}^{m \times n}$ , we have  $\mathbf{A}^\top \mathbf{1}_t \mathbf{A} = \mathbf{A}^\top \mathbf{V}\mathbf{\Lambda}_t\mathbf{V}^\top \mathbf{A} = (\mathbf{A}^\top \mathbf{V})\mathbf{\Lambda}_t(\mathbf{V}^\top \mathbf{A})$ .

$$\mathbf{A}_{t+1} = \mathbf{A}_t - \eta_t (\mathbf{A}_t) + \mathbf{1}_t$$

$$\mathbf{1}_{t+1} = \mathbf{1}_t + \mathbf{1}(\|\mathbf{A}_t - \mathbf{1}_t\|_2 - 1)_+$$

Note that  $\|\mathbf{1}_t\|_2 \leq 1$  and  $\mathbf{1}_t \mathbf{1}_t = \mathbf{1}_t$ .

**Algorithm 2**  $\mathcal{E} \text{ r } \mathcal{P} \text{ r } \mathcal{S} \text{ r } \mathcal{S} \text{ r } \mathcal{P}$  (SECONE-S)

```

1: Initialize:  $\mathbf{A}_1 = \mathbf{1} = 0 \in \mathbb{R}^{m \times n}$ 
2: for  $t = 1$  to  $T$  do
3:    $\mathcal{S}$ 
4:    $\mathbf{U} \leftarrow \mathbf{A}_{t+1}$ 
        $\mathbf{A}_{t+1} = \mathbf{A}_t - \eta_t (\langle \mathbf{A}_t; \mathbf{t} \rangle + \mathbf{t})$ 
5:    $\mathbf{U} \leftarrow \mathbf{t}_{t+1}$ 
        $\mathbf{t}_{t+1} = \dots ( \|\mathbf{t}\|_2 - 1)_+$ 
6: end for
7: Output:  $\mathbf{A}$ 

```

$T$   $v$   $r$   $\dots$   $\mathcal{V} \text{ r } \mathcal{P} \text{ f } \text{ r } \mathcal{I}$   
 $r$   $f$   $r$   $\dots$   $r$   $f$   $r$   $\dots$   
 $\mathcal{P} \text{ r } \mathcal{P} \text{ r } \mathcal{P} \text{ r } \mathcal{P}$   $\mathcal{P} \text{ r } \mathcal{P} \text{ r } \mathcal{P} \text{ r } \mathcal{P}$   
 $\mathcal{I} \text{ r } \mathcal{V} \text{ r } \mathcal{P} \text{ r } \mathcal{P}$   $\mathcal{I} \text{ r } \mathcal{V} \text{ r } \mathcal{P} \text{ r } \mathcal{P}$

**3.3 Problem with Total Regularizer**

$\dots$

$\dots$

$\dots$

$\dots$

$\dots$

2 11dof [374-0 1(Algorithm)-37,2 d [3 :w249-995(updatuceorem)-62.002(0d (-)T)/T1\_71\_7874 Ff80T

$$\begin{aligned} L_t \eta_t &= \frac{1}{2\eta_t} (\|A - A_t\|_F^2 - \|A - A_{t+1}\|_F^2) \\ &\leq \frac{1}{2\eta_1} \|A - A_1\|_F^2 + \sum_{t=1}^{T-1} \left( \frac{1}{2\eta_{t+1}} - \frac{1}{2\eta_t} \right) \|A - A_t\|_F^2 \\ &\leq \frac{1}{2\eta_1} \eta_1^2 + \left( \frac{1}{2\eta_T} - \frac{1}{2\eta_1} \right) \eta_1^2 \leq \frac{1}{2\eta_T} \eta_1^2 = \frac{\sqrt{T}}{2} \eta_1^2 \\ \text{where } \eta_1 &= \|A\| + \sqrt{\sum_{t=1}^T \frac{1}{\eta_t}} \geq \max_t \|A - A_t\|_F. \end{aligned}$$

$$\begin{aligned} \sum_{t=1}^T (A_t - A) &\leq \sum_{t=1}^T (A - A_t) + \frac{\sqrt{T}}{2} \eta_1^2 + \frac{\sqrt{T}}{2} \eta_1^2 \\ &\quad + \frac{1}{2} \sqrt{T} (\eta_1^2 + \eta_1^2) \\ \text{where } \sum_{t=1}^T \frac{1}{\sqrt{t}} &\leq 1 + \int_{t=1}^T \frac{1}{\sqrt{t}} dt \leq 2\sqrt{T} \\ \text{where } \sum_{t=1}^T \frac{1}{t} &= \ln T + \gamma \leq \ln T + 1 \\ \text{where } \hat{A}_T &= \sum_{t=1}^T \frac{A_t}{T}, \quad \tilde{A}_T = \sum_{t=1}^T \frac{1}{t} A_t. \end{aligned}$$

$$\begin{aligned} \|\hat{A}_T - A\|_F &\leq \frac{1}{\sqrt{T}} \left( \frac{\eta_1^2}{2} + \frac{1}{2} (\eta_1^2 + \eta_1^2) \right) \\ &\quad + \frac{1}{\sqrt{T}} \left( \frac{\eta_1^2}{2} + \frac{1}{2} (\eta_1^2 + \eta_1^2) \right) \end{aligned}$$

Part II: Let  $\hat{A}_T = \sum_{t=1}^T \frac{A_t}{T}$ . We have  $\|\hat{A}_T - A\|_F \leq \frac{1}{\sqrt{T}} (\frac{\eta_1^2}{2} + \frac{1}{2} (\eta_1^2 + \eta_1^2)) + \frac{1}{\sqrt{T}} (\frac{\eta_1^2}{2} + \frac{1}{2} (\eta_1^2 + \eta_1^2))$ .

$$\begin{aligned} \|\hat{A}_T - A_*\|_F &\leq \|\hat{A}_T - \hat{A}_T^*\|_F + \|\hat{A}_T^* - A_*\|_F \\ \text{where } \hat{A}_T^* &= \arg \max_{\|U\|_2 \leq 1} t(\hat{A}_T). \text{ For } \hat{A}_T^*, \\ \|\hat{A}_T^* - A_*\|_F &= \|\hat{A}_T^* - \hat{A}_T\|_F + \|\hat{A}_T - A_*\|_F \end{aligned}$$

$$\begin{aligned} \|\hat{A}_T^* - A_*\|_F &\leq t(\hat{A}_T^*) - [(\|\hat{A}_T\|_2 - 1)]_+ \\ \text{where } t(\hat{A}_T^*) &\leq \|\hat{A}_T\|_2 \|\hat{A}_T^*\|_F \leq \|\hat{A}_T\|_2 \|\hat{A}_T\|_F \\ \text{where } \|\hat{A}_T\|_2 &\leq 1, \text{ where } \|\hat{A}_T\|_2 \geq 1. \\ \text{Let } \hat{A}_T &= \sum_{i=1}^m \lambda_i \hat{u}_i \hat{u}_i^T, \text{ where } \hat{u}_i \text{ are the top } m \text{ eigenvectors of } \hat{A}_T. \\ \text{where } \lambda_i &= \frac{1}{\sum_{j=1}^m \lambda_j} \sum_{j=1}^m \lambda_j \lambda_i. \\ \text{where } \tilde{A}_T &= \sum_{i=1}^m \tilde{\lambda}_i \tilde{u}_i \tilde{u}_i^T, \text{ where } \tilde{u}_i \text{ are the top } m \text{ eigenvectors of } \tilde{A}_T. \\ \text{where } \tilde{\lambda}_i &= \frac{1}{\sum_{j=1}^m \tilde{\lambda}_j} \sum_{j=1}^m \tilde{\lambda}_j \tilde{\lambda}_i. \end{aligned}$$

$$\begin{aligned} \|\hat{A}_T^* - A_*\|_F &\leq t(\hat{A}_T^*) - [(\|\hat{A}_T\|_2 - 1)]_+ \\ &\leq \|\hat{A}_T - \tilde{A}_T\|_2 \|\hat{A}_T^*\|_F \leq (\|\hat{A}_T\|_2 - 1) \\ &\leq (\|\hat{A}_T\|_2 - 1) \\ \text{where } \|\hat{A}_T\|_2 &\leq 1, \text{ where } \|\hat{A}_T\|_2 \geq 1. \end{aligned}$$

### 4.2 Proof of Theorem 3

$$\begin{aligned} \text{where } (A_t) &= (A) + t(\hat{A}_T) + \gamma(A) - [(\|\hat{A}_T\|_2 - 1)]_+ \\ \text{where } t(A) &= (A) + t(\hat{A}_T) - \gamma(A) \end{aligned}$$

$$\begin{aligned} \text{where } t(A) &= (A) + t(\hat{A}_T) - \gamma(A) \\ \text{where } t(A) &= t(A_t) - \gamma(A_t) \\ \text{where } \eta_t &= \langle A_t - A, \eta_t \rangle + \langle A_{t+1} - A, \eta_t \rangle \\ &\leq \langle A_t - A, \eta_t \rangle + \langle A_{t+1} - A, \eta_t \rangle \\ &= \langle A - A_{t+1}, A_t - A_{t+1} - \eta_t \rangle - \langle A - A_{t+1}, A_{t+1} - A_t \rangle \\ &\quad + \langle A - A_{t+1}, A_{t+1} - A_t \rangle + \eta_t \langle A_t - A_{t+1}, \eta_t \rangle \end{aligned}$$

$$\text{where } \langle A - A_{t+1}, A_{t+1} - A_t + \eta_t \rangle + \eta_t \langle A_t - A_{t+1}, \eta_t \rangle \geq 0$$

$$\begin{aligned} \text{where } \eta_t &= \langle A_t - A, \eta_t \rangle + \langle A_{t+1} - A, \eta_t \rangle \\ &\leq \langle A_* - A_{t+1}, A_{t+1} - A_t \rangle + \eta_t \langle A_t - A_{t+1}, \eta_t \rangle \\ &\leq \frac{1}{2} (\|A_* - A_t\|_F^2 - \|A_* - A_{t+1}\|_F^2 - \|A_{t+1} - A_t\|_F^2) \\ &\quad + \frac{1}{2} (\|A_t - A_{t+1}\|_F^2 + \eta_t^2 \|\eta_t\|_F^2) \\ &\leq \frac{1}{2} (\|A_* - A_t\|_F^2 - \|A_* - A_{t+1}\|_F^2) + \frac{\eta_t^2}{2} \|\eta_t\|_F^2 \end{aligned}$$

$$\begin{aligned} \text{where } t(A_t) &= (A_t) - \gamma(A_t) \\ &= t(A_t) + (A_t) - t(A_*) - \gamma(A_*) \\ &\leq \frac{1}{2\eta_t} (\|A_* - A_t\|_F^2 - \|A_* - A_{t+1}\|_F^2) + \frac{\eta_t}{2} \|\eta_t\|_F^2 \\ &\quad + \gamma((A_t) - (A_{t+1})) \end{aligned}$$

$$\begin{aligned} \text{where } \gamma((A_t) - (A_{t+1})) &= \gamma(A_1) - \gamma(A_2) = 0 \\ \text{where } \gamma(A_1) &= (0) = 0. \end{aligned}$$

### 5 Experiments

where  $\hat{A}_T = \sum_{t=1}^T \frac{A_t}{T}$ . We have  $\|\hat{A}_T - A\|_F \leq \frac{1}{\sqrt{T}} (\frac{\eta_1^2}{2} + \frac{1}{2} (\eta_1^2 + \eta_1^2)) + \frac{1}{\sqrt{T}} (\frac{\eta_1^2}{2} + \frac{1}{2} (\eta_1^2 + \eta_1^2))$ .

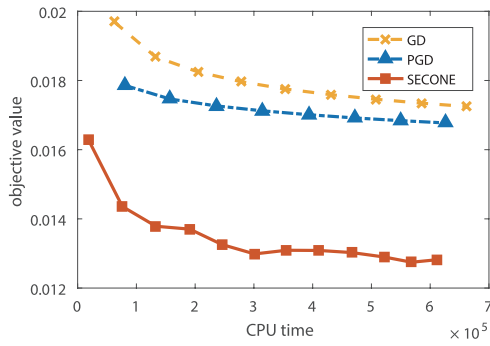


Figure 1: R<sub>1</sub> (GD, PGD, SECONE)

T<sub>1</sub>: Sparse matrix

Method	1	2	T	T <sub>1</sub> CPU
SECONE	1.7	1.4	8500	6.125
PGD	1.6		80	6.265
GD	1.6		90	6.625

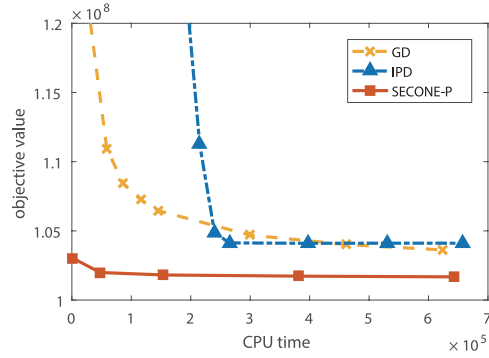


Figure 2: R<sub>2</sub> (GD, IPD, SECONE-P)

T<sub>2</sub>: Sparse matrix

Method	1	2	T	T <sub>2</sub> CPU
SECONE-P	1	1-5	2000	6.435
IPD	0.1		50	6.575
GD	1		40	6.255

### 5.1 Robust Low-rank Matrix Approximation

Wang et al., 2012; Crussell et al., 1996; Crussell et al., 1998; Kraskov et al., 2005 :]

$$\min_{A \in \mathbb{R}^{m \times n}} \frac{1}{m} \sum_{i=1}^m \sum_{j=1}^n |a_{ij} - A_{ij}| + \|A\|_*$$

Robust low-rank matrix approximation is a fundamental problem in machine learning and data analysis. The goal is to find a low-rank matrix  $A$  that approximates a given matrix  $A$  in a robust manner, minimizing the sum of absolute errors and the nuclear norm of  $A$ . This problem is NP-hard in general, but can be solved approximately using various algorithms. In this paper, we compare the performance of three algorithms: GD, PGD, and SECONE. We use a synthetic dataset with  $m=20^2$  rows and  $n=20 \cdot 302$  columns. The matrix  $A$  is generated by  $A = UV^T + E$ , where  $U$  and  $V$  are random matrices of size  $m \times r$  and  $n \times r$  respectively, and  $E$  is a sparse matrix with non-zero entries at  $(1, 2)$ ,  $(2, 1)$ ,  $(1, 10)$ , and  $(10, 1)$ . The rank  $r$  is 10. We compare the performance of three algorithms: GD, PGD, and SECONE. The results are shown in Table 1. SECONE consistently outperforms GD and PGD in terms of both the objective value and the CPU time.

### 5.2 Sparse and Low-rank Link Prediction

Given a sparse matrix  $A \in \mathbb{R}^{m \times n}$ , the goal is to predict missing entries in  $A$ . This problem can be formulated as a matrix completion problem. In this paper, we compare the performance of three algorithms: GD, PGD, and SECONE. We use a synthetic dataset with  $m=20^2$  rows and  $n=20 \cdot 302$  columns. The matrix  $A$  is generated by  $A = UV^T + E$ , where  $U$  and  $V$  are random matrices of size  $m \times r$  and  $n \times r$  respectively, and  $E$  is a sparse matrix with non-zero entries at  $(1, 2)$ ,  $(2, 1)$ ,  $(1, 10)$ , and  $(10, 1)$ . The rank  $r$  is 10. We compare the performance of three algorithms: GD, PGD, and SECONE. The results are shown in Table 1. SECONE consistently outperforms GD and PGD in terms of both the objective value and the CPU time.

$$\min_{A \in \mathbb{R}^{m \times n}} \sum_{ij} \max(1 - (2|a_{ij} - 1| - 1) \cdot A_{ij}, 0) + \gamma \|A\|_1 + \|A\|_*$$

<sup>2</sup>  $\|A\|_1 = \sum_{i,j} |a_{ij}|$ ,  $\|A\|_* = \sum \sigma_i(A)$

Robust low-rank matrix approximation is a fundamental problem in machine learning and data analysis. The goal is to find a low-rank matrix  $A$  that approximates a given matrix  $A$  in a robust manner, minimizing the sum of absolute errors and the nuclear norm of  $A$ . This problem is NP-hard in general, but can be solved approximately using various algorithms. In this paper, we compare the performance of three algorithms: GD, PGD, and SECONE. We use a synthetic dataset with  $m=41 \cdot 554$  rows and  $n=15 \cdot 000$  columns. The matrix  $A$  is generated by  $A = UV^T + E$ , where  $U$  and  $V$  are random matrices of size  $m \times r$  and  $n \times r$  respectively, and  $E$  is a sparse matrix with non-zero entries at  $(1, 2)$ ,  $(2, 1)$ ,  $(1, 10)$ , and  $(10, 1)$ . The rank  $r$  is 10. We compare the performance of three algorithms: GD, PGD, and SECONE. The results are shown in Table 2. SECONE-P consistently outperforms GD and IPD in terms of both the objective value and the CPU time.

### Acknowledgments

This work is supported by NSFC (61603177), Jilin SF (BK20160658), and the Chinese Academy of Sciences (KJ951-A1-100-01-0001). We thank the anonymous reviewers for their helpful comments. This work is also supported by the National Natural Science Foundation of China (61631001, 61631002).

### References

[Amit et al., 2009] Amit T, Ben-Zion B, and El-Yaniv R. Robust low-rank matrix approximation. *Journal of Machine Learning Research*, 10:803–826, 2009.

[Arora et al., 2008] Arora, S., Ge, R., and Valiant, G. Learning in the presence of outliers. *Machine Learning*, 73(3):243–272, 2008.

[Bartlett et al., 1996] Bartlett, P., Bredensteiner, A., and Friedman, J. PCA on a set of functions. In *Proceedings of the International Conference on Ordinal and Symbolic Data Analysis*, pages 359–368, 1996.

[Cai et al., 2010] Cai, F., Candes, E., and Sapiro, G. A fast algorithm for robust principal component analysis. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[Cannelloni et al., 2009] Cannelloni, R., Eidelson, J., and Bredensteiner, A. Foundations of Computational Mathematics, 9(6):717–772, 2009.

[Crispian, 1998] Crispian, P. Proceedings in Computational Statistics, 245–250, 1998.

[Davenport et al., 2009] Davenport, D., and Sapiro, G. Efficient algorithms for robust principal component analysis. *Journal of Machine Learning Research*, 10(D):2899–2934, 2009.

[Ding et al., 2012] Ding, M., Dhillon, I., and Han, J. J. Matrix L<sub>1,2</sub> norm minimization for robust principal component analysis. In *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics*, pages 327–336, 2012.

[Hale, 2008] Hale, J. Proceedings of the 8th Latin American Conference on Theoretical Informatics, 306–316, 2008.

[Hale et al., 2014] Hale, J., and P. A. Proceedings of The 31st International Conference on Machine Learning, 575–583, 2014.

[Jain et al., 2010] Jain, A., and S. V. Proceedings of the 27th International Conference on Machine Learning, 471–478, 2010.

[Jain, 2013] Jain, A. Proceedings of the 30th International Conference on Machine Learning, 427–435, 2013.

[Jain, 2009] Jain, A. Proceedings of the 26th Annual International Conference on Machine Learning, 457–464, 2009.

[Kane et al., 2005] Kane, T., and R. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 739–746, 2005.

[Liu, 2012] Liu, G., and L. Mathematical Programming, 133:365–397, 2012.

[Liu et al., 2014] Liu, Q., and C. Computational Optimization and Applications, 58(2):455–482, 2014.

[Mishra et al., 2012] Mishra, M., and T. Advances in Neural Information Processing Systems (NIPS), 503–511, 2012.

[Nesterov et al., 1982] Nesterov, Y., and D. Problem complexity and method efficiency in optimization. *Wiley*, 1982.

[Nesterov, 2004] Nesterov, Y. Introductory lectures on convex optimization: a basic course, 87 Applied optimization. Kluwer Academic Publishers, 2004.

[Nesterov, 2013] Nesterov, Y. Mathematical Programming, 140(1):125–161, 2013.

[Pillai et al., 2010] Pillai, K., and P. SIAM Journal on Optimization, 20(6):3465–3489, 2010.

[Rajaraman et al., 2005] Rajaraman, J., and N. Proceedings of the 22nd International Conference on Machine Learning, 713–719, 2005.

[Rajaraman et al., 2012] Rajaraman, J., and S. Proceedings of the 29th International Conference on Machine Learning, 1351–1358, 2012.

[Sivaraman et al., 2011] Sivaraman, S., and A. Proceedings of the 28th International Conference on Machine Learning, 329–336, 2011.

[Sridharan et al., 2013] Sridharan, M., and V. Proceedings of the IEEE Conference on Decision and Control, 2013.

[Sridharan et al., 2005] Sridharan, N., and J. Advances in Neural Information Processing Systems 17, 1329–1336, 2005.

[Tibshirani, 2010] Tibshirani, R. Pacific Journal of Optimization, 6(615–640):15, 2010.