Adaptive Subspace Clustering for Matrix Completion

Takuto Wada*, Ryohei Sasaki†, and Katsumi Konishi*
* Hosei University, Tokyo
E-mail: takuto.wada.5w@stu.hosei.ac.jp, konishi@hosei.ac.jp Tel/Fax: +81-332649240
† Tokyo University of Technology, Tokyo
E-mail: sasakirh@stf.teu.ac.jp Tel/Fax: +81-426372111

***Abstract*—This paper deals with subspace clustering for a matrix completion, which is a problem of estimating missing entries in a matrix under the assumption that row or column vectors belongs to multiple low-dimensional linear spaces. Various methods for the problem have been proposed. Some of them assume that row or column vectors in a matrix can be divided into several clusters where vectors span a low-dimensional linear space and provide mathematical optimization techniques which divide a matrix into several smaller low rank matrices. However, the performance of these approaches depends on the initial values and conditions, and the accuracy of a matrix completion becomes worse when each subspace is not strictly low-dimensional due to noise. In order to improve the accuracy of a matrix completion, this paper proposes a method to reduce the dependence of initial values and to enhance robustness against noise. Numerical examples show that the proposed method achieves higher accuracy in subspace clustering and matrix completion compared to conventional methods.**

I. Introduction

This paper deals with subspace clustering for matrix completion, which is a problem of estimating missing entries in a matrix under the assumption that row or column vectors belongs to multiple low-dimensional linear spaces. While low-rank matrix approaches [1], [2], [3] have been proposed for a matrix completion, these approaches are effective for the case that row or column vectors belong to a single low-dimensional linear space and do not work well when they belong to multiple low-dimensional linear spaces. Since a matrix completion with multiple low-dimensional linear spaces has more practical applications, this paper deals with a matrix completion method with subspace clustering. Recently, a matrix completion with subspace clustering has been used in various signal processing applications, including speech restoration [4] and image restoration [5].

Some mathematical optimization-based methods have been proposed for a matrix completion with subspace clustering [5], [6]. In [6] kernel method is applied and estimates missing entries in a matrix by minimizing the rank of the kernel matrix. This method works well when both the dimensions of each subspace and the degree of a monomial are small, and requires a huge number of data (number of row or columns in a matrix) to achieve high completion accuracy. In [5] a matrix completion with subspace clustering is formulated as a problem of finding combinations of row or column vectors to divide a given matrix into several low-rank submatrices with estimating missing entries. An alternating optimization method is proposed to solve this problem and works well when a matrix is correctly divided into submatrices. However, its performance depends on the initial values and noise.

In order to improve the accuracy of a matrix completion, this paper proposes a method to reduce the dependence of initial values and to enhance robustness against noise. Numerical examples show that the proposed method achieves higher accuracy in subspace clustering and matrix completion compared to existing methods [5].

II. Subspace Clustering

We consider subspace clustering which a problem of finding L low-dimensional linear subspaces consisting of column vectors in a matrix $X \in \mathbb{R}^{M\times N}$. Takahashi et al. [5] formulated this problem as follows using the truncated nuclear norm [7],

$$\begin{aligned} \text{Find} \quad & \left\{D^{(i)}\right\}_{i=1}^{L} \\ \text{subject to} \quad & \left\|XD^{(i)}\right\|_{*,r_i} \le \varepsilon_i, i = 1,2,\dots,L, \\ & \sum_{i=1}^{L} D^{(i)} = I, \\ & (D^{(i)})_{j,j} \in \{0,1\}, \end{aligned} \tag{1}$$

where I is the identity matrix, $D^{(i)} \in [0,1]^{N\times N}$ denotes diagonal matrices, and $\|\cdot\|_{*,r}$ represents the truncated nuclear norm, which is the sum of the first (largest) singular value to the rth singular value of a matrix. $D^{(i)}$ has diagonal elements that are 0 or 1, and their sum equals the identity matrix. Using such $D^{(i)}$, $XD^{(i)}$ becomes a matrix where the columns of X corresponding to the indices where the diagonal elements of $D^{(i)}$ are 0 become zero vectors. Consequently, $XD^{(i)}$ represents a matrix that retains only the elements of the original column vectors corresponding to the indices where the diagonal elements of $D^{(i)}$ are 1, generating pseudo-submatrices corresponding to each subspace. The problem (1) describes the subspace clustering problem by imposing a low-rank constraint on these pseudo-submatrices $XD^{(i)}$ using the truncated nuclear norm. If the constant ε_i is 0, it equals to a strict low-rank constraint where the rank of $XD^{(i)}$ equal to r_i. However, in practical applications, even if $D^{(i)}$ is ideally obtained, $XD^{(i)}$ is rarely exact low-rank. Therefore Takahashi et al. [5] formulated the problem to perform low-rank approximation of $XD^{(i)}$ by setting ε_i to accommodate noise in the data. However, it is difficult to find $D^{(i)}$ such that the truncated nuclear norm of $XD^{(i)}$ is less than a certain value. Therefore they relaxed this problem by formulating it

as an objective function:

$$\begin{aligned} \text{Minimize} \quad & \sum_{i=1}^{L} \|XD^{(i)}\|_{*,r_i} \\ \text{subject to} \quad & \sum_{i=1}^{L} D^{(i)} = I, \\ & (D^{(i)})_{j,j} \in \{0,1\}. \end{aligned} \tag{2}$$

However, there are two issues in obtaining a solution of this problem. The first issue arises from the constraints $(D^{(i)})_{j,j} \in \{0,1\}$, that is, the problem is a $0-1$ combinatorial optimization. The second issue is related to the minimization of the truncated nuclear norm. Even if $(D^{(i)})_{j,j}$s take continuous values, there is no practical method to minimize $\|XD^{(i)}\|_{r_i}$. Hence, based on the idea in [7], Takahashi et al. [5] formulated the following optimization problem using auxiliary variables $Z^{(i)}$:

$$\begin{aligned} \text{Minimize} \quad & f\left(X, \{Z^{(i)}\}_{i=1}^{L}, \{D^{(i)}\}_{i=1}^{L}\right) \\ \text{subject to} \quad & \sum_{i=1}^{L} D^{(i)} = I, \\ & (D^{(i)})_{j,j} \in [0,1]. \end{aligned} \tag{3}$$

where, the objective function f is given by

$$\begin{aligned} & f\left(X, \{Z^{(i)}\}_{i=1}^{L}, \{D^{(i)}\}_{i=1}^{L}\right) \\ & = \sum_{i=1}^{L} \left(\frac{1}{2}\|Z^{(i)} - XD^{(i)}\|_F^2 + \lambda \|Z^{(i)}\|_{*,r_i} \right), \end{aligned} \tag{4}$$

where a tehnique of the alternating direction method of multiplier (ADMM) is utilized, and its solution can be obtained through alternating optimization of $D^{(i)}$ and $Z^{(i)}$. Takahashi et al. [5] finally proposed the following subspace clustering method based on the iterative method of mathematical optimization as the minimization algorithm:

$$Z^{(i)} \leftarrow \mathcal{S}_{r_i}(XD^{(i)}) \text{ for } i = 1, \cdots, L, \tag{5}$$

$$\begin{aligned} (D^{(i)})_{j,j} \leftarrow & \frac{1}{L}\left(1 - \sum_{l=1}^{L} \frac{\langle \boldsymbol{x}_j, \boldsymbol{z}_j^{(l)} \rangle}{\langle \boldsymbol{x}_j, \boldsymbol{x}_j^{(l)} \rangle}\right) + \frac{\langle \boldsymbol{x}_j, \boldsymbol{z}_j^{(i)} \rangle}{\langle \boldsymbol{x}_j, \boldsymbol{x}_j \rangle} \\ & \text{for } i = 1, \cdots, L, \quad j = 1, \cdots, N, \end{aligned} \tag{6}$$

where $\langle , \rangle$ denotes the inner product of vectors, and $\boldsymbol{x}_j$ and $\boldsymbol{z}_j^{(i)}$ represent the j-th column vectors of X and $Z^{(i)}$, respectively. $\mathcal{S}_{r,\lambda}$ denotes a soft thresholding operator that decreases the singular values which are smaller than rth singular value by λ [7]. This thresholding operator gives $Z^{(i)}$ which minimizes the objective function for the given $D^{(i)}$. Subsequently, the above procedure provides $(D^{(i)})_{j,j}$ using the cosine similarity $\langle \boldsymbol{x}_j, \boldsymbol{z}_j^{(i)} \rangle / \langle \boldsymbol{x}_j, \boldsymbol{x}_j \rangle$ of (6). The closer the cosine similarity between the reduced non-principal components $\boldsymbol{z}_j^{(i)}$ and the original vector $\boldsymbol{x}_j$ is to 1, the closer $(D^{(i)})_{j,j}$ is to 1. In other words, $XD^{(i)}$ obtained through appropriate partitioning tends to have its non-principal components close to 0, making this cosine similarity close to 1. Equation (6) also finds $D^{(i)}$ that minimizes the objective function similarly to (5), and alternating these operations always leads to a quasi-optimal solution.

Algorithm 1 Matrix completion based on subspace clustering using the truncated nuclear norm [5]

Require: $X, \Omega, L, \alpha, \eta_\alpha, \gamma, t_{max}$

1: $t \leftarrow 0$
2: **repeat**
3: $\quad \alpha \leftarrow \alpha / \eta_\alpha$
4: $\quad t \leftarrow t + 1$
5: $\quad$ **for** $i = 1$ to L **do**
6: $\quad\quad [U, S, V] \leftarrow \text{svd}(XD^{(i)})$
7: $\quad\quad r \leftarrow \underset{\hat{r}}{\text{argmax}}\, S_{\hat{r},\hat{r}}$ s.t. $S_{\hat{r},\hat{r}} > \alpha S_{1,1}$
8: $\quad\quad S_{j,j} \leftarrow \min\left(S_{j,j} - \gamma S_{r,r}, 0\right)$ for $j > r$
9: $\quad\quad Z^{(i)} \leftarrow USV^T$
10: $\quad$ **end for**
11: $\quad$ **for** $i = 1$ to L, $j = 1$ to N **do**
12: $\quad\quad (D^{(i)})_{j,j} \leftarrow \frac{1}{L}\left(1 - \sum_{l=1}^{L} \frac{\langle \boldsymbol{x}_j, \boldsymbol{z}_j^{(l)} \rangle}{\langle \boldsymbol{x}_j, \boldsymbol{x}_j^{(l)} \rangle}\right) + \frac{\langle \boldsymbol{x}_j, \boldsymbol{z}_j^{(i)} \rangle}{\langle \boldsymbol{x}_j, \boldsymbol{x}_j \rangle}$
13: $\quad$ **end for**
14: $\quad X \leftarrow \left(\sum_{i=1}^{L} Z^{(i)} D^{(i)}\right)\left(\sum_{l=1}^{L} {D^{(l)}}^2\right)^{-1}$
15: $\quad x_{m,n} \leftarrow x^*_{m,n}$ for $(m,n) \in \Omega$
16: **until** $t_{max} < t$

Ensure: X.

Next we consider a matrix completion with subspace clustering. For matrix completion, we can modify the above method by adding the constraint $x_{m,n} = x^*_{m,n}$ for $(m,n) \in \Omega$ to (3), where Ω denotes the index set of known elements, and $x^*_{m,n}$ represents the known elements. As an optimization method for the problem with the additional constraints, Takahashi et al. [5] proposed an optimization method for X using the $Z^{(i)}$ and $D^{(i)}$ obtained from (5) and (6). For the optimization of X, a simple quadratic optimization can be performed using the following update formula:

$$\begin{aligned} X \quad & \leftarrow \quad \left(\sum_{i=1}^{L} Z^{(i)} D^{(i)}\right)\left(\sum_{l=1}^{L} {D^{(l)}}^2\right)^{-1} \\ x_{m,n} \quad & \leftarrow \quad x^*_{m,n} \quad \text{for } (m,n) \in \Omega \end{aligned} \tag{7}$$

Finally, Algorithm 1 is obtained. Note that α and η_α in Algorithm 1 represent parameters for estimating the matrix rank r in line 7. Details are written in [7]. While this works well, the clustering accuracy deteriorates if the initial values of $D^{(i)}$ are not appropriate, For example, if $D^{(i)}_{j,j}$ is set to 0, $\boldsymbol{x}_j D^{(i)}_{j,j}$ becomes a zero vector, and $\boldsymbol{z}_j^{(i)}$ is also calculated as a zero vector, resulting in a cosine similarity of 0 regardless

of whether $\boldsymbol{x}_j$ belongs to the i-th cluster. Therefore, this algorithm depends on the initial values of $D^{(i)}$. Furthermore, numerical examples show that the algorithm often fails to estimate $D^{(i)}$ if there exists noise in X.

III. Proposed Method

This section proposes a new subspace clustering method. The original idea of subspace clustering is to find the subspaces related to each cluster and the data belonging to those subspaces. In the method by Takahashi et al. [5], auxiliary variables $Z^{(i)}$ are used to represent these subspaces. Ideally, the column space of $Z^{(i)}$ matches each target subspace. However, as mentioned above, the deterioration of clustering accuracy derives from the use of auxiliary variables $Z^{(i)}$. Therefore, this paper formulates the following optimization problem using auxiliary variables $U^{(i)} \in \mathbb{R}^{M \times r}$ that can directly represent the column space to represent the subspaces.

$$\begin{aligned} \text{Minimize} \quad & g\left(X, \{D^{(i)}\}_{i=1}^{L}, \{U^{(i)}\}_{i=1}^{L}\right) \\ \text{subject to} \quad & \sum_{i=1}^{L} D^{(i)} = I \\ & D^{(i)} \in \{0,1\} \\ & {U^{(i)}}^T U^{(i)} = I \end{aligned} \tag{8}$$

$$\begin{aligned} & g\left(X, \{D^{(i)}\}_{i=1}^{L}, \{U^{(i)}\}_{i=1}^{L}\right) \\ =& \sum_{i=1}^{L} \sum_{j=1}^{N} D_{j,j}^{(i)} \left\| \left(I - U^{(i)} {U^{(i)}}^T\right) \boldsymbol{x}_j \right\|_2^2 \\ =& \sum_{i=1}^{L} \left\| \left(I - U^{(i)} {U^{(i)}}^T\right) X D^{(i)} \right\|_F^2 \end{aligned} \tag{9}$$

The idea of minimizing the Frobenius norm to minimize the matrix rank is originally proposed in [3], and minimizing the objective function g is equivalent to minimizing the rank of the submatrix $XD^{(i)}$. A solution of this problem can be obtained by alternating optimization as follows,

$$\begin{aligned} [\tilde{U}, \tilde{S}, \tilde{V}] &\leftarrow \text{svd}(XD^{(i)}) \\ U^{(i)} &\leftarrow \tilde{U}_{:,1:r} \\ & \quad \text{for } i = 1, \cdots, L \end{aligned} \tag{10}$$

$$(D^{(i)})_{j,j} \leftarrow \begin{cases} 1 & \text{if } i = \underset{\hat{i}}{\text{argmin}} \left\| \left(I - U^{(\hat{i})} {U^{(\hat{i})}}^T\right) \boldsymbol{x}_j \right\|_2^2 \\ 0 & \text{otherwise} \end{cases} \quad \text{for } j = 1, \cdots, N \tag{11}$$

where $A_{:,1:r}$ denotes the matrix formed by the first to r-th columns of the matrix A. The singular values $\tilde{S}_{j,j}$ obtained by the singular value decomposition $\text{svd}(XD^{(i)})$ are assumed to be arranged in descending order. The above equations are methods for alternately solving for $U^{(i)}$ and $D^{(i)}$. Let us consider the problem of minimizing (9) with respect to $U^{(i)}$ for given $D^{(i)}$. Based on classical theorems concerning matrices [8], the optimal solution can be obtained through singular value decomposition of the matrix. Let $XD^{(i)}$ be decomposed as $XD^{(i)} = \tilde{U}\tilde{S}\tilde{V}^T$, then the optimal solution $U^{(i)}$ is equal to the matrix formed by the column vectors corresponding to the first to r-th principal components of $\tilde{U}$. Next we consider the problem of minimizing (9) with respect to $D^{(i)}$ for given $U^{(i)}$. This problem can be performed independently for each j, and the minimum solution is obtained by setting $(D^{(i)})_{j,j} = 1$ for the i that minimizes $\|(I - U^{(i)}{U^{(i)}}^T)\boldsymbol{x}_j\|_2^2$ and 0 otherwise. By repeating the above two schemes until $D^{(i)}$ converges, a solution can be obtained. In practice, since r is unknown, we propose a method that starts with $r = 1$ and iteratively repeats the above schemes. Finally this paper proposes Algorithm 2. An advantage of Algorithm 2 is that it requires fewer parameters to be specified compared to existing methods.

Algorithm 2 Adaptive subspace reconstruction algorithm (proposed method)

Require: X, L, t_{max}

for $r = 1$ to $M - 1$ **do**
  $t \leftarrow 0$
  repeat
    $t \leftarrow t + 1$
    $D_{old}^{(i)} \leftarrow D^{(i)}$ for $i = 1, \cdots, L$
    for $i = 1$ to L **do**
      $[\tilde{U}, \tilde{S}, \tilde{V}] \leftarrow \text{svd}(XD^{(i)})$
      $U^{(i)} \leftarrow \tilde{U}_{:,1:r}$
    end for
    for $i = 1$ to L, $j = 1$ to N **do**
      $(D^{(i)})_{j,j} \leftarrow 1$
      if $i = \underset{\hat{i}}{\text{argmin}} \left\| \left(I - U^{(\hat{i})} {U^{(\hat{i})}}^T\right) \boldsymbol{x}_j \right\|_2^2$
      0 otherwise
    end for
  until $t_{max} < t$ or $D^{(i)} = D_{old}^{(i)}$
end for

Ensure: $\{D^{(i)}\}_{i=1}^{L}$.

As described in section 2, we consider performing matrix completion using subspace clustering with Algorithm 2. If $D^{(i)}$ is appropriately obtained by Algorithm 2, the matrix completion problem can be solved by imposing the constraint $x_{m,n} = x_{m,n}^*$ for $(m,n) \in \Omega$ on (8), and minimizing (9) with respect to X, which performs matrix rank minimization within each cluster. Based on this idea, we present Algorithm 3 for matrix completion with subspace clustering, where the truncated nuclear norm minimization is applied. In this algorithm, it is not necessary to update $D^{(i)}$ if there is no significant change in X, and therefore the algorithm executes Algorithm 2 only when there is a significant change in X.

Algorithm 3 Matrix completion using adaptive subspace reconstruction algorithm (proposed method)

Require: $X, \Omega, L, \alpha, \eta_\alpha, \gamma, t_{max}$

$t \leftarrow 0$
$r_{max} \leftarrow 1, r_{max\,old} \leftarrow 0$
repeat
  $t \leftarrow t + 1$
  if $r_{max} \neq r_{max\,old}$ **then**
    obtain $\{D^{(i)}\}_{i=1}^{L}$ using Algorithm 2
  end if
  for $i = 1$ to L **do**
    $[U, S, V] \leftarrow \mathrm{svd}(XD^{(i)})$
    $r_i \leftarrow \underset{\hat{r}}{\mathrm{argmax}}\, S_{\hat{r},\hat{r}}$ s.t. $S_{\hat{r},\hat{r}} > \alpha S_{1,1}$
    $S_{j,j} \leftarrow \min\left(S_{j,j} - \gamma S_{r_i,r_i}, 0\right)$ for $j > r_i$
    $Z^{(i)} \leftarrow USV^T$
  end for
  $X \leftarrow \sum_{i=1}^{L} Z^{(i)}$
  $x_{m,n} \leftarrow x^*_{m,n}$ for $(m,n) \in \Omega$
  $r_{max\,old} \leftarrow r_{max}$
  $r_{max} \leftarrow \max r_i$
until $t_{max} < t$

Ensure: X.

IV. Numerical Examples

This section demonstrates the effectiveness of the proposed algorithm by comparing with Algorithm 1. In order to confirm how the estimation accuracy changes with the matrix rank, we apply algorithms with matrix ranks $r = 10, 11, \cdots, 15$. In all experiments, matrix size is set to 150×150, and the total number of clusters is set to 50.

First, we compare the estimation accuracy of subspace clustering. Assuming that the true matrix X is given, we examine performances of subspace clustering for matrices with no missing entries. White noise $SNR = 20$ is added to each column vector of the matrix, and Algorithm 1 and Algorithm 2 are applied. The maximum number of iterations t_{max} for both alfgorithms is set to $t_{max} = 1,000$, and the estimation accuracy of $D^{(i)}$ is compared. Since the $D^{(i)}$ obtained by Algorithm 1 can take values in $[0, 1]$, we calculate the confusion matrix C with respect to the true value $D^{(i)}_{true}$ as $C_{k,l} = \sum_{j=1}^{N} (D^{(k)})_{j,j} (D^{(l)}_{true})_{j,j}$, and compare the inverse condition number of this matrix. If $D^{(i)}$ has even one different entry from an entry of $D^{(i)}_{true}$, the inverse condition number of C will be 0.972 or less. Table 1 shows the average of the inverse condition numbers of C obtained over 100 trials for each method. From this table, it can be seen that the proposed method performs subspace clustering with sufficient accuracy for noise existing case.

Next we compare Algorithm 1 with Algorithm 3 for matrix completion with subspace clustering ith a missing rate of 50%. Similar to Table 1, we conducted 100 trials and calculated the average normalized mean square error (NMSE) between the matrix X obtained by each algorithm and the true value X^*.

$$NMSE = \sum_{(m,n)\notin\Omega} (x_{m,n} - x^*_{m,n})^2 \;/\; \sum_{(m,n)\notin\Omega} (x^*_{m,n})^2$$

Figure 1 shows the NMSE of each algorithm. As can be seen from the figure, Algorithm 1 gives a significant deterioration in completion accuracy for $r \geq 12$, while the proposed method maintains an NMSE of less than 0.1 for all ranks. These results demonstrate the effectiveness of the proposed method.

TABLE I
Average of the inverse condition numbers of confusion matrix C obtained by subspace clustering in 100 trials

Method \ rank r	10	11	12	13	14	15
Algorithm 1 [5]	0.16	0.19	0.22	0.18	0.19	0.20
Algorithm 2	1.00	1.00	1.00	1.00	0.98	0.97

V. Conclusions

This paper deals with subspace clustering for matrix completion. One of the existing methods is the subspace clustering method using an optimization problem with the truncated nuclear norm. This method is based on mathematical optimization and has the advantage of guaranteeing a quasi-optimal solution. However, due to the nature of the algorithm, it tends to depend on initial values and is strongly affected by noise, which can result in degraded matrix completion accuracy. This paper proposes a new subspace clustering method using auxiliary variables representing each subspace and apply it to matrix completion. Numerical experiments demonstrated that the proposed method showed higher estimation accuracy compared to existing methods.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number JP24K14845.

References

[1] J. F. Cai, E. J. Candés, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optimization*, vol.20, no.4, pp.1956–1982, 2010.

[2] K. Mohan and M. Fazel, "Iterative reweighted algorithms for matrix rank minimization," *J. Machine Learning Research (JMLR)*, vol.13, no.1, pp.3441–3473, 2012.

[3] K. Konishi, "Parallel GPU implementation of null space based alternating optimization algorithm for large-scale matrix rank minimization," *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing.*, pp.3689-3692, 2012.

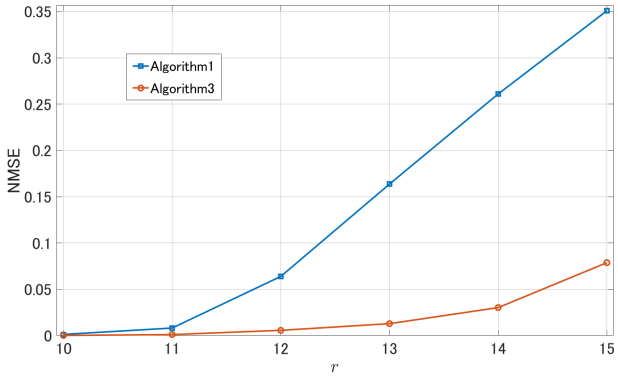

Fig. 1. Average NMSE comparison of each method

[4] R. Sasaki, K. Konishi, T. Takahashi, and T. Furukawa, "Multiple matrix rank minimization approach to audio declipping," *IEICE Transactions on Information and Systems*, vol.E101-D, no.3, pp.821–825, 2018.

[5] T. Takahashi, K. Konishi, K. Uruma, and T. Furukawa: Multiple Subspace Model and Image-Inpainting Algorithm Based on Multiple Matrix Rank Minimization, *IEICE Trans. on Info. and Syst.*, vol.E103.D, no.12, pp.2682–2692, 2020

[6] G. Ongie, R. Willett, R. D. Nowak, and L. Balzano, "Algebraic variety models for high-rank matrix completion," *Proc. of the 34th Int. Conf. on Machine Learning (PMLR)*, vol.70, 2691-2700 (2017)

[7] K. Konishi, K. Uruma, T. Takahashi, and T. Furukawa, "Iterative partial matrix shrinkage algorithm for matrix rank minimization," *Signal Processing*, vol.100, pp.124–131, 2014.

[8] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol.1, pp.211–218, 1936.