

Direct Robust Matrix Factorization for Anomaly Detection

Liang Xiong
Machine Learning Department
Carnegie Mellon University
lxiong@cs.cmu.edu

Xi Chen
Machine Learning Department
Carnegie Mellon University
xichen@cs.cmu.edu

Jeff Schneider
Robotics Institute
Carnegie Mellon University
schneide@cs.cmu.edu

Abstract—Matrix factorization methods are extremely useful in many data mining tasks, yet their performances are often degraded by outliers. In this paper, we propose a novel robust matrix factorization algorithm that is insensitive to outliers. We directly formulate robust factorization as a matrix approximation problem with constraints on the rank of the matrix and the cardinality of the outlier set. Then, unlike existing methods that resort to convex relaxations, we solve this problem directly and efficiently. In addition, structural knowledge about the outliers can be incorporated to find outliers more effectively. We applied this method in anomaly detection tasks on various data sets. Empirical results show that this new algorithm is effective in robust modeling and anomaly detection, and our direct solution achieves superior performance over the state-of-the-art methods based on the L_1 -norm and the nuclear norm of matrices.

Keywords-matrix factorization, robust, anomaly detection

I. INTRODUCTION

Real world problems almost always involve data that do not conform to the assumptions we made in our models. These data are called outliers or anomalies. These outliers can severely degrade the models' quality and performances, therefore we want robust methods to reduce the impact of outliers. In novelty detection problems, we are also interested in finding and studying these outliers since they might lead to discoveries. To do this, we also need reliable models that are not distorted by outliers.

The definition of outlier varies depending on the application and the behavior of data we want to capture. A popular assumption is that the normal data are close together, and consequently outliers are far away from the others *i.e.* lie in the low-density region of the data distribution [2], [30]. For a survey of the outlier detection field readers can refer to [5]. In this paper, we consider another common definition called the *subspace outlier*, which comes from the assumptions that the normal data reside in a low-dimensional linear subspace, and the outliers are outside of this subspace. This means, for example in signal processing, that a normal signal can be reconstructed by a few bases. If a signal cannot be well reconstructed by these bases, it is an outlier. This subspace-based modeling is widely used in various problems such as dimensionality reduction, signal/image processing, time series analysis, and collaborative filtering.

Matrix factorization techniques, such as *principal component analysis* (PCA) and *non-negative matrix factorization*

(NMF) [16], are extremely useful in learning subspace structures from data. However, traditional methods are prone to be distorted by outliers [13]. Since factorizations are usually done by minimizing the error made by the model, a popular way of achieving robustness is to use error measurements that are insensitive to outliers. Though being pervasively used, the *mean squared error* or the L_2 error measure is known to be vulnerable to outliers [27]. In machine learning and statistics, the L_1 error measure (mean absolute error) is widely used for the purpose of robustness [1], [3]. Other measures like the *Huber loss* [11] and the *Geman-McClure* function have also been employed [14], [23]. These robust measurements usually increase the algorithms' complexities significantly. Another strategy is to exclude the outliers: we can first guess which data are outliers, and then reduce their influences to the model [13], [27].

The contribution of this paper is to propose a novel algorithm for learning robust subspace models based on matrix factorization. For a data matrix \mathbf{X} , we assume that it is approximately low-rank, and a small portion of this matrix has been corrupted by some arbitrary outliers. The goal of the proposed algorithm is to get a reliable estimation of the true low-rank structure of this matrix. To achieve this, our basic idea is to exclude the outliers from the model estimation. Specifically, the proposed algorithm directly answers the question: if you are allowed to ignore some data (outliers), what is the best low-rank model you can get?

We formulate this problem as a constrained optimization problem. This formulation aims at minimizing the L_2 error of the low-rank approximation subject to that the number of ignored outliers is small, without any further assumptions. This formulation reflects our direct understanding of outliers and robust estimation. Thus we call it *direct robust matrix factorization* (DRMF).

It can be shown that DRMF is the original problem that the recently popular *nuclear norm* based methods (*e.g.* [3], [28]) are trying to solve. However, unlike these methods that resort to relaxation techniques, we directly form these constraints in terms of the *matrix rank* and the *cardinality of the outlier set*. Despite that matrix rank and set cardinality are often very difficult to handle in optimization, we are able to solve this problem directly in its original form. We observe that better quality results are produced by this direct solution compared to the relaxed methods.

We adopt *block coordinate descent* to solve the DRMF problem. The resulting algorithm is based on existing factorization routines such as the *singular value decomposition* (SVD), and efficient thresholding procedures. Therefore DRMF is simple to implement, efficient, and easy to use. DRMF is also very flexible: we can impose additional constraints on both the factorization (*e.g.* nonnegative factors [16]) and the outliers (*e.g.* outlier columns instead of entries [28]) to incorporate knowledge for better performance.

We applied DRMF to both synthetic and real-world data sets for the purpose of robust modeling and anomaly detection. We compare DRMF to its state-of-the-art competitors based on the nuclear norm and the L_1 error measurement. Based on extensive empirical results we conclude that DRMF is able to get better performance than these relaxed methods. In addition, the parameters of DRMF are intuitive and easy to tune, making it a practical tool for robust analysis.

The rest of this paper is structured as follows. We introduce background and notations in Section II. Section III describes the proposed algorithm. Related work and discussions are in Section IV and V. Experiments are presented in section VI. Finally we make our conclusions.

II. BACKGROUND AND NOTATION

Matrices are very useful in representing data. For example, in regression and classification, samples are often organized into a *design matrix* in which each row represents a sample and each column represents a feature. *Connectivity matrices* are widely used to express network and graph data. We denote a $m \times n$ data matrix as $\mathbf{X} \in \mathbb{R}^{m \times n}$. $X_{i,j}$ denotes the (i, j) -th entry of \mathbf{X} . For submatrices, we use the Matlab notation, *e.g.* $\mathbf{X}_{1:100}$, denotes the first 100 rows of \mathbf{X} . We also use the operator $\mathcal{D}_l(\cdot)$ to return an $l \times l$ diagonal matrix whose diagonal is the input vector.

One of the most common analysis we can do on \mathbf{X} is factorization, as in *principal component analysis* (PCA). We assume that \mathbf{X} has a low rank and can be factorized as

$$\mathbf{X} \approx \mathbf{U}\mathbf{V}^T, \mathbf{U} \in \mathbb{R}^{m \times K}, \mathbf{V} \in \mathbb{R}^{n \times K}, \quad (1)$$

where K is the rank of the factorization. For design matrices, factors given by PCA/SVD reveals the linear structure and intrinsic dimensionality of the data. The low-rank assumption is also useful in *matrix completion* [4], [22] and *collaborative filtering* [25], [26].

In a more general form, low-rank matrix factorization can be written as the following optimization problem

$$\begin{aligned} \min_{\mathbf{L}} \quad & \|\mathbf{X} - \mathbf{L}\|_F \\ \text{s.t.} \quad & \text{rank}(\mathbf{L}) \leq K, \end{aligned} \quad (2)$$

where $\|\cdot\|_F$ is the *Frobenius norm*, \mathbf{L} is the low-rank approximation to \mathbf{X} , and K is the maximal rank of \mathbf{L} .

Singular value decomposition (SVD) is perhaps the most commonly used tool for low-rank analysis. SVD decomposes

a matrix into three factors:

$$\mathbf{X} = \mathbf{U}\mathcal{D}(\mathbf{s})\mathbf{V}^T = \sum_{i=1}^l s_i \mathbf{u}_i \mathbf{v}_i^T, \quad (3)$$

where $l = \min(m, n)$, $\mathbf{s} = [s_1, \dots, s_l]$ is the vector of singular values of \mathbf{X} in the descending order, columns of $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_l] \in \mathbb{R}^{m \times l}$ and $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_l] \in \mathbb{R}^{n \times l}$ are the corresponding the left and right singular vectors. The significance of SVD is reflected in the following theorem [8]:

Theorem 1 (Eckart-Young): Let the SVD of \mathbf{X} be (3). For any K with $0 \leq K \leq \text{rank}(\mathbf{X})$, let

$$\hat{\mathbf{L}}_K = \mathbf{U}_{:,1:K} \mathcal{D}(\mathbf{s}_{1:K}) \mathbf{V}_{:,1:K}^T = \sum_{i=1}^K s_i \mathbf{u}_i \mathbf{v}_i^T, \quad (4)$$

then

$$\|\mathbf{X} - \hat{\mathbf{L}}_K\|_F = \min_{\text{rank}(\mathbf{L}) \leq K} \|\mathbf{X} - \mathbf{L}\|_F.$$

In other words, the rank- K truncated SVD approximation $\hat{\mathbf{L}}_K$ is a globally optimal solution to problem (2).

From SVD we can derive the *nuclear norm* of matrices. The nuclear norm of matrix \mathbf{X} is defined as $\|\mathbf{X}\|_* = \sum_{i=1}^l s_i$ *i.e.* the sum of \mathbf{X} 's singular values. The nuclear norm can serve as a convex relaxation of the matrix rank, and has attracted much research interest recently. We shall discuss more in Section IV.

Next we consider robust error measurement. Let the error matrix be $\mathbf{E} = \mathbf{X} - \mathbf{L}$. In (2), we used the Frobenius norm, $\|\mathbf{E}\|_F = \sqrt{\sum_{i,j} E_{i,j}^2}$, a.k.a the L_2 -**norm**, to measure \mathbf{E} . The L_2 -norm is pervasively used but is known to be sensitive to outliers [13]. A common robust alternative is the L_1 -**norm** $\|\mathbf{E}\|_1 = \sum_{i,j} |E_{i,j}|$ [1], [3], in which the errors are not squared so the impact of large errors is reduced. A more aggressive choice is the L_0 -**norm**¹ $\|\mathbf{E}\|_0 = \sum_{i,j} I(E_{i,j} \neq 0)$, where $I(\cdot)$ is the indicator function. The L_0 -norm only counts the number of errors despite their magnitudes.

Recently, *structured norms* become popular in handling problems such as *group lasso* [29] and *multitask learning* [19] with structural knowledge. These norms can also be used to incorporate knowledge about the structure of outliers (*e.g.* when outliers in the same row is correlated) [28]. Here we introduce the $L_{2,1}$ -**norm** and $L_{2,0}$ -**norm**. The $L_{2,1}$ -norm $\|\mathbf{E}\|_{1,2} = \sum_{i=1}^m \|\mathbf{E}_{i,:}\|_2$ is the sum of the L_2 -norm of rows of \mathbf{E} (*i.e.* the sum of the lengths of the row vectors), and $L_{2,0}$ -norm $\|\mathbf{E}\|_{2,0} = \sum_{i=1}^m I(\|\mathbf{E}_{i,:}\|_2 > 0)$ is the number of non-zero rows in \mathbf{E} . These two norms compare similarly as the L_1 and L_2 norms except that errors are measured in groups according to the assumed structure.

III. DIRECT ROBUST FACTORIZATION

We adopt the common assumption that there is only a small amount of outliers in the data matrix \mathbf{X} . Then, we define the robust low-rank approximation of \mathbf{X} as the answer

¹Rigorously this L_0 measurement is not a norm.

to the question: *if you are allowed to ignore some data as outliers, what is the best low-rank approximation?*

A directly formulation of the above question is the following problem *direct robust matrix factorization* (DRMF):

$$\begin{aligned} \min_{\mathbf{L}, \mathbf{S}} \quad & \|(\mathbf{X} - \mathbf{S}) - \mathbf{L}\|_F \quad (5) \\ \text{s.t.} \quad & \text{rank}(\mathbf{L}) \leq K \\ & \|\mathbf{S}\|_0 \leq e, \end{aligned}$$

where \mathbf{L} is the low-rank approximation as before, K is the rank, \mathbf{S} is the matrix of *outliers*, and e is the maximal number of non-zeros entries in \mathbf{S} *i.e.* the maximal number of entries that can be ignored as outliers. Comparing DRMF to the regular problem (2), we can see that the only difference is that we allow the outliers \mathbf{S} to be excluded from the low-rank approximation, as long as the number of outliers is not too large *i.e.* \mathbf{S} is sufficiently *sparse*. Note that we do not need the actual number of outliers. Instead, we only use e to put an upper limit on it.

By excluding the outliers from the effort of low-rank approximation, we can ensure the reliability of the estimated low-rank structure. On the other hand, the number of outliers is constrained so that the estimation is still faithful to the data. DRMF is advantageous over existing methods in its simplicity and directness: no special robust error measurement is introduced, nor do we make assumptions about the outliers beyond necessity. In fact, several state-of-the-art methods are relaxed versions of DRMF, as we shall discuss in section IV.

A. Algorithm

Usually, optimization problems involving the *rank* or the L_0 -norm *i.e.* *set cardinality* are difficult to solve. Nevertheless, the DRMF problem admits a simple solution due to its decomposable structure *w.r.t.* variables \mathbf{L} and \mathbf{S} . To take advantage of this property, we adopt the *block coordinate descent* strategy, and the resulting algorithm is described in algorithm 1: We first fix \mathbf{S} the current estimate of outliers, exclude them from \mathbf{X} to get the “clean” data \mathbf{C} , and fit \mathbf{L} based on \mathbf{C} . Then, we update the outliers \mathbf{S} based on the error $\mathbf{E} = \mathbf{X} - \mathbf{L}$.

It is easy to see that the solution to the low-rank approximation problem (6) is directly given by SVD according to Theorem 1. Therefore, the solution to \mathbf{L} is simply the truncated SVD approximation to \mathbf{C} given in (4), which can be obtained efficiently. Since only the first K singular vectors are required, we can further accelerate the computation using *partial SVD* algorithms such as PROPACK [15].

The outlier detection problem (7) can also be solved efficiently. To solve the general problems of L_0 -norm constrained minimization of decomposable objectives, we give the following theorem which extends the work of [20]:

Theorem 2: Let \mathcal{A} be a domain with $0 \in \mathcal{A}$; $A = \{a_1, \dots, a_n\} \in \mathcal{A}^n$; $\{f_i | f_i : \mathcal{A} \rightarrow \mathbb{R}, i = 1, \dots, n\}$ be a set

Algorithm 1 Direct Robust Matrix Factorization (DRMF)

1) **Input:**

- \mathbf{X} the data matrix.
- K the maximal rank of the factorization.
- e the maximal number of outliers.
- \mathbf{S} the initial outliers.

2) While not converged:

a) Solve the factorization problem:

$$\begin{aligned} \mathbf{L} = \quad & \arg \min_{\mathbf{L}} \|\mathbf{C} - \mathbf{L}\|_F, \mathbf{C} = \mathbf{X} - \mathbf{S} \quad (6) \\ \text{s.t.} \quad & \text{rank}(\mathbf{L}) \leq K \end{aligned}$$

b) Solve the outlier detection problem:

$$\begin{aligned} \mathbf{S} = \quad & \arg \min_{\mathbf{S}} \|\mathbf{E} - \mathbf{S}\|_F, \mathbf{E} = \mathbf{X} - \mathbf{L} \quad (7) \\ \text{s.t.} \quad & \|\mathbf{S}\|_0 \leq e \end{aligned}$$

3) **Output:**

- \mathbf{L} the robust low-rank approximation.
 - \mathbf{S} the outliers.
-

of n functions mapping from \mathcal{A} 's elements to real numbers. Also, let $\hat{a}_i = \arg \min_{a_i} f_i(a_i)$; $b_i = f_i(0) - f_i(\hat{a}_i) \geq 0$; $\|A\|_0$ be the number of non-zero elements in A ; e be a positive integer. Then for the following problem

$$\begin{aligned} \min_A \quad & f(A) = \sum_{i=1}^n f_i(a_i) \quad (8) \\ \text{s.t.} \quad & \|A\|_0 \leq e, \end{aligned}$$

an optimal solutions is given by $A^* = \{a_1^*, \dots, a_n^*\}$ with

$$a_i^* = \begin{cases} \hat{a}_i & b_i \geq b_{(e)} \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

where $b_{(e)}$ is the e -th largest element in $\{b_i\}_{i=1, \dots, n}$.

Proof: The proof is derived similarly as in [20] and therefore omitted here. ■

Based on Theorem 2, problem (7) can easily be solved by letting $a_{i,j} = S_{i,j}$ and $f_{i,j}(S_{i,j}) = (S_{i,j} - E_{i,j})^2$. Specifically, the solution is

$$S_{i,j} = \begin{cases} E_{i,j} & b_{i,j} \geq b_{(e)} \\ 0 & \text{otherwise.} \end{cases} \quad (10)$$

where $b_{i,j} = E_{i,j}^2$ and $b_{(e)}$ is the e -th largest value in $\{b_{i,j}\}_{i=1, \dots, m, j=1, \dots, n}$. This result is very intuitive: in each round, large errors are considered outliers, and are put into \mathbf{S} to be excluded from the low-rank fitting in the next round.

The above results give the global optima to step (a) and (b) in Algorithm 1. They are guaranteed to improve the objective value within the feasible region, and thus the algorithm is going to converge. In each iteration, we do one rank- K partial SVD plus one $\frac{e}{mn}$ quantile computation, so the total complexity is $O(mn(K + \log(e)))$. Since K and e

are usually fixed and small, DRMF can handle large-scale problems.

The DRMF problem (5) is not convex due to the constraints on the rank of \mathbf{L} and the L_0 -norm of \mathbf{S} . Therefore, local minima exist depending on the starting point. This fact is reflected in that the algorithm starts with an initial guess of outliers. However, in experiments we found that DRMF is quite stable *w.r.t.* starting point, and good initialization methods exist. More details can be found in Section V-B.

DRMF has two parameters K and e that need the user’s attention. Yet, their clear meanings (the rank and the maximally allowed number of outliers) help the user select their values. Particularly, we emphasize that the value of e does not need to match the actual number of outliers. It is only used as a safeguard to ensure that not too many data are regarded as outliers. For this purpose we can easily set e to be say 5% of the whole data set. From Eq. (10) we can see that normal data with small factorization errors will not be thrown as outliers. On the other hand, if there are more than 5% outliers, the ones with largest errors will be taken care of. We will show that this default behavior gives us good performance in various situations in Section VI.

IV. RELATED WORK

Matrix factorization is widely used in data mining and machine learning, and robust subspace analysis methods are of great value in practical situations. Many robust estimators has been proposed (*e.g.* [17], [10], [12], [14], [13]). They usually involves alternative error measurements, complex estimation procedures, or problem specific heuristics. On the other hand, the DRMF algorithm is both conceptually and computationally simple: it excludes some data and fit the rest, and the solution is obtained by iteratively applying SVD and thresholding the errors.

Another limitation of traditional robust methods is that performance cannot be guaranteed in high dimensions [7], [28]. Recently, constraining the nuclear norm [4], [22] of the matrix instead of its rank becomes a popular strategy for overcoming this problem [3], [32], [28], and has been shown to outperform traditional algorithms. These methods can be summarized as the *nuclear norm minimization* (NNM) problem. To compare, we also rewrite DRMF to one of its equivalent lagrangian form, and show them in Table I.

We can immediately see the relationship between DRMF and the NNM methods: DRMF minimizes the *rank*, while NNM minimizes the *nuclear norm*; DRMF measures outliers by the L_0 -norm, while NNM uses the L_1 -norm. In fact, the nuclear norm and the L_1 -norm in the NNM problem are proposed as convex relaxations of the rank and the L_0 -norm in the first place. In this sense, DRMF is the “original problem” that NNM is trying to solve.

By using the relaxations, NNM is convex and the globally optimal solutions can be found. In addition, theories have been provided for choosing λ to guarantee the correct

NNM	$\min_{\mathbf{L}, \mathbf{S}}$ s.t.	$\ \mathbf{L}\ _* + \lambda \ \mathbf{S}\ _1$ $\ \mathbf{X} - \mathbf{L} - \mathbf{S}\ _F \leq \sigma$
DRMF	$\min_{\mathbf{L}, \mathbf{S}}$ s.t.	$\text{rank}(\mathbf{L}) + \lambda \ \mathbf{S}\ _0$ $\ \mathbf{X} - \mathbf{L} - \mathbf{S}\ _F \leq \sigma$

Table I
COMPARING THE NUCLEAR NORM MINIMIZATION (NNM) PROBLEM AND DRMF. \mathbf{L} IS LOW-RANK; \mathbf{S} IS THE SPARSE OUTLIER. $\|\cdot\|_*$ IS THE NUCLEAR NORM; σ IS THE ALLOWED APPROXIMATION ERROR.

recovery of the principal subspace under certain conditions [3], [28]. Yet, it is unknown how well these relaxations approximate the original problem in general. On the other hand, the original DRMF problem is non-convex and has the local-minima problem. As a remedy, we can initialize DRMF with the NNM results to obtain results that are better than using either NNM or DRMF alone. We expand this point further in Section V-B. The theoretical properties of DRMF are difficult to analyze due to the non-continuous and non-convex nature of the L_0 -norm and the matrix rank. Yet we shall show that DRMF can achieve better empirical performance than the relaxed NNM methods.

The NNM methods often set $\sigma = 0$ for exact recovery [3], [28]. Yet real-world noisy data invalidate this choice and make the algorithm inefficient. When NNM uses $\sigma > 0$ (*e.g.* in [32], [28]), it needs more assumptions to ensure the theoretical soundness and introduces extra parameters (*e.g.* the amount of Gaussian noise) that need careful tuning. On the other hand, DRMF can be applied in both situations, thanks to the fact that it solves the problem in the constrained form (5); the only difference between noisy and noiseless data is that the former will have non-zero objective values.

V. DISCUSSION

A. Extensions to Incorporating Prior Knowledge

In many situations, additional knowledge is available for us to find outliers. For example, in a *design matrix*, if one sample point has been corrupted, then it is very likely that most of the entries in its corresponding row are outliers. In this case, we should look for outlier rows so that evidences of anomalies can aggregated to enhance the performance. DRMF can easily be extended to handle this situation. Here, we consider the *outlier patterns* to be groups of entries that are anomalous. Instead of counting the number of outlier entries, we can count the number of outlier patterns using structured norms such as the $L_{2,0}$ -norm. Concretely, the following DRMF-Row (DRMF-R) problem handles row

outliers:

$$\begin{aligned} \min_{\mathbf{L}, \mathbf{S}} \quad & \|(\mathbf{X} - \mathbf{S}) - \mathbf{L}\|_F \quad (11) \\ \text{s.t.} \quad & \text{rank}(\mathbf{L}) \leq K \\ & \|\mathbf{S}\|_{2,0} \leq e, \end{aligned}$$

where e is the maximal number of outlier rows allowed. DRMF-R can be solved by replacing step (b) in Algorithm 1 with the following problem:

$$\begin{aligned} \mathbf{S} = \quad & \arg \min_{\mathbf{S}} \|\mathbf{E} - \mathbf{S}\|_F, \mathbf{E} = \mathbf{X} - \mathbf{L} \quad (12) \\ \text{s.t.} \quad & \|\mathbf{S}\|_{2,0} \leq e. \end{aligned}$$

Row-wise outliers has also been considered in *outlier pursuit* (OP) [28]. OP extends the NNM algorithm by using the $L_{2,1}$ -norm to capture outlier rows. Not surprisingly, OP is the convex relaxation of the DRMF-R problem (11).

Problem (12) can also be solved based on Theorem 2 by treating each row of \mathbf{S} as an element. Without giving details, we show that the solutions is:

$$\mathbf{S}_{i,:} = \begin{cases} \mathbf{E}_{i,:} & l_i \geq l_{(e)} \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (13)$$

where $l_i = \|\mathbf{E}_{i,:}\|_2$ and $l_{(e)}$ is the e -th largest value among $\{l_i\}_{i=1,\dots,m}$. Again, the solution is obtained efficiently by thresholding. In fact, it is very easy to capture arbitrarily shaped outlier patterns to accommodate specific problems.

Finally, the low-rank component of DRMF can also be extended. For example, we can require the factor matrices in (1) to be non-negativity as in *non-negative matrix factorization* (NMF) [6]. To do this, we replace the constraint $\text{rank}(\mathbf{L}) \leq K$ in (5) by the explicit factorization form $\mathbf{L} = \mathbf{U}\mathbf{V}^T$ and then impose non-negativity constraints on \mathbf{U} and \mathbf{V} . DRMF can also easily be extended to handle missing values in collaborative filtering. Fast and pass-efficient algorithms such as [24] can also be integrated into DRMF to do robust analysis on massive data sets.

B. Implementation

When applying DRMF, we need to answer several important practical questions: how to choose the parameters e the maximal number of outliers allowed, K the rank of the factorization, and the starting point *i.e.* the initial guess of outliers \mathbf{S} . As discussed in Section III-A, we can set e to be *e.g.* 5% of the whole data set so that the algorithm is not ignoring to much data.

Like most matrix factorization methods, in DRMF the rank of the factorization K is selected according to prior knowledge, cross-validation, or other heuristics. For example, we can observe the singular values of the data matrix, and choose a K to preserve certain amount of data variability. In some situations, the value of K is constrained by available computational resources, so we have to make trade-offs between accuracy and running time.

The initial guess of outliers \mathbf{S} affects the final solution, since DRMF is non-convex and can be trapped in local minima. For many moderate situations we found that the simple choice of $\mathbf{S} = \mathbf{0}$ works well. But in extreme cases where the regular SVD is completely disrupted by outliers, this simple heuristic would lead DRMF into irrecoverable local minima. One such example is shown in Figure 1.



Figure 1. An example where DRMF with initial $\mathbf{S} = \mathbf{0}$ would fail. Blue crosses are normal points and the red circle is the outlier. Blue arrow shows the true principle subspace and the red dashed arrow shows the wrong one DRMF would get starting from $\mathbf{S} = \mathbf{0}$. Note that when starting from an \mathbf{S} that correctly indicates the circle as an outlier, DRMF is able to achieve the correct blue subspace.

We found that an effective way is to solve this problem is to leverage the convexity of nuclear norm minimization (NNM) methods. Since NNM is a convex relaxation of DRMF, we can first compute the NNM solution of \mathbf{S} , and then use it to initialize DRMF. This strategy is similar to the case where the *linear programming* relaxation is used to approximate the original *integer programming* problems. In practice, we can run NNM for a few iterations and terminate before convergence. This is usually enough to guide DRMF to a good convergence region. In this way, we can get results that are better than using either NNM or DRMF alone. Other methods (*e.g.* [27]) can also be used to initialize DRMF. Using these initialization schemes, DRMF is able to overcome the problem posed in Figure 1 and get higher quality results than NNM.

Very recently we noticed a parallel work *GoDec* [31] that shares the same idea with DRMF. By comparison, DRMF extends to structured outliers as discussed in Section V-A. In addition, the non-convexity of DRMF/GoDec is not addressed in [31] and the GoDec algorithm in its original form would likely get stuck in the extreme case in Figure 1.

VI. EXPERIMENTS

In this section we show the empirical effectiveness of DRMF on both simulation and real-world data sets. We compare DRMF to the following state-of-the-art competitors:

- **Robust PCA (RPCA)** [3] We use the code from <http://perception.csl.uiuc.edu/matrix-rank>. The efficient “inexact augmented Lagrange multiplier” implementation is used.
- **Stable principal component pursuit (SPCP)** [32] We implemented SPCA in Matlab using the *proximal gradient* method according to [9].
- **Outlier Pursuit (OP)** [28] We implemented OP in Matlab using the *proximal gradient* method according to [28].

In terms of Table I, RPCA and SPCP solve the NNM problem with $\sigma = 0$ and $\sigma > 0$ respectively; OP solves NNM when the outlier is measured by $\|\mathbf{S}\|_{2,1}$ and $\sigma = 0$. The truncated SVD results (4) are also provided as a baseline.

DRMF and DRMF-R are implemented in Matlab. Partial SVD is done using PROPACK [15]. We terminate the iteration when the relative change of the objective value is diminishing.

DRMF, SPCP, and OP are all initialized by the solution produced by 10 iterations of RPCA. For DRMF, we *always* set the maximal number of allowed outliers to be $e = 0.05n^2$ without tuning unless indicated otherwise.

A. Simulation Data

First, we study the performances of different methods on simulated data sets. We follow the set up in [3] to create the data matrix. Let $\mathcal{N}(\mu, \sigma^2)$ denote the Gaussian distribution with mean μ and variance σ^2 , and $\mathcal{U}(a, b)$ denote the uniform distribution on the interval $[a, b]$. We generate the rank- K matrix as $\mathbf{L} = \mathbf{U}\mathbf{V}^T \in \mathbb{R}^{n \times n}$, where entries of the factor matrices \mathbf{U} and \mathbf{V} are *i.i.d.* samples from Gaussian distributions as $\mathbf{U} \in \mathbb{R}^{n \times K} \sim \mathcal{N}(0, 1/K)$, $\mathbf{V} \in \mathbb{R}^{n \times K} \sim \mathcal{N}(0, 1/K)$. To generate the outlier matrix \mathbf{S} , we first select γn^2 entries from \mathbf{S} and then draw their values from the uniform distribution $\mathcal{U}(-\sigma_o, +\sigma_o)$, where σ_o is the magnitude of outliers. Finally, we put them together and add *i.i.d.* Gaussian noise for each entry to get $\mathbf{X} = \mathbf{L} + \mathbf{S} + \mathcal{N}(0, \sigma_n^2)$, where σ_n is the level of the Gaussian observation noise.

1) *Recovery Quality and Detection Rate:* In this part we test how well the methods can detect the outliers and recover the underlying low-rank \mathbf{L} accurately. We compare the performances on three different indices. To measure the accuracy of robust modeling, we compute the *root mean squared error* (RMS) of the recovered $\hat{\mathbf{L}}$ *w.r.t.* the true \mathbf{L} . NNM results are “debiased” as in [21] to compensate the shrunken singular values. Outlier scores are computed as the absolute difference between the estimated $\hat{\mathbf{L}}$ and observation \mathbf{X} , and *average precision* (AP) is used to measure the detection performance. The simulation parameters we use are $K = 0.05n$, $\gamma = 0.05$, $\sigma_o = 1$. Finally, the running time is also compared.

First we examine the *entry outliers, noiseless observation* case by selecting uniformly random entries in \mathbf{S} to be outliers and setting $\sigma_n = 0$. This situation satisfies the assumption made by RPCA. We compare the performances of SVD, RPCA, and DRMF. Note that SPCP and OP cannot be applied to this data set. For RPCA, we use parameter $\lambda = 1/\sqrt{n}$ as suggested in [3]. For SVD and DRMF, the true K is used for factorization. Matrices with sizes n between [100, 2000] are used. Mean performances of 20 random runs are reported in Figure 2. We see that both RPCA and DRMF achieved much better performances than plain SVD, showing the necessity and effectiveness of robust factorization. Further, even in this noiseless case, DRMF

is able to outperform RPCA consistently, using much less running time (only slightly slower than partial SVD).

Next we examine the *entry outliers, noisy observation* case. Compared to the previous simulation, we use $\sigma_n = 0.1$ and other settings remain the same. Note that this situation violates the assumption made by RPCA. We compare SVD, RPCA, SPCP, and DRMF here. The same settings for SVD, RPCA, and DRMF are used as before. For SPCP, the parameter regarding the level of regular Gaussian noise is set as suggested by [32]. Mean performances of 20 random runs are reported in Figure 3. On this data set, we see DRMF achieves the best performance again. RPCA performs poorly because of the noise, which inflates the estimated rank dramatically. SPCP, which is essentially an extended version of RPCA to handle noisy data, shows much better accuracy here, but is still worse than DRMF. Based on these two experiments, we conclude that DRMF can handle both noisy or noiseless data sets, and is able to achieve better results than RPCA and SPCP.

Further we examine the *row outliers, noisy observation* case. Unlike the entry outlier case, here we randomly select γn ($\gamma = 0.05$) rows in \mathbf{S} and fill them with outliers from $\mathcal{U}(-1, 1)$. Note that this situation violates the assumptions made by RPCA and SPCP. We compare SVD, RPCA, SPCP, OP, DRMF, and DRMF-R here. For OP, we use parameter $\lambda = 0.4/\sqrt{\gamma n}$ as suggested by [28]. For DRMF-R, we directly specify that there can be γn outlier rows. Mean performances of 20 random runs are reported in Figure 4. In the presence of row outliers, SVD and SPCP failed to work for large matrices. By contrast, OP performs poorly for small matrices, but then catches up as n grows larger. RPCA, DRMF, and DRMF-R show stable performances and DRMF-R beats the others by a large margin. This verifies that utilizing additional knowledge about outlier patterns helps robust modeling and finding outliers. It is also interesting to see that even though DRMF is design to handle entry outliers, its recovery quality is not affected by row outliers as SPCP is.

Based on the above results, we conclude that DRMF algorithms outperform the NNM methods in various cases, including noiseless and noisy observations as well as different outlier patterns.

2) *Sensitivity:* In this section, we study the sensitivity of DRMF’s performance *w.r.t.* the magnitude of outliers and values of parameters.

First we examine how the magnitude of outliers affects the recovery quality. We simulate noiseless matrices with entry outliers, using $n = 400$, $K = 20$, and $\gamma = 0.05$. Then we change σ_o the magnitude of outliers from 1 to 10^5 , and calculate the RMS between the recovered $\hat{\mathbf{L}}$ and \mathbf{L} . Results produced by RPCA and DRMF are shown in Figure 5. We can see that the recovery quality of DRMF is not affected by the magnitude of outliers at all. This is expected: the L_0 -norm used in DRMF totally disregards the magnitude of

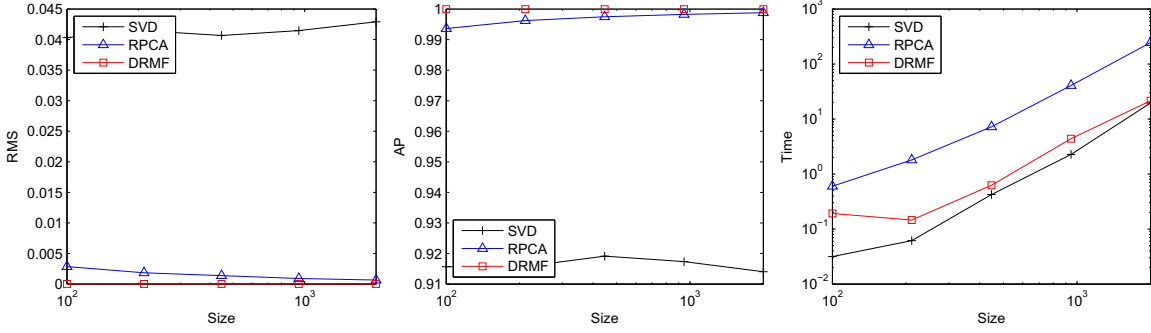


Figure 2. Performances on noiseless data with entry outliers. Note that the running time is shown in log-scale.

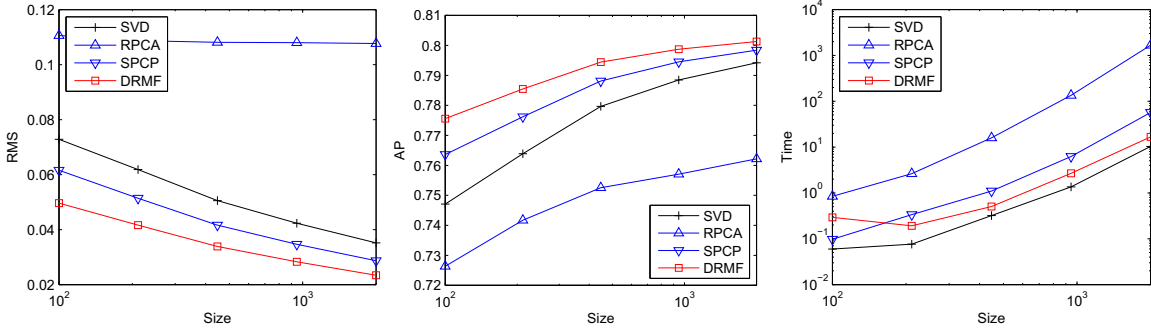


Figure 3. Performances on noisy data with entry outliers.

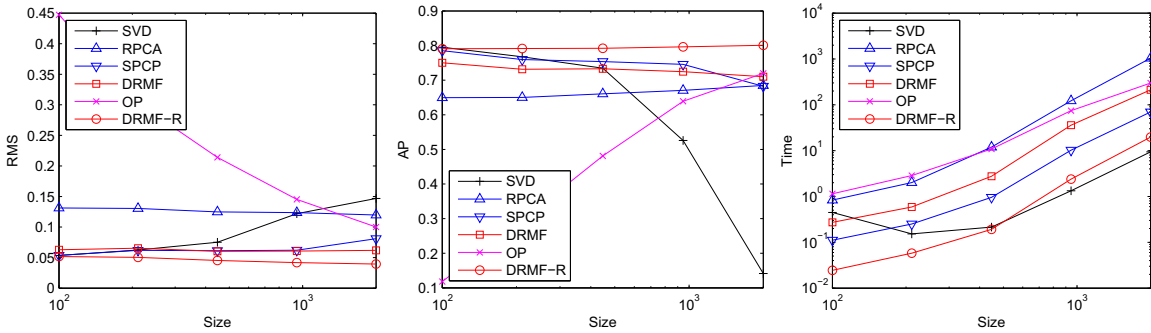


Figure 4. Performances on noisy data with row outliers.

outliers and only counts the number of them. On the other hand, though being robust, the L_1 -norm used in RPCA is still influenced by large outliers, and we observe that this influence grows linearly with the magnitude of outliers.

We also examine how the recovery quality of DRMF is affected by the choice of parameters K the rank and e the number (or equivalently the proportion) of allowed outliers. The matrices are generated in the same way as the previous experiment with $\sigma_o = 1$. Then we run DRMF with different K 's between $[14, 60]$ and different e 's between $[0, 0.2]$. Recovery RMS are shown in Figure 6. We can see that in a large range of parameters the performance is stable. It is especially interesting to see that moderately larger values of e actually produces better results. This behavior

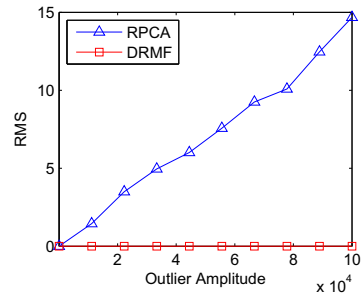


Figure 5. Recovery RMS of RPCA and DRMF versus the outliers' magnitude.

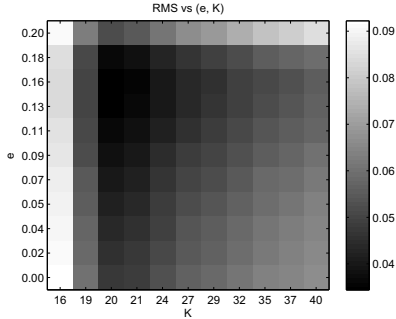


Figure 6. Recovery RMS of DRMF versus the parameters K the rank and e the proportion of allowed outliers. Darker color indicates smaller error.

verifies the role of e in DRMF: it is only a safeguard to prevent excessive data being regarded as outliers, and it does not need to be same as the true number of outliers. We also observe that performance can be degraded when using too small K s and too large e 's ($\geq 20\%$). This is expected: when e is too large, a large portion of data can be treated as outliers (this actually violates our definition of outliers) and thus the results become unfaithful. When K is too small, the model lacks the capability to capture the normal variability of data.

B. Video Background Modeling and Activity Detection

In this experiment we consider the problem of modeling the background of videos. Estimating the background in videos is important for many computer vision tasks such as activity detection, yet also difficult because of the variability of the background (*e.g.* due to lighting conditions) and the presence of foreground objects such as moving people.

Here we apply robust matrix factorization methods to solve this problem. We assume that the background variations in videos are of low-rank (*i.e.* the background scenes can be approximated by linear combinations of several “basis” images), and the foreground objects are sparse outliers. By applying robust factorization methods to these video data, we want that the low-rank component will capture the background and its variations, while the foreground activities will be recognized as outliers so that they will not interfere the estimation of background.

Video sequences “Hall” (size 128×160 , frames 2100-2400), “Lobby” (size 144×176 , frames 1300-1700), “Restaurant” (size 120×160 , frames 2500-3000), and “Shopping Mall” (size 128×160 , frames 1500-2000) from [18] are used. “Hall” contains a relatively static background and many foreground activities. “Lobby” contains few foreground activities and large background variations. “Restaurant” and “Shopping Mall” are noisier and contain much more foreground activities. Sample images are shown in Figure 7.

We flatten and stack the video frames into a matrix, with one row corresponds to a frame. Then we use SVD, RPCA, SPCA, DRMF to estimate the background. The anomaly

scores of pixels are computed as the absolute difference between the estimated background and the observation, so our hope is that pixels corresponding to foreground activities will receive high scores. The performance is measured by the average precision of detecting foreground pixels on the ground truth frames. We use the suggested parameters for RPCA and SPCA (For SPCA, the median of pixels’ standard deviation is used to estimate the Gaussian noise level). For SVD and DRMF, rank-5 models are used for “Hall”, “Lobby” and rank-7 models are used for “Restaurant”, “Shopping Mall” to capture the background variations.

Detection results on some ground-truth frames using DRMF and RPCA are shown in Figure 7. Both methods are able to separate the foreground and background and produce good results. By more detailed examination, we can see that the backgrounds images captured by DRMF are smoother and contains less artifacts than RPCA. Figure 8 shows the detection performance and running time of different methods. Again, we see that DRMF consistently gives better detection performance than RPCA and SPCP.

C. Hand-written Digit Modeling

In the last experiment, we use these factorization methods to find anomalous digit images. The assumption is that images of the same digits have a low-rank structure (*i.e.* these images reside in a low-dimensional subspace), and if we inject in a small amount of different digits, these injections will violate the low-rank structure and stand out as outliers.

We use digits ‘1’ and ‘7’ from the USPS data set as in [28]. The image size is 16×16 . We select a data set that is a mixture of 220 images of ‘1’ and 11 images of ‘7’. The goal is to detect all the ‘7’s in an unsupervised way. To do this, we flatten all images as row vectors and stack them into a 231×256 matrix \mathbf{X} . Then, factorization methods are applied to estimate low-rank matrices $\hat{\mathbf{L}}$ which are expected to capture the ‘1’s. Finally, each image (a row of \mathbf{X}) is scored by the L_2 -norm of its corresponding row in the error matrix $\mathbf{X} - \hat{\mathbf{L}}$. Ideally, ‘7’s should receive higher scores than ‘1’s.

We compare SVD, RPCA, SPCP, DRMF, OP, DRMF-R on this task. for SVD and DRMF methods, rank $K = 3$ is used. For NNM methods, suggested parameters are used as before. Performances are measured by the average precision of detecting ‘7’s. In each run, we randomly re-select the images. Results of 20 random runs are shown in Figure 9a.

We can see that DRMF-R gives the best results, showing the advantage of our direct solution, and the benefit from incorporating knowledge about the outliers’ structure. On the other hand, RPCA and SPCP failed in this case, since the non-uniform and non-random outliers in this data set violate their basic assumptions. The difference between OP and DRMF-R is significant: a *paired t-test* gives a p-value of 0.95×10^{-6} . Figure 9b shows a list of images ranked by their anomaly scores. We conclude that the ‘1’s are clearly captured by the low-rank structure in DRMF, and

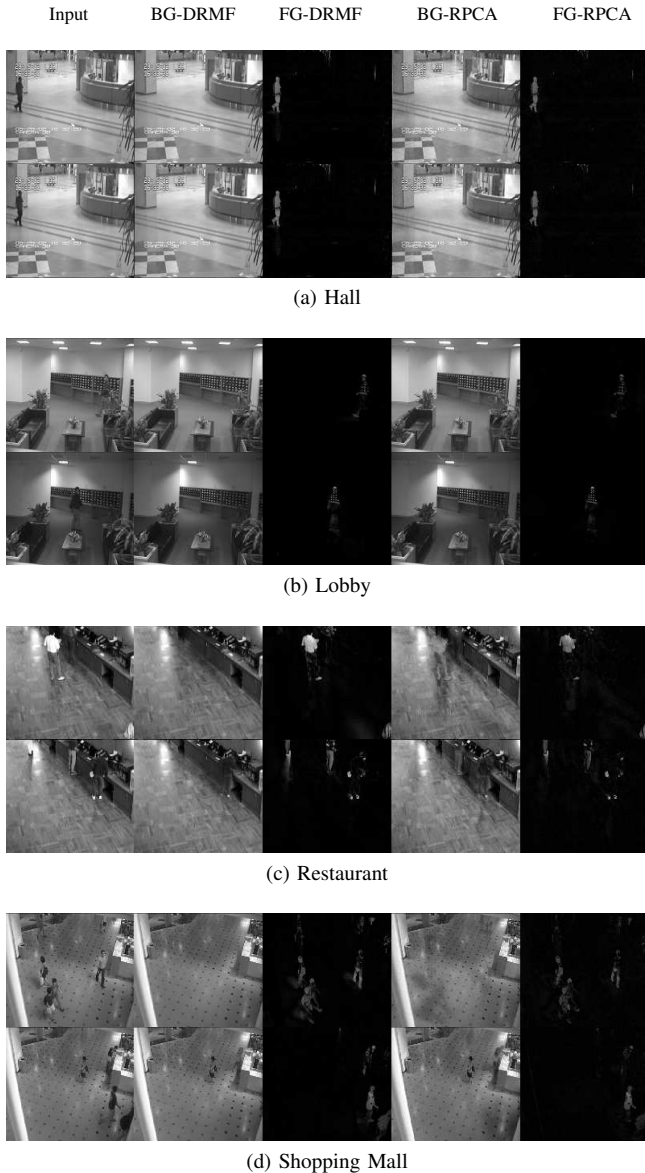


Figure 7. Video activity detection result frames. In each sub-figure, the images from left to right are: the original frame, background and foreground from DRMF, background and foreground from RPCA.

it is interesting to observe the behavior of the (1, 2)-th and the (2, 5)-th image.

VII. CONCLUSION

We proposed the direct robust matrix factorization (DRMF) algorithm as a simple and effective way for robust low-rank factorizations and outlier detection on matrices. We start from the fundamental notion of outliers and use a direct formulation to address this problem. DRMF is conceptually simple (SVD + error thresholding), easy to implement (about 10 lines of Matlab code), efficient (linear complexity *w.r.t.* number of entries), and flexible to incorporate prior knowl-

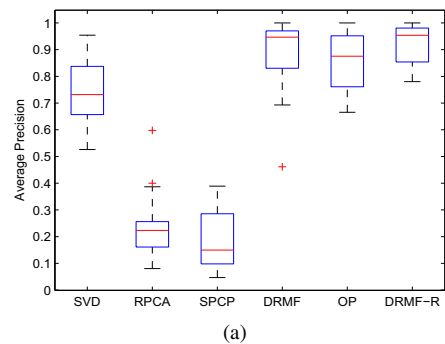
	Hall	Lobby	Restaurant	Mall
SVD	0.669	0.695	0.547	0.721
RPCA	0.768	0.754	0.596	0.713
SPCA	0.746	0.770	0.549	0.753
DRMF	0.805	0.792	0.666	0.774

(a) Average Precision

	Hall	Lobby	Restaurant	Mall
SVD	4.887	1.537	2.007	3.237
RPCA	405.938	451.965	618.082	572.738
SPCA	18.627	17.401	25.058	42.341
DRMF	23.760	16.922	29.598	57.409

(b) Running Time (seconds)

Figure 8. Video activity detection performance



(a)



(b)

Figure 9. USPS anomaly detection results. (a) the average precisions of detecting '7's among '1's. (b) images ranked by their anomaly scores in the descending order.

edge about both the outliers and the low-rank structure.

DRMF is compared to the recently proposed nuclear norm minimization (NNM) family methods. We show that NNM methods are in fact convex relaxations of DRMF. In extensive empirical evaluations we find that the solutions given by DRMF achieve better performances over the state-of-the-art competitors that use relaxations, showing the advantage of our direct formulation.

ACKNOWLEDGMENT

This work was funded in part by the National Science Foundation under grant number NSF-IIS0911032 and the Department of Energy under grant number DESC0002607.

REFERENCES

- [1] P. Bloomfield and W. L. Steiger, *Least Absolute Deviations: Theory, Applications, and Algorithms (Progress in Probability)*. Birkhäuser Boston, Mass, USA, 1983.
- [2] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: Identifying density-based local outliers," in *ACM SIGMOD Record*, 2000.
- [3] E. Candes, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *preprint*, 2009.
- [4] E. J. Candès and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," *IEEE Trans. Information Theory*, vol. 56(5), pp. 2053–2080, 2009.
- [5] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, vol. 41(3), pp. 1–72, 2009.
- [6] C. Ding, T. Li, and M. I. Jordan, "Convex and semi-nonnegative matrix factorizations," *IEEE T-PAMI*, vol. 32(1), pp. 45–55, 2010.
- [7] D. L. Donoho, "Breakdown properties of multivariate location estimators," Ph.D. dissertation, Harvard University, 1982.
- [8] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, pp. 211–218, 1936.
- [9] A. Ganesh, Z. Lin, J. Wright, L. Wu, M. Chen, and Y. Ma, "Fast algorithms for recovering a corrupted low-rank matrix," in *International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, 2009.
- [10] D. M. Hawkins, L. Liu, and S. S. Young, "Robust singular value decomposition," National Institute of Statistical Sciences, Tech. Rep., 2001.
- [11] Huber and P. J., "Robust estimation of a location parameter," *Annals of Statistics*, vol. 53, pp. 73–101, 1964.
- [12] Q. Ke and T. Kanade, "Robust l_1 norm factorization in the presence of outliers and missing data by alternative convex programming," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.
- [13] H.-P. Kriegel, P. Kroger, E. Schubert, and A. Zimek, "A general framework for increasing the robustness of pca-based correlation clustering algorithms," in *Scientific and Statistical Database Management Conference (SSDBM)*, 2008.
- [14] F. D. la Torre and M. J. Black, "A framework for robust subspace learning," *International Journal of Computer Vision*, vol. 54, pp. 117–142, 2003.
- [15] R. M. Larsen, "Propack - software for large and sparse svd calculations." [Online]. Available: <http://soi.stanford.edu/~rmunk/PROPACK>
- [16] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.
- [17] G. Li and Z. Chen, "Projection-pursuit approach to robust dispersion matrices and principal components: Primary theory and monte carlo," *J. of American Statistical Association*, vol. 80, no. 391, pp. 759 – 766, 1985.
- [18] L. Li, W. Huang, I. Y.-H. Gu, and Q. Tian, "Statistical modeling of complex backgrounds," *IEEE Trans. Image Processing*, vol. 13, no. 11, pp. 1459–1472, 2004.
- [19] J. Liu, S. Ji, and J. Ye, "Multi-task feature learning via efficient $l_{2,1}$ -norm minimization," in *The Twenty-fifth Conference on Uncertainty in Artificial Intelligence (UAI)*, 2009.
- [20] Z. Lu and Y. Zhang, "Penalty decomposition methods for l_0 -norm minimization," Department of Mathematics, Simon Fraser University, Tech. Rep., 2010.
- [21] S. Ma, D. Goldfarb, and L. Chen, "Fixed point and bregman iterative methods for matrix rank minimization," *Math. Program., Ser. A*, to appear.
- [22] R. Mazumder, T. Hastie, and R. Tibshirani, "Spectral regularization algorithms for learning large incomplete matrices," *Journal of Machine Learning Research*, 2009.
- [23] M. H. Nguyen and F. D. la Torre, "Robust kernel principal components analysis," in *NIPS*, 2009.
- [24] N. H. Nguyen, T. T. Do, and T. D. Tran, "A fast and efficient algorithm for low-rank approximation of a matrix," in *STOC*, 2009.
- [25] J. D. M. Rennie and N. Srebro, "Fast maximum margin matrix factorization for collaborative prediction," in *ICML*, 2005.
- [26] R. Salakhutdinov and A. Minh, "Probabilistic matrix factorization," in *NIPS*, 2007.
- [27] H. Xu, C. Caramanis, and S. Mannor, "Principal component analysis with contaminated data: The high dimensional case," in *Annual Conference on Learning Theory (CoLT)*, 2010.
- [28] H. Xu, C. Caramanis, and S. Sanghavi, "Robust pca via outlier pursuit," in *NIPS*, 2010.
- [29] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *Journal of the Royal Statistical Society, Series B*, vol. 68, pp. 49–67, 2006.
- [30] M. Zhao and V. Saligrama, "Anomaly detection with score functions based on nearest neighbor graphs," in *NIPS*, 2009.
- [31] T. Zhou and D. Tao, "Godec: Randomized low-rank & sparse matrix decomposition in noisy case," in *International Conference on Machine Learning*, 2011.
- [32] Z. Zhou, X. Li, J. Wright, E. Candès, and Y. Ma, "Stable principal component pursuit," in *International Symposium on Information Theory*, 2010.