

---

# Low-Rank Matrix Recovery from Row-and-Column Affine Measurements

---

Avishai Wagner

Or Zuk

Dept. of Statistics, The Hebrew University of Jerusalem, Mt. Scopus, Jerusalem, 91905, Israel

AVISHAI.WAGNER@MAIL.HUJI.AC.IL

OR.ZUK@MAIL.HUJI.AC.IL

## Abstract

We propose and study a row-and-column affine measurement scheme for low-rank matrix recovery. Each measurement is a linear combination of elements in one row or one column of a matrix  $X$ . This setting arises naturally in applications from different domains. However, current algorithms developed for standard matrix recovery problems do not perform well in our case, hence the need for developing new algorithms and theory for our problem. We propose a simple algorithm for the problem based on Singular Value Decomposition (SVD) and least-squares (LS), which we term SVLS. We prove that (a simplified version of) our algorithm can recover  $X$  exactly with the minimum possible number of measurements in the noiseless case. In the general noisy case, we prove performance guarantees on the reconstruction accuracy under the Frobenius norm. In simulations, our row-and-column design and SVLS algorithm show improved speed, and comparable and in some cases better accuracy compared to standard measurements designs and algorithms. Our theoretical and experimental results suggest that the proposed row-and-column affine measurements scheme, together with our recovery algorithm, may provide a powerful framework for affine matrix reconstruction.

## 1. Introduction

In the low-rank affine matrix recovery problem, an unknown matrix  $X \in \mathbb{R}_{n_1 \times n_2}$  with  $\text{rank}(X) = r$  is measured indirectly via an affine transformation  $\mathcal{A} : \mathbb{R}_{n_1 \times n_2} \rightarrow \mathbb{R}^d$  and possibly with additive (typically Gaussian) noise  $z \in \mathbb{R}^d$ . Our goal is to recover  $X$  from the vector of noisy measurements  $\mathbf{b} = \mathcal{A}(X) + z$ . The problem has found numerous applications throughout science

and engineering, in different fields such as collaborative filtering (Koren et al., 2009), face recognition (Basri & Jacobs, 2003), quantum state tomography (Gross et al., 2010) and computational biology (Chi et al., 2013). The problem has been studied mathematically quite extensively in the last few years. Most attention thus far has been given to two particular ensembles of random transformations  $\mathcal{A}$ : (i) the Matrix Completion (MC) setting, in which each element of  $\mathcal{A}(X)$  is a single entry of the matrix where the subset of the observed measurements is sampled uniformly at random (Candès & Recht, 2009; Candès & Plan, 2010; Candès & Tao, 2010; Keshavan et al., 2009; 2010; Recht, 2011) (ii) Gaussian-Ensemble (GE) affine-matrix-recovery, in which each element of  $\mathcal{A}(X)$  is a weighted sum of all elements of  $X$  with i.i.d. Gaussian weights (Candès & Plan, 2011; Recht et al., 2010). Remarkably, although the recovery problem is in general NP-hard, when  $r \ll \min(n_1, n_2)$  and under certain conditions on the matrix  $X$  or the measurement operator  $\mathcal{A}$ , one can recover  $X$  from  $d \ll n_1 n_2$  measurements with high probability and using efficient algorithms (Candès & Recht, 2009; Recht et al., 2010; Candès & Tao, 2010; Recht, 2011). However, it is desirable to study the problem with other affine transformations  $\mathcal{A}$  beyond the two ensembles mentioned above for the following reasons: (i) In some applications we cannot control the measurements operator  $\mathcal{A}$ , and different models for the measurements may be needed to allow a realistic analysis of the problem (ii) When we can control and design the measurement operator  $\mathcal{A}$ , other measurement operators may outperform the two ensembles mentioned above with respect to optimizing different resources such as the number of measurements required, computation time and storage. The main goal of this paper is to present and study a different set of affine transformations, which we term row-and-column affine measurements. This setting may arise naturally in many applications, since it is often natural and possibly cheap to measure a single row or column of a matrix, or a linear combination of a few such rows and columns. For example, (i) In collaborative filtering, we may wish to recover a users-items preference matrix and have access to only a subset of the users, but can observe their preference scores for *all* items (ii) When recovering

a protein-RNA interactions matrix in molecular biology, a single experiment may simultaneously measure the interactions of a specific protein with all RNA molecules (Chu et al., 2011).

In general, we can represent any affine transformation  $\mathcal{A}$  in matrix representation  $\mathcal{A}(X) = A\text{vec}(X)$ , where  $\text{vec}(X)$  is a column vector obtained by stacking all columns of  $X$  on top of each other. In our row and column framework the measurement operator  $\mathcal{A}$  is represented differently using two matrices  $A^{(R)}, A^{(C)}$  which multiply  $X$  as a matrix (rather than multiplying the vector  $\text{vec}(X)$ ) from left and right, respectively. We focus on two ensembles of  $A^{(R)}, A^{(C)}$ : (i) Matrix Completion from single Columns and Rows (RCMC). Here we observe single matrix entries in similar to standard matrix completion case, however the measured entries are not scattered randomly along the matrix, but instead we sample a few rows and a few columns, and measure *all* entries in these rows and columns. This ensemble is implemented by setting the rows (columns) of  $A^{(R)}$  ( $A^{(C)}$ ) as random vectors from the standard basis of  $\mathbb{R}^{n_1}$  ( $\mathbb{R}^{n_2}$ ). (ii) Gaussian Row-and-Column (GRC) measurements. Here each set of measurements is a weighted linear combination of the matrix’s rows (or columns) with the weights taken as i.i.d. Gaussians. This ensemble is implemented by setting the entries of  $A^{(R)}, A^{(C)}$  as i.i.d. Gaussian random variables.

The measurement operators  $\mathcal{A}$  in our RCMC and GRC models do not satisfy standard requirements which hold for GE and MC. It is thus not surprising that algorithms such as nuclear norm minimization (Recht et al., 2010; Candès & Recht, 2009), which succeed for the GE and MC models, fail in our case, and different algorithms and theory are required. However, the specific algebraic structure provided by the row-and-column measurements, allows us to derive efficient and simple algorithms, and to analyze their performance. In addition, we provide extensive simulation results, which demonstrate the improved accuracy and speed of our approach over existing measurement designs and algorithms. All of our algorithms and simulations are implemented in a Matlab software package available at <https://github.com/avishaiwa/SVLS>.

### 1.1. Prior Work

Before giving a detailed derivation and analysis of our design and algorithms, we give an overview of existing designs and their properties. We concentrate on two properties: (i) storage required in order to represent the measurement operator, and (ii) measurement sparsity, defined as the sum over all measurements of the number of matrix entries participating in each measurement, that is  $S(\mathcal{A}) = \|\text{vec}(A)\|_0$ . The latter property may be related to measurement costs, as well as to computational time.

In the Gaussian Ensemble model, the entries of the matrix  $A$  in the matrix representation  $\mathcal{A}(X) = A\text{vec}(X)$  are i.i.d. Gaussian random variables,  $A_{ij} \sim N(0, 1)$ . For this ensemble, one can recover uniquely a low rank matrix  $X$  with  $O(r(n_1 + n_2))$  noiseless measurements using nuclear norm minimization (Recht et al., 2010; Candès & Plan, 2011) or other methods such as Singular Value Projection (SVP) (Jain et al., 2010), which is optimal up to constants. Recovery in this model is robust to noise, with only a small increase in number of measurements. The main disadvantage of this model is that the design requires  $O(dn_1n_2)$  storage space for  $A$ , which could be problematic for large matrices. Another possible disadvantage of this method is that measurements are dense - each measurement represents a linear combination of all  $O(n_1n_2)$  matrix entries, and the overall measurement sparsity of  $\mathcal{A}(X)$  is also  $O(dn_1n_2)$ , which could be problematic for large  $n_1, n_2$ .

In the standard matrix completion problem (Candès & Recht, 2009) we can recover  $X$  with high probability from single entries chosen uniformly at random using nuclear norm minimization (Cai et al., 2010; Toh & Yun, 2010; Candès & Tao, 2010; Ma et al., 2011; Recht, 2011) or using other methods such as *SVD* and gradient descent (Keshavan et al., 2009; 2010). This model has the lowest storage requirements ( $O(d)$ ) and measurement sparsity ( $O(d)$ ). However, recovery guarantees for this model are weaker: setting  $n = \max(n_1, n_2)$ , it is shown that  $\Theta(nr \log(n))$  measurements are required to recover a rank  $r$  matrix of size  $n_1 \times n_2$  (Candès & Tao, 2010). In addition, unique recovery from this number of measurements requires additional incoherence conditions on the matrix  $X$ , and recovery of matrices which fail to satisfy such conditions (e.g. matrices with a few spikes) may require a much larger number of measurements.

Recently a new design of rank one projections was proposed (Cai & Zhang, 2015), where each measurement is of the form  $\alpha^T X \beta$  and such that  $\alpha \in \mathbb{R}^{n_1}, \beta \in \mathbb{R}^{n_2}$  have i.i.d standard Gaussian entries. It was proven that nuclear norm minimization can recover  $X$  with high probability in this design from  $O(n_1r + n_2r)$  measurements. This is the first model deviating from MC and GE we are aware of. This model is different from our row-and-column model, as each measurement is obtained by multiplying  $X$  from both sides, whereas in our model each measurement is obtained by multiplying  $X$  from either left or right. Moreover, in our model the measurements are not chosen independently from each other but come in groups of size  $n_1$  or  $n_2$  (corresponding to rows or columns  $A^{(R)}, A^{(C)}$ ). An advantage of rank one projection is that it leads to a significance reduction in measurement storage needed for  $\mathcal{A}$  with overall  $O(dn_1 + dn_2)$  storage space. However, each measurement is still dense and involve all matrix elements, hence measurement sparsity is  $O(dn_1n_2)$ . In contrast, our

GRC model requires only  $O(d)$  storage for  $\mathcal{A}$ , and every measurement depends only on  $O(n)$  elements, leading to a reduced overall time for all measurements  $O(dn_1 + dn_2)$ . For RCMC, we need only  $O(\frac{d \log(n)}{n})$  storage for  $\mathcal{A}$ , and measurement sparsity is  $O(d)$ .

## 2. Preliminaries and Notations

We denote by  $\mathbb{R}_{n_1 \times n_2}$  the space of matrices of size  $n_1 \times n_2$ , by  $\mathcal{O}_{n_1 \times n_2}$  the space of matrices of size  $n_1 \times n_2$  with orthonormal columns, and by  $\mathcal{M}_{n_1 \times n_2}^{(r)}$  the space of matrices of size  $n_1 \times n_2$  and rank  $\leq r$ . We denote  $n = \max(n_1, n_2)$ .

We denote by  $\|\cdot\|_F$  the matrix Frobenius norm, by  $\|\cdot\|_*$  the nuclear norm, and by  $\|\cdot\|_2$  the spectral norm. For a vector,  $\|\cdot\|$  denotes the standard  $l_2$  norm.

For  $X \in \mathbb{R}_{n_1 \times n_2}$  we denote by  $\text{span}(X)$  the subspace of  $\mathbb{R}^{n_1}$  spanned by the columns of  $X$  and define  $P_X$  to be the orthogonal projection into  $\text{span}(X)$ .

For a matrix  $X$  we denote by  $X_{i\bullet}$  the  $i$ -th row, by  $X_{\bullet j}$  the  $j$ -th column and by  $X_{ij}$  the  $(i, j)$  element. For two sets of indices  $I, J$ , we denote by  $X_{IJ}$  the sub-matrix obtained by taking the rows with indices in  $I$  and columns with indices in  $J$  of  $X$ . We denote by  $[k]$  the set of indices  $1, \dots, k$ . We denote by  $\text{vec}(X)$  the (column) vector obtained by stacking all the columns of  $X$  on top of each other.

We use the notation  $X \stackrel{i.i.d.}{\sim} G$  to denote a random matrix  $X$  with i.i.d. entries  $X_{ij} \sim G$ .

For a rank- $r$  matrix  $X \in \mathcal{M}_{n_1 \times n_2}^{(r)}$  let  $X = U\Sigma V^T$  be the Singular Value Decomposition (SVD) of  $X$  where  $U \in \mathcal{O}_{n_1 \times r}$ ,  $V \in \mathcal{O}_{r \times n_2}$  and  $\Sigma = \text{diag}(\sigma_1(X), \dots, \sigma_r(X))$  with  $\sigma_1(X) \geq \sigma_2(X) \geq \dots \geq \sigma_r(X) > 0$  the (non-zero) singular values of  $X$  (we omit the zero singular values and their corresponding vectors from the decomposition). For a general matrix  $X \in \mathbb{R}_{n_1 \times n_2}$  we denote by  $X_{(r)}$  the top- $r$  singular value decomposition of  $X$ ,  $X_{(r)} = U_{\bullet[r]} \Sigma_{[r][r]} V_{\bullet[r]}^T$ .

Our model assumes two affine transformations applied to  $X$ , representing rows and columns,  $B^{(C,0)} = XA^{(C)}$  and  $B^{(R,0)} = A^{(R)}X$ , achieved by multiplications with two matrices  $A^{(R)} \in \mathbb{R}_{k^{(R)} \times n_1}$  and  $A^{(C)} \in \mathbb{R}_{n_2 \times k^{(C)}}$ , respectively. We obtain noisy observations of these transformations,  $B^{(R)}, B^{(C)}$  obtained by applying additive noise:

$$A^{(R)}X + Z^{(R)} = B^{(R)}; \quad XA^{(C)} + Z^{(C)} = B^{(C)} \quad (1)$$

where the total number of measurements is  $d = k^{(R)}n_1 + n_2k^{(C)}$ , and  $Z^{(R)} \in \mathbb{R}_{n_1 \times k^{(R)}}$ ,  $Z^{(C)} \in \mathbb{R}_{k^{(C)} \times n_2}$  are two zero-mean noise matrices. Our goal is to recover  $X$  from the observed measurements  $B^{(C)}$  and  $B^{(R)}$ . To achieve this goal, we define the squared loss function

$$\mathcal{F}(X) = \|A^{(R)}X - B^{(R)}\|_F^2 + \|XA^{(C)} - B^{(C)}\|_F^2 \quad (2)$$

and solve the least squares problem:

$$\text{Minimize } \mathcal{F}(X) \text{ s.t. } X \in \mathcal{M}_{n_1 \times n_2}^{(r)}. \quad (3)$$

If  $Z^{(R)}, Z^{(C)} \stackrel{i.i.d.}{\sim} N(0, \tau^2)$ , minimizing the loss function in eq. (2) is equivalent to maximizing the log-likelihood of the data, giving a statistical motivation for the above score. Problem (3) is non-convex due to the non-convex rank constraint  $\text{rank}(X) \leq r$ .

Our problem is a specialization of the general affine matrix recovery problem (Recht et al., 2010), in which a matrix is measured using a general affine transformation  $\mathcal{A}$  with  $\mathbf{b} = \mathcal{A}(X) + z$ . We consider next and throughout the paper two specific *random ensembles* of measurement matrices:

- 1. Row and Column Matrix Completion (RCMC):** In this ensemble each row of  $A^{(R)}$  and each column of  $A^{(C)}$  is a vector of the standard basis  $e_j$  for some  $j$  - thus each measurement  $B_{ij}^{(R)}$  or  $B_{ij}^{(C)}$  is obtained from a single entry of  $X$ . We define a row-inclusion probability  $p^{(R)}$  and column inclusion probability  $p^{(C)}$  such that each row (column) of  $X$  will be measured with probability  $p^{(R)}$  ( $p^{(C)}$ ). More precisely, we define  $r_1, \dots, r_{n_1}$  i.i.d. Bernoulli variables,  $P(r_i = 1) = p^{(R)}$ , and include  $e_i$  as a row in  $A^{(R)}$  if and only if  $r_i = 1$ . Similarly, we define  $c_1, \dots, c_{n_2}$  i.i.d. Bernoulli variables,  $P(c_i = 1) = p^{(C)}$ , and include  $e_i$  as a column in  $A^{(C)}$  if and only if  $c_i = 1$ . The expected number of observed rows (columns) is  $k^{(R)} = n_1 p^{(R)}$  ( $k^{(C)} = n_2 p^{(C)}$ ). The model is very similar to the possibly more natural model of picking  $k^{(R)}$  distinct rows and  $k^{(C)}$  distinct columns at random for fixed  $k^{(R)}, k^{(C)}$ , but allows for easier analysis.
- 2. Gaussian Rows and Columns (GRC):** In this ensemble  $A^{(R)}, A^{(C)} \stackrel{i.i.d.}{\sim} N(0, 1)$ . Each observation  $B_{ij}^{(R)}$  or  $B_{ij}^{(C)}$  is obtained by a weighted sum of a single row or column of  $X$ , with i.i.d. Gaussian weights.

### 2.1. Comparison to Standard Designs

Our proposed rows-and-columns design differs from standard designs appearing in the literature. It is instructive to compare our GRC ensemble to the Gaussian Ensemble (GE) (Candès & Plan, 2011), with the matrix representation  $\mathcal{A}(X) = \text{Avec}(X)$  where  $A \in \mathbb{R}_{d \times n_1 n_2}$  and  $A \stackrel{i.i.d.}{\sim} N(0, 1)$ . For the latter, the following  $r$ -Restricted Isometry Property (RIP) can be used:

**Definition 1.** ( $r$ -RIP) Let  $\mathcal{A} : \mathbb{R}_{n_1 \times n_2} \rightarrow \mathbb{R}^d$  be a linear map. For every integer  $r$  with  $1 \leq r \leq \min(n_1, n_2)$ , define the  $r$ -Restricted Isometry Constant to be the smallest number  $\epsilon_r$  such that

$$(1 - \epsilon_r)\|X\|_F \leq \|\mathcal{A}(X)\| \leq (1 + \epsilon_r)\|X\|_F \quad (4)$$

holds for all matrices  $X$  of rank at most  $r$ .

The GE model satisfies the  $r$ -RIP condition for  $d = O(rn)$  with high probability (Recht et al., 2010). Based on this property it is known that nuclear norm minimization (Recht et al., 2010; Candès & Plan, 2011) and other algorithms such as SVP (Jain et al., 2010) can recover  $X$  with high probability. Unlike GE, in our GRC model  $\mathcal{A}(X)$  doesn't satisfy the  $r$ -RIP, and nuclear norm minimization fails. Instead,  $A^{(R)}, A^{(C)}$  preserve matrix Frobenius norm in high probability - a weaker property which holds for *any* low-rank matrix. (see Lemma 7 in the Appendix).

We next compare RCMC to the standard Matrix Completion model (Candès & Recht, 2009), in which single entries are chosen at random to be observed. Unlike GE, for MC incoherence conditions on  $X$  are required in order to guarantee unique recovery of  $X$  (Candès & Recht, 2009):

**Definition 2.** (*Incoherence*). Let  $U$  be a subspace of  $\mathbb{R}^n$  of dimension  $r$ , and  $P_U$  be the orthogonal projection on  $U$ . Then the coherence of  $U$  (with respect to the standard basis  $\{e_i\}$ ) is defined as

$$\mu(U) \equiv \frac{n}{r} \max_i \|P_U(e_i)\|^2. \quad (5)$$

We say that a matrix  $X \in \mathbb{R}_{n_1 \times n_2}$  is  $\mu$ -incoherent if for the SVD  $X = U\Sigma V^T$  we have  $\max(\mu(U), \mu(V)) \leq \mu$ .

When  $X$  is  $\mu$ -incoherent, and when known entries are sampled uniformly at random from  $X$ , several algorithms (Keshavan et al., 2009; Cai et al., 2010; Jain et al., 2010) succeed to recover  $X$  with high probability. In particular, nuclear norm minimization has gained popularity as a solver for the standard MC problem because it provides recovery guarantees and a convenient representation as a convex optimization problem with availability of many iterative solvers for the problem. However, nuclear norm minimization fails for the RCMC design, even when the matrix  $X$  is incoherent, as shown by the next example:

**Example 1.** Take  $X \in \mathbb{R}_{n \times n}$  for  $\frac{n}{3} \in \mathbb{N}$  with  $X_{ij} = 1, \forall (i, j) \in [n] \times [n]$ . Thus  $\|X\|_* = n$ . Take  $k^{(R)} = k^{(C)} = \frac{n}{3}$ . Set all unknown entries to 0.5, giving a matrix  $X_0$  of rank 2 with  $\sigma_1(X_0) = \frac{(\sqrt{2}+1)n}{3}, \sigma_2(X_0) = \frac{(\sqrt{2}-1)n}{3}$ . Therefore  $\|X_0\|_* = \frac{n\sqrt{2}}{3} < \|X\|_*$  and nuclear norm minimization fails to recover the correct  $X$ .

In Section 3 we present our SVLS algorithm, which does not rely on nuclear-norm minimization. In Section 4 we show that SVLS successfully approximates  $X$  for the GRC ensemble.

### 3. Algorithms for Recovery of $X$

In this section we give an efficient algorithm which we call SVLS (Singular Value Least Squares). SVLS is very easy to implement - for simplicity, we start with Algorithm 1 for the noiseless case and then present Algorithm 2 (SVLS) which is applicable for the general (noisy) case.

#### 3.1. Noiseless Case

In the noiseless case we reduce the optimization problem (3) to solving a system of linear equations (Candès & Recht, 2009), and provide Algorithm 1, which often leads to a closed-form estimator. We then give conditions under which with high probability, the closed-form solution is unique and is equal to the true matrix  $X$ . If

---

#### Algorithm 1

---

Input:  $A^{(R)}, A^{(C)}, B^{(R)}, B^{(C)}$  and rank  $r$

1. Compute a basis (of size  $r$ ) to the column space of  $B^{(C)}$  using Gaussian elimination, represented as the columns of a matrix  $\hat{U} \in \mathbb{R}_{n_1 \times r}$ .
  2. Solve the linear system  $B_{\bullet j}^{(R)} = A^{(R)}\hat{U}Y_{\bullet j}$  for each  $j = 1, \dots, n_2$  and write the solutions as a matrix  $Y = Y_{\bullet 1} \dots Y_{\bullet n_2}$ .
  3. Output  $\hat{X} = \hat{U}Y$
- 

$\text{rank}(A^{(R)}\hat{U}) = r$  one can write the resulting estimator  $\hat{X}$  in closed-form as follows:

$$\hat{X} = \hat{U}Y = \hat{U}[\hat{U}^T A^{(R)T} A^{(R)}\hat{U}]^{-1} \hat{U}^T A^{(R)T} B^{(R)} \quad (6)$$

Algorithm 1 does not treat the row and column measurements symmetrically. We can apply the same algorithm, but changing the role of rows and columns. The resulting closed form solution is then:

$$\hat{X} = B^{(C)} A^{(C)} \hat{V} [\hat{V}^T A^{(C)} A^{(C)T} \hat{V}]^{-1} \hat{V}^T \quad (7)$$

for an orthogonal matrix  $\hat{V}$  representing a basis for the rows of  $X$ . Since the algorithm uses Gaussian elimination steps for solving systems of linear equations, it is crucial that we have exact noiseless measurements. Next, we modify the algorithm to work also for noisy measurements.

#### 3.2. General (Noisy) Case

In the noisy case we seek a matrix  $X$  minimizing the loss  $\mathcal{F}$  in eq. (2). The minimization problem is non-convex and there is no known algorithm with optimality guarantees. We propose Algorithm 2 (SVLS), which empirically returns a matrix estimator  $\hat{X}$  with a low value of the loss



$\mathcal{F}$ . In addition, we prove in Section 4 recovery guarantees on the performance of SVLS.

---

**Algorithm 2** SVLS
 

---

Input:  $A^{(R)}, A^{(C)}, B^{(R)}, B^{(C)}$  and rank  $r$

1. Compute  $\hat{U}$ , the  $r$  largest left singular vectors of  $B^{(C)}$ , ( $\hat{U}$  is a basis for the columns space of  $B_{(r)}^{(C)}$ ).

2. Find the least-squares solution

$$\hat{Y} = \operatorname{argmin}_Y \|B^{(R)} - A^{(R)}\hat{U}Y\|_F. \quad (8)$$

If  $\operatorname{rank}(A^{(R)}\hat{U}) = r$  we can write  $\hat{Y}$  in closed form as before:

$$\hat{Y} = [\hat{U}^T A^{(R)T} A^{(R)} \hat{U}]^{-1} \hat{U}^T A^{(R)T} B^{(R)}. \quad (9)$$

3. Compute the estimate  $\hat{X}^{(R)} = \hat{U}\hat{Y}$ .
  4. Repeat steps 1-3, replacing the roles of columns and rows to get an estimate  $\hat{X}^{(C)}$ .
  5. Set  $\hat{X} = \operatorname{argmin}_{\hat{X}^{(R)}, \hat{X}^{(C)}} \mathcal{F}(X)$ , for the loss  $\mathcal{F}(X)$  given in eq. (2).
- 

### 3.2.1. GRADIENT DESCENT

The estimator  $\hat{X}$  returned by SVLS may not minimize the loss function  $\mathcal{F}$  in eq. (2). We therefore perform an additional gradient descent stage starting from  $\hat{X}$  to achieve an estimator with lower loss (while still possibly only a local minimum since the problem is non-convex). SVLS can be thus viewed as a fast method for providing a desirable starting point for local-search algorithms. The details of the gradient descent are given in the Appendix, Section 7.2.

### 3.3. Estimation of Unknown Rank

In real life problems, one doesn't know the true rank of a matrix and should estimate it from data. Our rows-and-columns sampling design is particularly suitable for rank estimation since  $\operatorname{rank}(B^{(C,0)}) = \operatorname{rank}(B^{(R,0)}) = \operatorname{rank}(X)$  with high probability when enough rows and columns are sampled. In the noiseless case we can estimate  $\operatorname{rank}(X)$  by  $\hat{r} = \operatorname{rank}(B^{(C,0)})$  or  $\operatorname{rank}(B^{(R,0)})$ .

For the noisy case we estimate  $\operatorname{rank}(X)$  from  $B^{(C)}, B^{(R)}$ . We use the popular elbow method to estimate  $\operatorname{rank}(B^{(C)})$  in the following way

$$\hat{r}^{(C)} = \operatorname{argmax}_{i \in [k^{(C)}-1]} \left( \frac{\sigma_i(B^{(C)})}{\sigma_{i+1}(B^{(C)})} \right) \quad (10)$$

We compute similarly  $\hat{r}^{(R)}$  from  $B^{(R)}$  and take the average as our rank estimator,  $\hat{r} = \operatorname{round}\left(\frac{\hat{r}^{(C)} + \hat{r}^{(R)}}{2}\right)$ . We

demonstrate the performance of our rank estimation using simulations in the Appendix, Section 7.7.

Modern methods for rank estimation from singular values (Gavish & Donoho, 2014) can be similarly applied to  $B^{(R)}, B^{(C)}$  and may yield more accurate rank estimates. After we estimate the rank, we can plug-in  $\hat{r}$  as the rank parameter in the SVLS algorithm and recover  $X$ .

### 3.4. Low Rank Approximation

In the low rank matrix approximation problem, the goal is to approximate a (possibly full rank) matrix  $X$  by the closest (in Frobenius norm) rank- $r$  matrix  $X_{(r)}$ . By the Eckart-Young Theorem (Eckart & Young, 1936), this problem has a closed-form solution which is the truncated *SVD* of  $X$ . *SVD* is a powerful tool in affine matrix recovery and different algorithms such as SVT, OptSpace, SVP and others apply *SVD*. In (Halko et al., 2011) the authors try to find a low rank approximation to  $X$  using measurements  $XA^{(C)} = B^{(C)}$  and  $A^{(R)}X = B^{(R)}$ . For large  $n_1, n_2$  they give a single-pass algorithm which computes  $X_{(r)}$  using only  $B^{(C)}$  and  $B^{(R)}$ . We bring their algorithm in the Appendix, Section 7.6. The main difference between the above formulation and our problem in eq. (3) is the rank estimation. In (Halko et al., 2011) it is assumed that  $k^{(R)} = k^{(C)} = k$  and one estimates  $X_{(k)}$  instead of a rank- $r$  matrix which can lead to poor performance if  $r \ll k$ . We adjusted the algorithm presented in (Halko et al., 2011) to our problem and give a new estimator which is a combination of SVLS and (Halko et al., 2011)'s method, replacing  $\hat{X}^{(R)}$  and  $\hat{X}^{(C)}$  in steps 3,4 of SVLS by:

$$\hat{X}_P^{(R)} = \hat{X}^{(R)}\hat{V}\hat{V}^T, \quad \hat{X}_P^{(C)} = \hat{U}\hat{U}^T\hat{X}^{(C)}. \quad (11)$$

Here  $\hat{V}$  is the  $r$  largest right singular vectors of  $B^{(R)}$  and  $\hat{U}$  is the  $r$  largest left singular vectors of  $B^{(C)}$ . We call this new estimator *SVLSP*. Simulations show almost identical and in some cases slightly better performance of this modified algorithm compared to SVLS (see Appendix, Section 7.6). This modified estimator is however difficult to analyze rigorously, and therefore we present throughout the paper our results for the SVLS estimator.

## 4. Performance Guarantees

In this section we give guarantees on the accuracy of the estimator  $\hat{X}$  returned by SVLS. Our guarantees are probabilistic, with respect to randomizing the design matrices  $A^{(R)}, A^{(C)}$ . For the noiseless case we give conditions which are close to optimal for exact recovery.

#### 4.1. Noiseless Case

A rank  $r$  matrix of size  $n_1 \times n_2$  has  $r(n_1 + n_2 - r)$  degrees of freedom, and can therefore not be uniquely recovered by fewer measurements. Setting  $k^{(R)} = k^{(C)} = r$  gives precisely this minimal number of measurements. We next show that this number suffices, with probability 1, to guarantee accurate recovery of  $X$  in the GRC model. In the RCMC model the number of measurements is increased by a logarithmic factor in  $n$  and we need an additional incoherence assumption on  $X$  in order to guarantee accurate recovery with high probability. We first present two Lemmas which will be useful. Their proofs are given in the Appendix, Section 7.1.

**Lemma 1.** *Let  $X_1, X_2 \in \mathcal{M}_{n_1 \times n_2}^{(r)}$  and  $A^{(R)} \in \mathbb{R}_{k^{(R)} \times n_1}, A^{(C)} \in \mathbb{R}_{n_2 \times k^{(C)}}$  such that  $\text{rank}(A^{(R)}X_1) = \text{rank}(X_1A^{(C)}) = r$ . If  $A^{(R)}X_1 = A^{(R)}X_2$  and  $X_1A^{(C)} = X_2A^{(C)}$  then  $X_1 = X_2$ .*

**Lemma 2.** *Let  $X \in \mathcal{M}_{n_1 \times n_2}^{(r)}$  and  $A^{(R)} \in \mathbb{R}_{k^{(R)} \times n_1}, A^{(C)} \in \mathbb{R}_{n_2 \times k^{(C)}}$  such that  $\text{rank}(A^{(R)}X) = \text{rank}(XA^{(C)}) = r$ . For Algorithm 1 with inputs  $A^{(R)}, A^{(C)}, B^{(R,0)}, B^{(C,0)}$  and  $r$  the output  $\hat{X}$  satisfies*

$$A^{(R)}X = A^{(R)}\hat{X}, XA^{(C)} = \hat{X}A^{(C)} \quad (12)$$

##### 4.1.1. EXACT RECOVERY FOR GRC

For the noiseless case, we can recover  $X$  with the minimal number of measurements, as shown in Theorem 1 (proof given in the Appendix, Section 7.1):

**Theorem 1.** *Let  $\hat{X}$  be the output of Algorithm 1 in the GRC model with  $Z^{(C)}, Z^{(R)} = 0$  and  $k^{(R)}, k^{(C)} \geq r$ . Then  $P(\hat{X} = X) = 1$ .*

##### 4.1.2. EXACT RECOVERY FOR RCMC

In the RCMC model, rows and columns of  $X$  are sampled with replacement. Since the same row can be sampled over and over, we cannot guarantee uniqueness of solution, as was the case for the GRC model, but rather wish to prove that exact recovery of  $X$  is possible with high probability. We assume the Bernoulli rows and columns model as described in Section 2 and assume for simplicity that  $k^{(R)} = k^{(C)} = k$ .

**Theorem 2.** *Let  $X = U\Sigma V^T$  be the SVD of  $X \in \mathbb{R}_{n_1 \times n_2}$ , and  $\max(\mu(U), \mu(V)) < \mu$ . Take  $A^{(R)}$  and  $A^{(C)}$  as in the RCMC model without noise and probabilities  $p^{(R)} = \frac{k}{n_1}$  and  $p^{(C)} = \frac{k}{n_2}$ . Let  $\beta > 1$  such that  $C_R \sqrt{\frac{\beta \log(n) r \mu}{k}} < 1$  where  $C_R$  is uniform constant and let  $\hat{X}$  be the output of Algorithm 1. Then  $P(\hat{X} = X) > 1 - 6 \min(n_1, n_2)^{-\beta}$ .*

The proof of Theorem 2 is in the Appendix, Section 7.3.

**Remark 1.** *Both row and column measurements are needed in order to guarantee unique recovery. If, for example, we observe only rows then even with  $n - 1$  observed rows and rank  $r = 1$  we can only determine the unobserved row up to a constant, and thus cannot recover  $X$  uniquely.*

#### 4.2. General (Noisy) Case

In the noisy case we cannot guarantee exact recovery of  $X$ , and our goal is to minimize the error  $\|X - \hat{X}\|_F$  for  $\hat{X}$  the output of SVLS. Here, we give bounds on the error for the GRC model. For simplicity, we show the result for  $k^{(R)} = k^{(C)} = k$ .

We focus on the high dimensional case  $k \leq n$ , where the number of measurements is low. In this case our bound is similar to the bound of the Gaussian Ensemble (GE). In (Candès & Plan, 2011) it is shown for GE that  $\|X - \hat{X}\|_F < C_G \sqrt{\frac{nr\tau^2}{d}}$  holds with high probability for some constant  $C_G$ . We next give an analogous result for our GRC model (proof in the Appendix, Section 7.4).

**Theorem 3.** *Let  $A^{(R)}$  and  $A^{(C)}$  with  $k \geq \max(4r, 40)$  be as in the GRC model with noise matrices  $Z^{(R)}, Z^{(C)}$ . Let  $\hat{X}$  be the output of SVLS. Then there exist constants  $c, c^{(R)}, c^{(C)}$  such that with probability  $> 1 - 5e^{-ck}$ :*

$$\|X - \hat{X}\|_F \leq \sqrt{\frac{r}{k}} \left[ c^{(C)} \|Z^{(C)}\|_2 + c^{(R)} \|Z^{(R)}\|_2 \right]. \quad (13)$$

Theorem 3 applies for any  $Z^{(C)}$  and  $Z^{(R)}$ . If  $k \leq n$  and  $Z^{(R)}, Z^{(C)} \stackrel{i.i.d.}{\sim} N(0, \tau^2)$ , then from eq. (35) we get  $\max(\|Z^{(R)}\|_2, \|Z^{(C)}\|_2) \leq 4\tau\sqrt{n}$  with probability  $1 - e^{-2n}$ . We therefore get the next Corollary for i.i.d. additive Gaussian noise:

**Corollary 1.** *Let  $A^{(R)}, A^{(C)}$  as in the GRC with  $n \geq k \geq \max(4r, 40)$ , model and  $Z^{(R)}, Z^{(C)} \stackrel{i.i.d.}{\sim} N(0, \tau^2)$ . Then there exist constants  $c, C_{GRC}$  such that:*

$$P\left(\|X - \hat{X}\|_F \leq C_{GRC} \sqrt{\frac{\tau^2 nr}{k}}\right) > 1 - 5e^{-ck} - e^{-2n}. \quad (14)$$

## 5. Simulations Results

We studied the performance of our algorithm using simulations. We measured the reconstruction accuracy using the Relative Root-Mean-Squared-Error (RRMSE), defined as

$$RRMSE \equiv RRMSE(X, \hat{X}) \equiv \|X - \hat{X}\|_F / \|X\|_F. \quad (15)$$

For simplicity, we concentrated on square matrices with  $n_1 = n_2 = n$  and used an equal number of row and column measurements,  $k^{(R)} = k^{(C)} = k$ . In all simula-

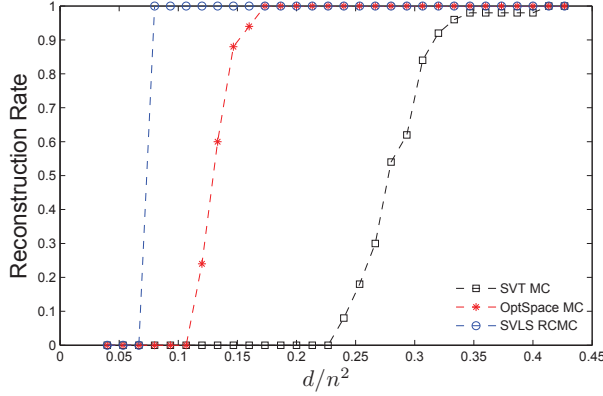


Figure 1. Reconstruction rates for matrices with dimension  $n = 150$  and  $r = 3$  where  $d$  is the number of known entries varied between 0 to 8000. SVT and OptSpace are applied to the standard MC design and Algorithm 1 to RCMC. For each  $d$  we sampled 50 matrices and calculated the reconstruction rate as described in the main text.

tions we sampled a random rank- $r$  matrix  $X = UV^T$  with  $U, V \in \mathbb{R}_{n \times r}$ ,  $U, V \stackrel{i.i.d.}{\sim} N(0, \sigma^2)$ .

In all simulations we assumed that  $\text{rank}(X)$  is unknown and estimated using the elbow method in eq. (10).

### 5.1. Row-Column Matrix Completion (RCMC)

In the noiseless case we compared our design to standard MC. We compared the reconstruction rate (probability of exact recovery of  $X$  as function of the number of measurements  $d$ ) for the RCMC design with SVLS to the reconstruction rate for the standard MC design with the OptSpace (Keshavan et al., 2010) and SVT (Cai et al., 2010) algorithms. To allow for numerical errors, for each simulation yielding  $X$  and  $\hat{X}$  we defined recovery as successful if their  $RRMSE$  was lower than  $10^{-3}$ , and for each value of  $d$  recorded the percentage of simulations for which recovery was successful. In Figure 1 we show results for  $n = 150, r = 3$  and  $\sigma = 1$ . SVLS recovers  $X$  with probability 1 with the optimal number of measurements  $d = r(2n - r) = 894$  yielding  $\frac{d}{n^2} \approx 0.04$  while MC with OptSpace and SVT need roughly 3-fold and 8-fold more measurements, respectively, to guarantee exact recovery.

The improvement in accuracy is not due to our design or our algorithm alone, but due to their combination. We compared our method to OptSpace and SVT for RCMC. We sampled a matrix  $X$  with  $n = 100, r = 3, \sigma = 1$  and noise level  $\tau^2 = 0.25^2$ , and varied the number of row and column measurements  $k$ . Figure 2 shows that while the performance of SVLS is very stable even for small  $k$ , the performance of OptSpace varies, with multiple instances achieving poor accuracy, and SVT which minimizes the nuclear

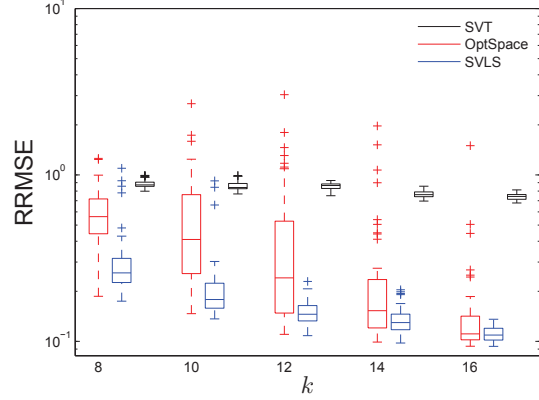


Figure 2. Box-plots represent the distribution of  $RRMSE$  as a function of the number of column and row measurements  $k$  over 50 different sampled matrices  $X = UV^T$  with  $U, V \stackrel{i.i.d.}{\sim} N(0, 1)$  and  $Z^{(R)}, Z^{(C)} \stackrel{i.i.d.}{\sim} N(0, 0.25^2)$ . OptSpace (red) fails to recover  $X$  on many instances while SVLS (blue) performs very well on all of them. SVT (black) fails to recover  $X$  for all instances. The trimming of dense rows and columns in OptSpace was skipped, since such trimming in our settings may delete all measurement information for low  $k$ .

norm achieves poor accuracy for all problem instances.

**Remark 2.** *The OptSpace algorithm has a trimming step which delete dense columns. We omitted this step in the RCMC model since it would delete all the known columns and rows and it's not stable for this type of measurements, but it still get better result than SVT.*

Next, we compared our RCMC to standard MC. We sampled  $X$  as before with  $U, V \in \mathbb{R}_{1000 \times r}$  with standard Gaussian distribution, different rank and different noise ratio. The observations were corrupted by additive Gaussian noise  $Z$  with relative noise level  $NR \equiv \|Z\|_F / \|X\|_F$ .

Results, displayed in Table 1, show that SVLS is significantly faster than the other two algorithms. It is also more accurate than MC for small number of measurements, and comparable to MC for large number of measurements.

Finally, we checked for RCMC and MC our performance only on unobserved entries, to examine if  $RRMSE$  is optimistic due to overfitting to observed entries. Results, shown in the Appendix, Section 7.8, indicate than no overfitting is observed.

### 5.2. Gaussian Rows and Columns (GRC)

We tested the performance of the GRC model with  $A^{(R)}, A^{(C)} \stackrel{i.i.d.}{\sim} N(0, \frac{1}{n})$  and with  $X = UV^T$  where  $U, V \stackrel{i.i.d.}{\sim} N(0, \frac{1}{\sqrt{r}})$ . We compare our results to the Gaussian Ensemble model (GE) where for each  $n$ ,  $\mathcal{A}(X)$  was

Table 1. *RRMSE* and time in seconds (in parenthesis) for SVLS applied to RCMC, and OptSpace and SVT applied to the standard MC. Results represent average of 5 different random matrices. SVLS is faster than OptSpace and SVT by 1 to 3 orders of magnitudes, and shows comparable or better *RRMSE* in all cases.

NR	$d$	$r$	SVLS	OptSpace	SVT
$10^{-2}$	120156	10	0.0063 (0.15)	0.004 (20.8)	0.0073 (18.7)
$10^{-1}$	120156	10	0.064 (0.15)	0.044 (21.7)	0.05 (11)
1	120156	10	0.612 (0.16)	0.49 (24.5)	0.51 (1)
$10^{-2}$	59100	20	0.029 (0.12)	0.97 (25.6)	0.76 (4.4)
$10^{-1}$	59100	20	0.3 (0.12)	0.98 (40.1)	0.86 (6.5)
$10^{-1}$	391600	50	0.081 (0.7)	0.05 (1200)	0.069 (13)
1	391600	50	0.72 (0.6)	0.61 (1300)	0.59 (5)

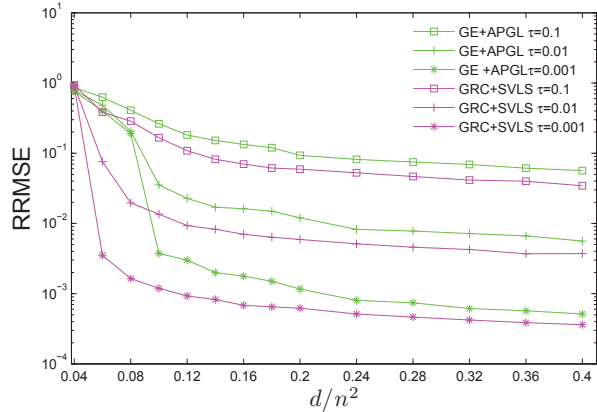


Figure 3. *RRMSE* as function of  $d$ , the number of measurements, where we take  $X \in \mathcal{M}_{100 \times 100}^{(2)}$ ,  $d$  is varied from 400 to 4000 and for different noise levels:  $\tau = 0.1, 0.01$  and  $0.001$ . For every point we simulated 5 random matrices and computed the average *RRMSE*.

normalized to allow a fair comparison. In Figure 3 we take  $n = 100$  and  $r = 2$ , and change the number of measurements  $d = 2nk$  (where  $A^{(R)} \in \mathbb{R}_{k \times n}$  and  $A^{(C)} \in \mathbb{R}_{n \times k}$ ). We added Gaussian noise  $Z^{(R)}, Z^{(C)}$  with different noise levels  $\tau$ . For all noise levels, the performance of GRC was better than the performance of GE. The *RRMSE* error decays at a rate of  $\sqrt{k}$ . For GE we used the APGL algorithm (Toh & Yun, 2010) for nuclear norm minimization.

In the next tests we ran SVLS for measurements with different noise levels. We take  $n = 1000$  and  $k = 100$  with different rank level every entry in  $Z^{(C)}, Z^{(R)} \stackrel{i.i.d.}{\sim} N(0, \tau^2)$  and different values of  $\tau$ . Results are shown in Figure 4. The change in the relative error *RRMSE* is linear in  $\tau$  while the rate depends on  $r$ .

We next examined the behaviour of the *RRMSE* when  $n \rightarrow \infty$  and when  $n, k, r \rightarrow \infty$  together, while the ratios  $\frac{k}{n}$  and  $\frac{d}{r}$  are kept constant. Results (shown in the Appendix, Section 7.5) indicate that when properly scaled, the *RRMSE* error is not sensitive to the value of  $n$  and other

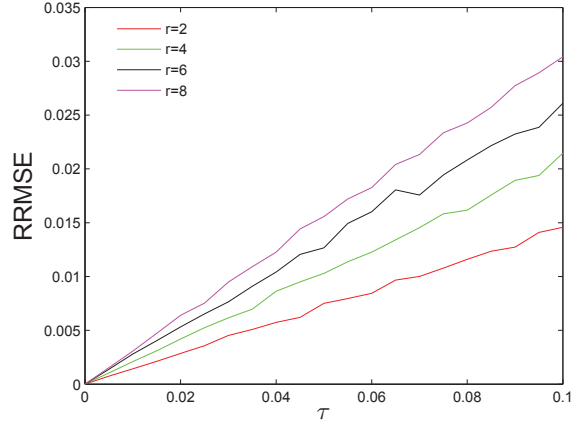


Figure 4. *RRMSE* as a function of noise level  $\tau$  varied from 0 to 0.1, for matrices  $X \in \mathbb{R}_{1000 \times 1000}$  of different ranks. For each curve we fitted a linear regression line, with fitted slopes 0.145, 0.208, 0.25, 0.3 for  $r = 2, 4, 6, 8$ , respectively. The slope is roughly proportional to  $\sqrt{r}$  in concordance with the error bound in Theorem 3. Further investigation of the relation using extensive simulations is required in order to evaluate the dependency of the recovery error in  $r$  in a more precise manner.

parameters, in agreement with Theorem 3.

## 6. Discussion

We introduced a new measurements ensemble for low rank matrix recovery where every measurements is an affine combination of a row or column of  $X$ . We focused on two models: matrix completion from single columns and rows (RCMC) and matrix recovery from Gaussian combination of columns and rows (GRC). We proposed a fast algorithm for this ensemble. For the RCMC model we proved that in the noiseless case our method recovers  $X$  with high probability and simulation results show that the RCMC model outperforms the standard approach for matrix completion in both speed and accuracy for models with small noise.

For the GRC model we proved that our method recovers  $X$  with the optimal number of measurements in the noiseless case and gave an upper bounds on the error for the noisy case. For RCMC, our simulations show that the RCMC design may achieve comparable or favorable results, compared to the standard MC design, especially for low noise level. Proving recovery guarantees for this RCMC model is an interesting future challenge.

Our proposed measurement scheme is not restricted to recovery of low-rank matrices. One can employ this measurement scheme and recover  $X$  by minimizing other matrix norms. This direction can lead to new algorithms that may improve matrix recovery for real datasets.



## References

- Basri, Ronen and Jacobs, David W. Lambertian reflectance and linear subspaces. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(2):218–233, 2003.
- Cai, Jian-Feng, Candès, Emmanuel J, and Shen, Zuowei. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.
- Cai, T Tony and Zhang, Anru. ROP: Matrix recovery via rank-one projections. *The Annals of Statistics*, 43(1):102–138, 2015.
- Candès, Emmanuel J and Plan, Yaniv. Matrix completion with noise. *Proceedings of the IEEE*, 98(6):925–936, 2010.
- Candès, Emmanuel J and Plan, Yaniv. Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. *IEEE Transactions on Information Theory*, 57(4):2342–2359, 2011.
- Candès, Emmanuel J and Recht, Benjamin. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, 2009.
- Candès, Emmanuel J and Romberg, Justin. Sparsity and incoherence in compressive sampling. *Inverse Problems*, 23(3):969, 2007.
- Candès, Emmanuel J and Tao, Terence. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5):2053–2080, 2010.
- Chi, Eric C, Zhou, Hua, Chen, Gary K, Del Vecchio, Diego Ortega, and Lange, Kenneth. Genotype imputation via matrix completion. *Genome Research*, 23(3):509–518, 2013.
- Chu, Ci, Qu, Kun, Zhong, Franklin L, Artandi, Steven E, and Chang, Howard Y. Genomic maps of long noncoding RNA occupancy reveal principles of RNA-chromatin interactions. *Molecular cell*, 44(4):667–678, 2011.
- Dasgupta, Sanjoy and Gupta, Anupam. An elementary proof of a theorem of Johnson and Lindenstrauss. *Random Structures & Algorithms*, 22(1):60–65, 2003.
- Eckart, Carl and Young, Gale. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.
- Gavish, Matan and Donoho, David L. The optimal hard threshold for singular values is  $4/\sqrt{3}$ . *IEEE Transactions on Information Theory*, 60(8):5040–5053, 2014.
- Gross, David, Liu, Yi-Kai, Flammia, Steven T, Becker, Stephen, and Eisert, Jens. Quantum state tomography via compressed sensing. *Physical Review Letters*, 105(15):150401, 2010.
- Halko, Nathan, Martinsson, Per-Gunnar, and Tropp, Joel A. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011.
- Jain, Prateek, Meka, Raghu, and Dhillon, Inderjit S. Guaranteed rank minimization via singular value projection. In *Advances in Neural Information Processing Systems*, pp. 937–945, 2010.
- Keshavan, Raghunandan H, Montanari, Andrea, and Oh, Sewoong. Matrix completion from noisy entries. In *Advances in Neural Information Processing Systems*, pp. 952–960, 2009.
- Keshavan, Raghunandan H, Montanari, Andrea, and Oh, Sewoong. Matrix completion from a few entries. *IEEE Transactions on Information Theory*, 56(6):2980–2998, 2010.
- Koren, Yehuda, Bell, Robert, and Volinsky, Chris. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
- Ma, Shiqian, Goldfarb, Donald, and Chen, Lifeng. Fixed point and Bregman iterative methods for matrix rank minimization. *Mathematical Programming*, 128(1-2):321–353, 2011.
- Recht, Benjamin. A simpler approach to matrix completion. *The Journal of Machine Learning Research*, 12(Dec):3413–3430, 2011.
- Recht, Benjamin, Fazel, Maryam, and Parrilo, Pablo A. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52(3):471–501, 2010.
- Shalev-Shwartz, Shai and Ben-David, Shai. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.
- Szarek, Stanislaw J. Condition numbers of random matrices. *Journal of Complexity*, 7(2):131–149, 1991.
- Talagrand, Michel. New concentration inequalities in product spaces. *Inventiones Mathematicae*, 126(3):505–563, 1996.
- Toh, Kim-Chuan and Yun, Sangwoon. An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. *Pacific Journal of Optimization*, 6(3):615–640, 2010.