Matrix Completion: 
Fundamental Limits and Efficient Algorithms

Sewoong Oh

PhD Defense
Stanford University

July 23, 2010
Matrix completion

- Find the missing entries in a huge data matrix
Example 1. Recommendation systems

- Given less than 1% of the movie ratings
- Goal: Predict missing ratings
Example 2. Positioning

- Only distances between close-by sensors are measured
- Goal: Find the sensor positions up to a rigid motion
More applications:

- Computer vision: Structure-from-motion
- Molecular biology: Microarray
- Numerical linear algebra: Fast low-rank approximations
- etc.
Outline

1. Background

2. Algorithm and main results

3. Applications in positioning
Background
The model

- Rank-\(r\) matrix \(M\)
- Random uniform sample set \(E\)
- Sample matrix \(M^E\)

\[
M_{ij}^E = \begin{cases} 
M_{ij} & \text{if } (i,j) \in E \\
0 & \text{otherwise}
\end{cases}
\]
The model

- Rank-$r$ matrix $M$
- Random uniform sample set $E$
- Sample matrix $M^E$

$$M^E_{ij} = \begin{cases} M_{ij} & \text{if } (i, j) \in E \\ 0 & \text{otherwise} \end{cases}$$
Which matrices?

- Pathological example

\[
M = \begin{bmatrix}
1 & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 \\
\end{bmatrix} = \begin{bmatrix} 1 \\
0 \\
\vdots \\
0 \\
\end{bmatrix} \cdot \begin{bmatrix} 1 \\
0 \\
\vdots \\
0 \\
\end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & \cdots & 0 \\
\end{bmatrix}
\]

Intuition ▶ \( \mu \) is small if singular vectors are well balanced ▶ We need low-coherence for matrix completion
Which matrices?

- Pathological example

\[
M = \begin{bmatrix}
1 & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 \\
\end{bmatrix} = 
\begin{bmatrix}
1 \\
0 \\
\vdots \\
0 \\
\end{bmatrix} \cdot [1] \cdot \begin{bmatrix}
1 \\
0 \\
\vdots \\
0 \\
\end{bmatrix}
\]

- [Candès, Recht '08] \( M = U\Sigma V^T \) has coherence \( \mu \) if

\[
A0. \quad \max_{1 \leq i \leq \alpha n} \sum_{k=1}^{r} U_{ik}^2 \leq \mu \frac{r}{n}, \quad \max_{1 \leq j \leq n} \sum_{k=1}^{r} V_{jk}^2 \leq \mu \frac{r}{n}
\]

\[
A1. \quad \max_{i,j} \left| \sum_{k=1}^{r} U_{ik} V_{jk} \right| \leq \mu \frac{\sqrt{r}}{n}
\]

- Intuition
  - \( \mu \) is small if singular vectors are well balanced
  - We need low-coherence for matrix completion
Previous work

Rank minimization

\[
\begin{align*}
\text{minimize} & \quad \text{rank}(X) \\
\text{subject to} & \quad X_{ij} = M_{ij}, \ (i, j) \in E
\end{align*}
\]

- NP-hard

Heuristic [Fazel '02]

\[
\text{minimize} \quad \|X\|_*
\]

Convex relaxation

Nuclear norm

\[\|X\|_* = \sum_{i=1}^n \sigma_i(X)\]

Can be solved using Semidefinite Programming (SDP)
Previous work

**Rank minimization**

\[
\begin{align*}
\text{minimize} & \quad \text{rank}(X) \\
\text{subject to} & \quad X_{ij} = M_{ij}, \ (i, j) \in E
\end{align*}
\]

- NP-hard

**Heuristic [Fazel '02]**

\[
\begin{align*}
\text{minimize} & \quad \|X\|_* \\
\text{subject to} & \quad X_{ij} = M_{ij}, \ (i, j) \in E
\end{align*}
\]

- Convex relaxation
- Nuclear norm

\[
\|X\|_* = \sum_{i=1}^{n} \sigma_i(X)
\]

- Can be solved using Semidefinite Programming (SDP)
Previous work

- [Candès, Recht ’08]
  - Nuclear norm minimization reconstructs $M$ exactly with high probability, if
    $$|E| \geq C \mu r n^{6/5} \log n$$
  - Surprise?
Previous work

- [Candès, Recht ’08]
  - Nuclear norm minimization reconstructs $M$ exactly with high probability, if
    $$|E| \geq C \mu r \frac{n^{6/5}}{\log n}$$
  - Degrees of freedom $\simeq (1 + \alpha)rn$
  - Open questions
    - Optimality: Do we need $n^{6/5} \log n$ samples?
    - Complexity: SDP is computationally expensive
    - Noise: Can not deal with noise
Previous work

- [Candès, Recht ’08]
  - Nuclear norm minimization reconstructs $M$ exactly with high probability, if
    $$|E| \geq C \mu r n^{6/5} \log n$$
  - Degrees of freedom $\simeq (1 + \alpha)rn$
  - Open questions
    - Optimality: Do we need $n^{6/5} \log n$ samples?
    - Complexity: SDP is computationally expensive
    - Noise: Can not deal with noise

A new approach to Matrix Completion: **OptSpace**
Example: $2000 \times 2000$ rank-8 random matrix

- **low-rank matrix** $M$
- **sampled matrix** $M^E$
- **OptSpace** output $\hat{M}$
- **squared error** $(M - \hat{M})^2$

0.25% sampled
Example: $2000 \times 2000$ rank-8 random matrix

- **low-rank matrix** $M$
- **sampled matrix** $M^E$
- **OPTSPACE output** $\hat{M}$
- **squared error** $(M - \hat{M})^2$

$0.50\%$ sampled
Example: 2000 × 2000 rank-8 random matrix

low-rank matrix $M$

sampled matrix $M^E$

$\text{OptSpace}$ output $\hat{M}$

squared error $(M - \hat{M})^2$

0.75% sampled
Example: $2000 \times 2000$ rank-8 random matrix

- low-rank matrix $M$
- sampled matrix $M^E$
- $\text{OptSpace}$ output $\hat{M}$
- squared error $(M - \hat{M})^2$

1.00% sampled
Example: \(2000 \times 2000\) rank-8 random matrix

- low-rank matrix \(M\)
- sampled matrix \(M^E\)
- \texttt{OptSpace} output \(\hat{M}\)
- squared error \((M - \hat{M})^2\)

1.25% sampled
Example: $2000 \times 2000$ rank-8 random matrix

- low-rank matrix $M$
- sampled matrix $M^E$
- $\text{OptSpace}$ output $\hat{M}$
- squared error $(M - \hat{M})^2$

1.50% sampled
Example: $2000 \times 2000$ rank-8 random matrix

low-rank matrix $M$

sampled matrix $M^E$

$\text{OptSpace}$ output $\hat{M}$

squared error $(M - \hat{M})^2$

1.75% sampled
Naïve approach

- Singular Value Decomposition (SVD)

\[ M^E = \sum_{i=1}^{n} \sigma_i x_i y_i^T \]

- Compute rank-\(r\) approximation \(\hat{M}_{\text{SVD}}\)

\[ \hat{M}_{\text{SVD}} \triangleq \frac{\alpha n^2}{|E|} \sum_{i=1}^{r} \sigma_i x_i y_i^T \]
Naïve approach fails

- **Singular Value Decomposition (SVD)**
  \[ M^E = \sum_{i=1}^{n} \sigma_i x_i y_i^T \]

- Compute rank-\(r\) approximation \(\hat{M}_{\text{SVD}}\)
  \[ \hat{M}_{\text{SVD}} \triangleq \frac{\alpha n^2}{|E|} \sum_{i=1}^{r} \sigma_i x_i y_i^T \]
Naïve approach fails

- Singular Value Decomposition (SVD)

\[ M^E = \sum_{i=1}^{n} \sigma_i x_i y_i^T \]

- Compute rank-\( r \) approximation \( \hat{M}_{\text{SVD}} \)

\[ \hat{M}_{\text{SVD}} \triangleq \frac{\alpha n^2}{|E|} \sum_{i=1}^{r} \sigma_i x_i y_i^T \]
Trimming

\[ M^E = \]

\[ \tilde{M}^E_{ij} = \begin{cases} 
0 & \text{if } \deg(\text{row}_i) > 2|E|/\alpha n \\
0 & \text{if } \deg(\text{col}_j) > 2|E|/n \\
M^E_{ij} & \text{otherwise}
\end{cases} \]

\( \deg(\cdot) \) is the number of samples in that row/column.
Trimming

\[ M^E = \begin{cases} 
0 & \text{if } \deg(\text{row}_i) > 2|E|/\alpha n \\
0 & \text{if } \deg(\text{col}_j) > 2|E|/n \\
M^E_{ij} & \text{otherwise}
\end{cases} \]

\[ \tilde{M}^E_{ij} = \begin{cases} 
0 & \text{if } \deg(\text{row}_i) > 2|E|/\alpha n \\
0 & \text{if } \deg(\text{col}_j) > 2|E|/n \\
M^E_{ij} & \text{otherwise}
\end{cases} \]

deg(\cdot) is the number of samples in that row/column
Trimming

\[ \tilde{M}^E_{ij} = \begin{cases} 
0 & \text{if } \deg(\text{row}_i) > 2|E|/\alpha n \\
0 & \text{if } \deg(\text{col}_j) > 2|E|/n \\
M^E_{ij} & \text{otherwise} 
\end{cases} \]

\(\deg(\cdot)\) is the number of samples in that row/column
Algorithm

**OptSpace**

**Input**: sample indices $E$, sample values $M^E$, rank $r$

**Output**: estimation $\hat{M}$

1: Trimming

2: Compute $\hat{M}_{\text{SVD}}$ using SVD

3: Greedy minimization of the residual error
Algorithm

**OPTSPACE**

**Input** : sample indices $E$, sample values $M^E$, rank $r$

**Output** : estimation $\hat{M}$

1: Trimming
2: Compute $\hat{M}_{SVD}$ using SVD

- $\hat{M}_{SVD}$ can be computed efficiently for sparse matrices
Main results

Theorem

For any $|E|$, $\hat{M}_{\text{SVD}}$ achieves, with high probability,

$$\text{RMSE} \leq CM_{\text{max}} \sqrt{\frac{nr}{|E|}}$$

- $\text{RMSE} = \left( \frac{1}{\alpha n^2} \sum_{i,j} (M - \hat{M}_{\text{SVD}})_{ij}^2 \right)^{1/2}$
- $M_{\text{max}} \triangleq \max_{i,j} |M_{ij}|$

Main results

Theorem

For any $|E|$, $\hat{M}_{\text{SVD}}$ achieves, with high probability,

$$\text{RMSE} \leq CM_{\text{max}} \sqrt{\frac{nr}{|E|}}$$

- [Achlioptas, McSherry ’07]
  If $|E| \geq n(8 \log n)^4$, with high probability,
  $$\text{RMSE} \leq 4M_{\text{max}} \sqrt{\frac{nr}{|E|}}$$

- For $n = 10^5$, $(8 \log n)^4 \approx 7.2 \cdot 10^7$

Main results

Theorem

For any $|E|$, $\hat{M}_{\text{SVD}}$ achieves, with high probability,

$$\text{RMSE} \leq CM_{\text{max}} \sqrt{\frac{nr}{|E|}}$$

- [Achlioptas, McSherry ’07]
  If $|E| \geq n(8 \log n)^4$, with high probability,

  $$\text{RMSE} \leq 4M_{\text{max}} \sqrt{\frac{nr}{|E|}}$$

- For $n = 10^5$, $(8 \log n)^4 \approx 7.2 \cdot 10^7$

Netflix dataset

A single user rated 17,000 movies.

“Miss Congeniality”: 200,000 ratings.

Can we do better?
Greedy minimization of residual error

- Starting from \((X_0, Y_0)\) for \(\hat{M}_{SVD} = X_0 S_0 Y_0^T\), use gradient descent methods to solve

\[
\begin{align*}
\text{minimize} & \quad F(X, Y) \\
\text{subject to} & \quad X^T X = I, \ Y^T Y = I
\end{align*}
\]

\[
F(X, Y) \triangleq \min_{S \in \mathbb{R}^{r \times r}} \sum_{(i, j) \in E} \left( M_{ij}^E - (XSY^T)_{ij} \right)^2
\]

- Can be computed efficiently for sparse matrices
Algorithm

$\text{OptSpace}$

**Input**: sample indices $E$, sample values $M^E$, rank $r$

**Output**: estimation $\hat{M}$

1: Trimming
2: Compute $\hat{M}_{\text{SVD}}$ using SVD
3: Greedy minimization of the residual error
Main results

Theorem (Trimming+SVD)

\( \hat{M}_{\text{SVD}} \) achieves, with high probability,

\[
\text{RMSE} \leq C M_{\text{max}} \sqrt{\frac{nr}{|E|}}
\]

Theorem (Trimming+SVD+Greedy minimization)

\text{OptSpace} reconstructs \( M \) exactly, with high probability, if

\[
|E| \geq C \mu r n \max\{\mu r, \log n\}
\]

OptSpace is order-optimal

**Theorem**

*If μ and r are bounded, OptSpace reconstructs M exactly, with high probability, if*

\[ |E| \geq C n \log n \]

- Lower bound (coupon collector’s problem):
  - If \(|E| \leq C' n \log n\), then exact reconstruction is impossible
**OptSpace** is order-optimal

**Theorem**

If \( \mu \) and \( r \) are bounded, **OptSpace** reconstructs \( M \) exactly, with high probability, if

\[
|E| \geq C n \log n
\]

- **Lower bound (coupon collector’s problem):**
  If \( |E| \leq C' n \log n \), then exact reconstruction is impossible

- **Nuclear norm minimization:**
  [Candès, Recht ’08, Candès, Tao ’09, Recht ’09, Gross et al. ’09]
  If \( |E| \geq C'' n (\log n)^2 \), then exact reconstruction by SDP
Comparison

- $1000 \times 1000$ rank-10 matrix $M$

![Comparison Graph]

$P_{\text{success}}$

0.5

Fundamental Limit

OptSpace

FPCA

SVT

ADMira

Fundamental Limit [Singer, Cucuringu '09], FPCA [Ma, Goldfarb, Chen '09], SVT [Cai, Candès, Shen '08], ADMira [Lee, Bresler '09]
Story so far

- **OptSpace** reconstructs $M$ from a few sampled entries, when $M$ is exactly low-rank and samples are exact

- In reality,
  - $M$ is only approximately low-rank
  - samples are corrupted by noise
The model with noise

- Rank-$r$ matrix $M$
- Random sample set $E$
- Sample noise $Z^E$
- Sample matrix $N^E = M^E + Z^E$
The model with noise

- Rank-$r$ matrix $M$
- Random sample set $E$
- Sample noise $Z^E$
- Sample matrix $N^E = M^E + Z^E$
Main results

Theorem

For $|E| \geq C \mu rn \max\{\mu r, \log n\}$, OptSpace achieves, with high probability,

$$\text{RMSE} \leq C' \frac{n \sqrt{r}}{|E|} \|Z^E\|_2,$$

provided that the RHS is smaller than $\sigma_r(M)/n$.

- $\| \cdot \|_2$ is the spectral norm

**OptSpace** is order-optimal when noise is i.i.d. Gaussian

**Theorem**

For $|E| \geq C \mu r n \max \{\mu r, \log n\}$, **OptSpace** achieves, with high probability,

$$\text{RMSE} \leq C' \sigma_z \sqrt{\frac{r n}{|E|}}$$

provided that the RHS is smaller than $\sigma_r(M)/n$.

- Lower bound: [Candès, Plan ’09]
  $$\text{RMSE} \geq \sigma_z \sqrt{\frac{2 r n}{|E|}}$$
- Trimming + SVD
  $$\text{RMSE} \leq C M_{\text{max}} \sqrt{\frac{r n}{|E|}} + C' \sigma_z \sqrt{\frac{r n}{|E|}}$$

  - missing entries
  - sample noise
Comparison

- 500 × 500 rank-4 matrix M, Gaussian noise with $\sigma_z = 1$
- Example from [Candès, Plan '09]
Comparison

- $500 \times 500$ rank-4 matrix $M$, Gaussian noise with $\sigma_z = 1$
- Example from [Candès, Plan '09]

Comparison of various methods for sampling rates and RMSE. The graph shows the comparison of FPCA [Ma, Goldfarb, Chen '09], ADMiRA [Lee, Bresler '09], and OptSpace.
Positioning
The model

- $n$ wireless devices uniformly distributed in a bounded convex region
- Distance measurements between devices within radio range $R$
- Find the locations up to a rigid motion
The model

How is it related to Matrix Completion?
- Need to find the missing entries
- \( \text{rank}(D) = 4 \)

How is it different?
- Non-uniform sampling
- Rich information not used in Matrix Completion
The model

- **MDS-MAP** [Shang et al. '03]
  1. Fill in the missing entries with shortest paths
  2. Compute rank-4 approximation
Main results

Theorem

For $R > C \sqrt{\frac{\log n}{n}}$, with high probability,

$$\text{RMSE} \leq \frac{C}{R} \sqrt{\frac{\log n}{n}} + o(1).$$

- $\text{RMSE} = \left( \frac{1}{n^2} \sum_{i,j} (D - \hat{D})_{ij}^2 \right)^{1/2}$

- Lower Bound:
  If $R < \sqrt{\frac{\log n}{\pi n}}$, then the graph is disconnected

- Generalized to quantized measurements and distributed algorithms

- We can add Greedy Minimization step

Karbasi, Oh, *ACM SIGMETRICS*, 2010
Numerical simulation

\[ R/\sqrt{n} \]
Conclusion

- Matrix Completion is an important problem with many practical applications

- \textsc{OptSpace} is an efficient algorithm for Matrix Completion

- \textsc{OptSpace} achieves performance close to the fundamental limit
Special thanks to:

PMCRQOAVNBETHRDAZOKNWXAOOURHLIO
Special thanks to:

PMCRQOAVNBETHRDAZOKNWXAOURHLIO
Special thanks to:

PMCRQOAVNBETHRDAZOKNWXAOEURHLO
Special thanks to:

PMCRQOAVNBETHRDAZOKNWXAOURHLIO
Special thanks to:

PMCRQOAVNBE
THRDAZOKNWXAO
URHLIO
Special thanks to:

- Officemates: Morteza, Yash, Jose, Raghu, Satish
- Friends: Mohsen, Adel, Farshid, Fernando, Arian, Haim, Sachin, Ivana, Ahn, Cha, Choi, Rhee, Kang, Kim, Lee, Na, Park, Ra, Seok, Song
Special thanks to:

- My family and Kyung Eun
Thank you!