Statistical Guarantees for Alternating Minimization

Praneeth Netrapalli

UT Austin

Joint work with Alekh Agarwal, Anima Anandkumar, Prateek Jain, Sujay Sanghavi and Rashish Tandon
Matrix completion: theory and practice

Theory: algorithm based on convex relaxation [CR09, CT09],...

Practice: alternating guesses for user preferences / movie features [Kor09]

Disconnect between theoretical and practical algorithms

Our work: understand why heuristics used in practice work so well
Alternating minimization (AltMin)

\[ \min_{U,V} f(U, V) \]

1. Choose a random $U$
2. Fix $U$ and optimize over $V$
3. Fix $V$ and optimize over $U$
4. Repeat steps 2 and 3

In many cases, steps 2 and 3 efficiently solvable

**Empirically:** widely used e.g., k-means; has good performance

**Theoretically:** little understood, (very) few performance guarantees
Our work

Theoretical performance guarantees for AltMin for:

1. Matrix completion (STOC 2013)
2. Phase retrieval (NIPS 2013)
3. Sparse coding
Key challenge

![Graph showing phase retrieval and objective value over iterations](image)

- Objective value vs. # iterations
- Sufficient conditions for decay
- Principlized initialization

Our approach

Praneeth Netrapalli (UT Austin)
Key challenge

Our approach
- Sufficient conditions for decay
- Principled initialization
This talk

Theoretical performance guarantees for AltMin for:

1. Matrix completion (STOC 2013)
2. Phase retrieval (NIPS 2013)
3. Sparse coding
Sparse coding/Dictionary learning

\[ Y = A^* X^* \]

Examples

Dictionary

Coefficients

Applications: Image compression, denoising, inpainting, ...
Feature learning using sparse coding

Theory: Classifier is a **nice** function of features
Practice: Have text data, pixel values in images etc.
Challenge: Learn features from data
Sparse coding, one such approach [YYGH09, GTCZ10]
Prior art

**Empirical**

- Extensive image processing applications: compression, denoising, inpainting, classification, …
- Several heuristics: K-SVD, Online dictionary learning

**Theory**

- No guarantees on above heuristics
- Recovery in the undercomplete setting using LP [SWW12]

**Our results**: First exact recovery in the overcomplete setting

Part of our results (and more) obtained independently by [AGM13]
AltMin for sparse coding

\[
\min_{A, X} \| Y - AX \|_F^2 \quad \text{s.t.} \quad X \text{ is sparse}
\]
AltMin for sparse coding

\[
\min_{A,X} \| Y - AX \|_F^2 \quad \text{s.t.} \quad X \text{ is sparse}
\]

1. Choose an initial \( A^{(0)} \)
2. In iteration \( t \), do:
   - estimate coefficients \( X^{(t)} \) (sparse recovery)
   - estimate dictionary \( A^{(t)} \) (least squares)
AltMin for sparse coding

\[
\min_{A, X} \| Y - AX \|_F^2 \quad \text{s.t.} \quad X \text{ is sparse}
\]

1. Choose an initial \( A^{(0)} \)

2. In iteration \( t \), do:
   - estimate coefficients \( X^{(t)} \) (sparse recovery)
   - estimate dictionary \( A^{(t-1)} \) (least squares)

Minimize constraining sparsity
AltMin for sparse coding

\[
\min_{A,X} \| Y - AX \|_F^2 \quad \text{s.t.} \quad X \text{ is sparse}
\]

1. Choose an initial \( A^{(0)} \)
2. In iteration \( t \), do:
   - estimate coefficients \( X^{(t)} \) (sparse recovery)
   - estimate dictionary \( A^{(t)} \) (least squares)

Minimize constraining sparsity
AltMin for sparse coding

\[
\min_{A,X} \| Y - AX \|_F^2 \quad \text{s.t.} \quad X \text{ is sparse}
\]

1. Choose an initial \( A^{(0)} \)
2. In iteration \( t \), do:
   - estimate coefficients \( X^{(t)} \) (sparse recovery)
   - estimate dictionary \( A^{(t)} \) (least squares)

\[ X_2^{(t)} \leftarrow \begin{bmatrix} Y_2 \end{bmatrix} - A^{(t-1)} \begin{bmatrix} X_2 \end{bmatrix} \]

Minimize constraining sparsity

\[ F \begin{bmatrix} 2 \\ \end{bmatrix} \]
AltMin for sparse coding

\[
\min_{A, X} \| Y - AX \|_F^2 \quad \text{s.t.} \quad X \text{ is sparse}
\]

1. Choose an initial \( A^{(0)} \)
2. In iteration \( t \), do:
   - estimate coefficients \( X^{(t)} \) (sparse recovery)
   - estimate dictionary \( A^{(t)} \) (least squares)

Minimize constraining sparsity
AltMin for sparse coding

\[
\min_{A,X} \| Y - AX \|_F^2 \quad \text{s.t.} \quad X \text{ is sparse}
\]

1. Choose an initial \( A^{(0)} \)
2. In iteration \( t \), do:
   - estimate coefficients \( X^{(t)} \) (sparse recovery)
     \[
     X_i^{(t)} \leftarrow \arg \min_{X_i} \| Y_i - A^{(t-1)} X_i \|_F^2 + \lambda \| X_i \|_1
     \]
   - estimate dictionary \( A^{(t)} \) (least squares)
AltMin for sparse coding

\[
\min_{A, X} \| Y - AX \|_F^2 \quad \text{s.t.} \quad X \text{ is sparse}
\]

1. Choose an initial \( A^{(0)} \)

2. In iteration \( t \), do:
   - estimate coefficients \( X^{(t)} \) (sparse recovery)
   - estimate dictionary \( A^{(t)} \) (least squares)
AltMin for sparse coding

\[
\min_{A,X} \|Y - AX\|_F^2 \quad \text{s.t.} \quad X \text{ is sparse}
\]

1. Choose an initial \(A^{(0)}\)
2. In iteration \(t\), do:
   - estimate coefficients \(X^{(t)}\) (sparse recovery)
   - estimate dictionary \(A^{(t)}\) (least squares)

\[
A^{(t)} \leftarrow \arg \min_A \|Y - AX^{(t)}\|_F^2
\]
A small modification

1. Choose an initial $A^{(0)}$
2. In iteration $t$, do:
   - estimate coefficients (sparse recovery)
     - threshold coefficients smaller than $\epsilon_t$ to 0
   - estimate dictionary (least squares)

Our results will be for this modified algorithm.
Incoherent dictionaries

Generative model: \( Y = A^*X^* \)

\( A^* \) is incoherent if

\[ \left| \langle A^*_i, A^*_j \rangle \right| \text{ is small for all } i \neq j \]

Natural assumption to enforce well-posedness

Widely used assumption in a lot of prior work [DE03, DET06, BDE09]
Initialization algorithm: main idea

Approach

- Find lots of examples sharing a common dictionary element
- Take top singular vector of those examples

\[
\begin{align*}
Y & \quad A^* \\
\text{Top singular vector} & \quad X^*
\end{align*}
\]
Initialization algorithm: main idea

Approach

- Find lots of examples sharing a common dictionary element
  - find a pair sharing this single dictionary element
- Take top singular vector of those examples

\[ Y \quad A^* \quad X^* \]

\[ \text{Top singular vector} \]
Initialization algorithm: main idea

Definition (Correlation graph)
- one node for each example
Initialization algorithm: main idea

Definition (Correlation graph)
- one node for each example
- edge \{Y_i, Y_j\} if \|Y_i, Y_j\| \geq \rho.

Finding pair sharing a single dictionary element
- edge between \(Y_i\) and \(Y_j\) ⇒ common dictionary element
- each dictionary element corresponds to a cluster
  for each edge, look at edge density among common neighbors
Initialization algorithm: main idea

**Definition (Correlation graph)**
- one node for each example
- edge \{Y_i, Y_j\} if |⟨Y_i, Y_j⟩| ≥ ρ.

**Finding pair sharing a single dictionary element**
- edge between \(Y_i\) and \(Y_j\) \(⇒\) common dictionary element
Initialization algorithm: main idea

Definition (Correlation graph)
- one node for each example
- edge $\{Y_i, Y_j\}$ if $|\langle Y_i, Y_j \rangle| \geq \rho$.

Finding pair sharing a single dictionary element
- edge between $Y_i$ and $Y_j \Rightarrow$ common dictionary element
- each dictionary element corresponds to a cluster
Initialization algorithm: main idea

Definition (Correlation graph)
- one node for each example
- edge \{Y_i, Y_j\} if |\langle Y_i, Y_j \rangle| \geq \rho.

Finding pair sharing a single dictionary element
- edge between Y_i and Y_j \implies common dictionary element
- each dictionary element corresponds to a cluster
- for each edge, look at edge density among common neighbors
Initialization algorithm: main idea

**Definition (Correlation graph)**
- one node for each example
- edge \( \{Y_i, Y_j\} \) if \( |\langle Y_i, Y_j \rangle| \geq \rho \).

**Finding pair sharing a single dictionary element**
- edge between \( Y_i \) and \( Y_j \) \( \Rightarrow \) common dictionary element
- each dictionary element corresponds to a cluster
- for each edge, look at edge density among common neighbors
Initialization algorithm: main idea

Definition (Correlation graph)

- one node for each example
- edge \( \{Y_i, Y_j\} \) if \( |\langle Y_i, Y_j \rangle| \geq \rho \).

Finding pair sharing a single dictionary element

- edge between \( Y_i \) and \( Y_j \) ⇒ common dictionary element
- each dictionary element corresponds to a cluster
- for each edge, look at edge density among common neighbors

Similar algorithm developed simultaneously and independently [AGM13].
Problem parameters

**Parameters**

- **$d$:** ambient dimension
- **$n$:** # of examples
- **$r$:** # of dictionary elements
- **$s$:** sparsity of each example in dictionary repsn.

\[ Y \quad \quad A^* \quad \quad X^* \]

\[ d \times n \quad d \times r \quad r \times n \]

Overcomplete: $r > d$
Assumptions

**Generative model:** \( Y = A^* X^* \)

**Dictionary**

**Coefficients**
Assumptions

Generative model: \( Y = A^*X^* \)

Dictionary

- **Pairwise incoherence**: \( |\langle A_i^*, A_j^* \rangle| < \frac{\mu_0}{\sqrt{d}} \ \forall \ i \neq j \)

- **Spectral norm bound**: \( \|A^*\|_2 < \mu_1 \sqrt{\frac{r}{d}} \)

Coefficients

- Uniform support: each column uniformly random
- \( s \)-sparse iid coefficients
- Lower and upper bounds: \( m < \|X^*\|_1 < M \ \forall \text{non-zero} \ X^*_{ij} \)
Assumptions

Generative model: \( Y = A^* X^* \)

Dictionary

- **Pairwise incoherence**: \( \left| \langle A^*_i, A^*_j \rangle \right| < \frac{\mu_0}{\sqrt{d}} \quad \forall \ i \neq j \)
- **Spectral norm bound**: \( \|A^*\|_2 < \mu_1 \sqrt{\frac{r}{d}} \)

Coefficients

- **Uniform support**: each column uniformly random \( s \)-sparse
- **iid coefficients**: each non-zero element iid
- **Lower and upper bounds**: \( m < \left| X^*_{ij} \right| < M \quad \forall \text{ non-zero } X^*_{ij} \)
Our result

Theorem ([AAN13, AAJ⁺13])

Suppose

1. **Initialization**: $A^{(0)}$ is obtained using Algorithm 1 of AAN’13,
2. **Sparsity**: $s = o\left(d^{\frac{1}{9}}\right)$,
3. **Choice of thresholds**: $\epsilon_0 = \frac{c}{s^2}$, $\epsilon_{t+1} = \frac{c\mu_1s^3}{\sqrt{d}}\epsilon_t$, and

then

- Sample complexity: $\mathcal{O}(r^2 \log r)$, and
- Exponential convergence
Proof outline

Step I: \[ X_i^{(t+1)} \leftarrow \arg \min_{X_i} \| Y_i - A^{(t)} X_i \|_2^2 + \lambda \| X_i \|_1 \]
Proof outline

Step I: $X_i^* = \arg \min_{X_i} \| Y_i - A(t)^{X_i} \|_2^2 + \lambda \| X_i \|_1$

Thresholding $\Rightarrow \text{Supp}(X_{t+1}) \subseteq \text{Supp}(X^*)$

Step II: $A(t+1) \leftarrow \arg \min_{A^*} \| Y_i - A^{(t)} X_{t+1} \|_2^2$

Supp($X_{t+1}$) $\subseteq$ Supp($X^*$)

$A_{t+1} \much closer to A^*$
Proof outline

Step I: \[ X_i^* = \arg \min_{X_i} \left\| Y_i - A^{(t)} X_i \right\|_2^2 + \lambda \| X_i \|_1 \]

\[ A^{(t)} \simeq A^* \Rightarrow X^{(t+1)} \simeq X^* \]
Proof outline

Step I: \[ X_i^* = \arg\min_{X_i} \left\| Y_i - A(t)^{x_i} X_i \right\|_2^2 + \lambda \| X_i \|_1 \]

\[ A(t) \approx A^* \Rightarrow X^{(t+1)} \approx X^* \]

Thresholding \( \Rightarrow \) \( \text{Supp} \left( X^{(t+1)} \right) \subseteq \text{Supp} \left( X^* \right) \)
Proof outline

**Step I:** \( X_i^* = \arg \min_{X_i} \left\| Y_i - A^{(t)} X_i \right\|_2^2 + \lambda \| X_i \|_1 \)

\[
A^{(t)} \approx A^* \implies X^{(t+1)} \approx X^*
\]

Thresholding \( \Rightarrow \) \( \text{Supp} \left( X^{(t+1)} \right) \subseteq \text{Supp} \left( X^* \right) \)

**Step II:** \( A^{(t+1)} \leftarrow \arg \min_{A} \left\| Y - AX^{(t+1)} \right\|_F^2 \)

\[
X^{(t+1)} \approx X^*
\]

\[
\text{Supp} \left( X^{(t+1)} \right) \subseteq \text{Supp} \left( X^* \right)
\]

\( \} \Rightarrow A^{(t+1)} \text{ much closer to } A^* \)
Practical variation

**Issue in initialization**
Dictionary elements with low coefficients not found

**Practical version**

1. **Clustering on residuals**: extract a (small) batch of dict. atoms
2. **AltMin on examples**: use the entire dictionary learnt so far
3. Recompute the residuals and repeat
Reconstructions on “mnist”

**mnist**: 60K images of handwritten digits

Original images

Reconstructed images
Learned dictionaries on “mnist”
Conclusion

Summary

- AltMin: popular empirical approach, usually good performance
- Our work
  - first theoretical guarantees for
    - Matrix completion
    - Phase retrieval
    - Sparse coding
  - initialization schemes
- Future directions
  - General understanding of AltMin
  - Why does random initialization work so well in practice?
  - Use these intuitions to develop more efficient and robust algorithms
- Ongoing work on image classification on ImageNet
Conclusion

Summary

- **AltMin**: popular empirical approach, usually good performance
- **Our work**
  - first theoretical guarantees for
    - Matrix completion
    - Phase retrieval
    - Sparse coding
  - initialization schemes

Future directions

- General understanding of AltMin
- Why does random initialization work so well in practice?
- Use these intuitions to develop more efficient and robust algorithms
  - Ongoing work on image classification on ImageNet
References

Alekh Agarwal, Animashree Anandkumar, Prateek Jain, Praneeth Netrapalli, and Rashish Tandon.
Learning sparsely used overcomplete dictionaries via alternating minimization.

Alekh Agarwal, Animashree Anandkumar, and Praneeth Netrapalli.
Exact recovery of sparsely used overcomplete dictionaries.

Sanjeev Arora, Rong Ge, and Ankur Moitra.
New algorithms for learning incoherent and overcomplete dictionaries.

Alfred M Bruckstein, David L Donoho, and Michael Elad.
From sparse solutions of systems of equations to sparse modeling of signals and images.

Emmanuel J. Candès and Benjamin Recht.
Exact matrix completion via convex optimization.

Emmanuel J. Candès and Terence Tao.
The power of convex relaxation: Near-optimal matrix completion.

David L Donoho and Michael Elad.
Optimally sparse representation in general (nonorthogonal) dictionaries via 1 minimization.

David L Donoho, Michael Elad, and Vladimir N Temlyakov.
Stable recovery of sparse overcomplete representations in the presence of noise.

Shenghua Gao, Ivor Waihung Tsang, Liang-Tien Chia, and Peilin Zhao.