The Projected Power Method: An Efficient Algorithm for Joint Alignment from Pairwise Differences

Yuxin Chen  Emmanuel Candès

Department of Statistics, Stanford University, Sep. 2016
Nonconvex optimization is everywhere

For instance, maximum likelihood estimation is nonconvex in numerous problems

\[
\begin{align*}
\text{maximize}_x & \quad \ell(x; y) \\
\text{subject to} & \quad x \in S
\end{align*}
\]

- matrix completion
- phase retrieval
- dictionary learning
- blind deconvolution
- robust PCA
- ...
Recent flurry of research in nonconvex procedures

Nice geometry within a neighborhood around $x$ (basin of attraction)

Keshavan et al'08, Netrapalli et al'13, Candès et al'14, Soltanolkotabi'14, Jain et al'14, Sun et al'14, Chen et al'15, Cai et al'15, Tu et al'15, Sun et al'15, White et al'15, Li et al'16, Yi et al'16, Zhang et al'16, Wang et al'16, ...
Recent flurry of research in nonconvex procedures

Nice geometry within a neighborhood around $x$ (basin of attraction)

Suggests two-stage paradigms

1. Start from an appropriate initial point

Keshavan et al’08, Netrapalli et al’13, Candès et al’14, Soltanolkotabi’14, Jain et al’14, Sun et al’14, Chen et al’15, Cai et al’15, Tu et al’15, Sun et al’15, White et al’15, Li et al’16, Yi et al’16, Zhang et al’16, Wang et al’16, ...
Recent flurry of research in nonconvex procedures

Nice geometry within a neighborhood around $x$ (basin of attraction)

Suggests two-stage paradigms

1. Start from an appropriate initial point
2. Proceed via some iterative updates

Keshavan et al’08, Netrapalli et al’13, Candès et al’14, Soltanolkotabi’14, Jain et al’14, Sun et al’14, Chen et al’15, Cai et al’15, Tu et al’15, Sun et al’15, White et al’15, Li et al’16, Yi et al’16, Zhang et al’16, Wang et al’16, ...
This talk: a discrete nonconvex problem
Joint alignment from pairwise differences

- $n$ unknown variables: $x_1, \cdots, x_n$
- $m$ possible states: $x_i \in \{1, 2, \cdots, m\}$

\[ x_1 = 1 \quad x_2 = 6 \quad x_3 = 12 \]
Joint alignment from pairwise differences

- $n$ unknown variables: $x_1, \cdots, x_n$
- $m$ possible states: $x_i \in \{1, 2, \cdots, m\}$
Joint alignment from pairwise differences

- $n$ unknown variables: $x_1, \cdots, x_n$
- $m$ possible states: $x_i \in \{1, 2, \cdots, m\}$
Joint alignment from pairwise differences

- **Measurements:** pairwise differences

\[
y_{i,j}^{\text{ind.}} = x_i - x_j + \eta_{i,j} \mod m, \quad i \neq j
\]

- \( \eta_{i,j} \): noise

\( x_i - x_j \mod m \)

---

*Bandiera, Charikar, Singer, Zhu '13; Chen, Guibas, Huang '14*
Joint alignment from pairwise differences

• **Measurements:** pairwise differences

\[ y_{i,j}^{\text{ind.}} = x_i - x_j + \eta_{i,j} \mod m, \quad i \neq j \]

- e.g. random corruption model

\[ y_{i,j}^{\text{ind}} = \begin{cases} 
  x_i - x_j \mod m & \text{with prob. } \pi_0 \\
  \text{Uniform}(m) & \text{else}
\end{cases} \]

- \( \pi_0 \): non-corruption rate

---

[Bandiera, Charikar, Singer, Zhu '13; Chen, Guibas, Huang '14]
Joint alignment from pairwise differences

- Measurements: pairwise differences
  \[ y_{i,j} \overset{\text{ind.}}{=} x_i - x_j + \eta_{i,j} \mod m, \quad i \neq j \]

  - e.g. random corruption model
    \[ x_i - x_j \mod m \]
    \[ x_i \mod m \]

  - \( \pi_0 \): non-corruption rate

- Goal: recover \( \{x_i\} \) (up to global offset)

\[ \text{Bandiera, Charikar, Singer, Zhu '13; Chen, Guibas, Huang '14} \]
Motivation: multi-image alignment

Jointly align a collection of images/shapes of the same physical object
Motivation: multi-image alignment

Jointly align a collection of images/shapes of the same physical object

- $x_i$: angle of rotation associated with each shape
Motivation: multi-image alignment

**Step 1:** compute pairwise estimates of relative angles of rotations

...
Motivation: multi-image alignment

**Step 1:** compute pairwise estimates of relative angles of rotations

**Step 2:** aggregate these pairwise information for joint alignment
Maximum likelihood estimates (MLE)

\[
\begin{align*}
\text{maximize} \{x_i\} & \quad \sum_{i,j} \ell(x_i, x_j; y_{i,j}) \\
\text{subj. to} & \quad x_i \in \{1, \cdots, m\}, \quad 1 \leq i \leq n
\end{align*}
\]

- Log-likelihood function $\ell$ may be complicated
Maximum likelihood estimates (MLE)

\[
\text{maximize}_{\{x_i\}} \sum_{i,j} \ell (x_i, x_j; y_{i,j}) \\
\text{subj. to} \quad x_i \in \{1, \ldots, m\}, \quad 1 \leq i \leq n
\]

- Log-likelihood function \( \ell \) may be complicated
- Discrete input space
Maximum likelihood estimates (MLE)

$$\text{maximize}_{\{x_i\}} \sum_{i,j} \ell (x_i, x_j; y_{i,j})$$

subj. to $$x_i \in \{1, \cdots, m\}, \quad 1 \leq i \leq n$$

- Log-likelihood function $\ell$ may be complicated
- Discrete input space
- Looks daunting
Another look in lifted space

Discrete variables $\rightarrow$ orthogonal vectors in higher-dimensional space

\[ x_i = 1 \iff x_i = e_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \]

\[ x_i = 2 \iff x_i = e_1 = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} \]

\[ \vdots \]

\[ x_i = j \iff x_i = e_j \]
Another look in lifted space

Pairwise sample $y_{i,j}$ $\rightarrow$ encode $\ell (x_i, x_j)$ by $L_{i,j} \in \mathbb{R}^{m \times m}$

$$[L_{i,j}]_{\alpha, \beta} = \ell (x_i = \alpha, x_j = \beta)$$
Another look in lifted space

Pairwise sample $y_{i,j} \rightarrow$ encode $\ell(x_i, x_j)$ by $L_{i,j} \in \mathbb{R}^{m \times m}$

$$[L_{i,j}]_{\alpha,\beta} = \ell(x_i = \alpha, x_j = \beta)$$

- e.g. random corruption model

$$y_{i,j} = \begin{cases} x_i - x_j, & \text{w.p. } \pi_0 \\ \text{Unif}(m), & \text{else} \end{cases} \Rightarrow \ell(x_i, x_j) = \begin{cases} \log(\pi_0 + \frac{1-\pi_0}{m}), & \text{if } x_i - x_j = y_{i,j} \\ \log\left(\frac{1-\pi_0}{m}\right), & \text{else} \end{cases}$$
Another look in lifted space

Pairwise sample $y_{i,j} \rightarrow$ encode $\ell(x_i, x_j)$ by $L_{i,j} \in \mathbb{R}^{m \times m}$

$$[L_{i,j}]_{\alpha,\beta} = \ell(x_i = \alpha, x_j = \beta)$$

- e.g. random corruption model

$$y_{i,j} = \begin{cases} x_i - x_j, & \text{w.p. } \pi_0 \\ \text{Unif}(m), & \text{else} \end{cases} \Rightarrow \ell(x_i, x_j) = \begin{cases} \log(\pi_0 + \frac{1-\pi_0}{m}), & \text{if } x_i - x_j = y_{i,j} \\ \log\left(\frac{1-\pi_0}{m}\right), & \text{else} \end{cases}$$

$\ell(x_i = 2, x_j = 5; y_{i,j} = 2)$
Another look in lifted space

Pairwise sample $y_{i,j} \rightarrow$ encode $\ell(x_i, x_j)$ by $L_{i,j} \in \mathbb{R}^{m \times m}$

$$\left[ L_{i,j} \right]_{\alpha,\beta} = \ell(x_i = \alpha, x_j = \beta)$$

- e.g. random corruption model

$$y_{i,j} = \begin{cases} x_i - x_j, & \text{w.p. } \pi_0 \\ \text{Unif}(m), & \text{else} \end{cases} \Rightarrow \ell(x_i, x_j) = \begin{cases} \log(\pi_0 + \frac{1-\pi_0}{m}), & \text{if } x_i - x_j = y_{i,j} \\ \log(\frac{1-\pi_0}{m}), & \text{else} \end{cases}$$

This enables quadratic representation

$$\ell(x_i, x_j) = x_i^\top L_{i,j} x_j$$
MLE is equivalent to a binary quadratic program
MLE is equivalent to a binary quadratic program

\[
\begin{align*}
\text{maximize} & \quad \sum_{i,j} \ell(x_i - x_j; y_{ij}) \\
\text{subj. to} & \quad x_i \in \{1, \cdots, m\}
\end{align*}
\]

\[
L = \begin{bmatrix}
L_{1,1} & \cdots & L_{1,n} \\
\vdots & \ddots & \vdots \\
L_{n,1} & \cdots & L_{n,n}
\end{bmatrix}
\]

\[
\begin{align*}
\text{maximize}_x & \quad x^\top Lx \\
\text{subj. to} & \quad x = \begin{bmatrix}
x_1 \\
\vdots \\
x_n
\end{bmatrix} \\
x_i & \in \{e_1, \cdots, e_m\}
\end{align*}
\]

This is essentially nonconvex constrained PCA
MLE is equivalent to a binary quadratic program

\[
\text{maximize} \quad \sum_{i,j} \ell(x_i - x_j; y_{ij})
\]
\[
\text{subj. to} \quad x_i \in \{1, \cdots, m\}
\]

This is essentially nonconvex constrained PCA
How to solve nonconvex constrained PCA?

**PCA**

maximize \( x^\top L x \)

subj. to \( \|x\| = 1 \)

**Power method:**

for \( t = 1, 2, \cdots \)

\[
\begin{align*}
  z^{(t)} &= L z^{(t-1)} \\
  z^{(t)} &\leftarrow \text{normalize } (z^{(t)}) \end{align*}
\]
How to solve nonconvex constrained PCA?

**PCA**

<table>
<thead>
<tr>
<th>maximize ( x )</th>
<th>( x^\top Lx )</th>
</tr>
</thead>
<tbody>
<tr>
<td>subj. to</td>
<td>( |x| = 1 )</td>
</tr>
</tbody>
</table>

**Constrained PCA**

<table>
<thead>
<tr>
<th>maximize ( x )</th>
<th>( x^\top Lx )</th>
</tr>
</thead>
<tbody>
<tr>
<td>subj. to</td>
<td>( x_i \in {e_1, \ldots, e_m} )</td>
</tr>
</tbody>
</table>

**Power method:**

for \( t = 1, 2, \ldots \)

\[
z^{(t)} = Lz^{(t-1)}
\]

\( z^{(t)} \leftarrow \text{normalize} \ (z^{(t)})

**Projected power method:**

for \( t = 1, 2, \ldots \)

\[
z^{(t)} = Lz^{(t-1)}
\]

\( z^{(t)} \leftarrow \text{Project}_{\Delta^n} (\mu z^{(t)})

- \( \mu \): scaling factor
Projection onto standard simplex

$$\text{maximize}_{x = \{x_i\}} \quad x^\top Lx \quad \text{s.t.} \quad x_i \in \{e_1, \cdots, e_m\}$$

$$z^{(t)} = Lz^{(t-1)}$$

$$z^{(t)} \leftarrow \text{Project}_{\Delta^n} (\mu z^{(t)})$$
Projection onto standard simplex

\[
\begin{align*}
\text{maximize } & \quad x = \{x_i\} \quad x^\top L x \\
\text{s.t.} & \quad x_i \in \{e_1, \ldots, e_m\}
\end{align*}
\]

\[
\begin{align*}
Lz(t) &= z(t-1) \\
z(t) &\leftarrow \text{Project}_{\Delta^n} (\mu z(t))
\end{align*}
\]

$\Delta^n$ is convex hull of feasibility set, i.e.
\[
\left\{ z = [z_i]_{1 \leq i \leq n} \mid \forall i: \ 1^\top z_i = 1; \ z_i \geq 0 \right\}
\]
Projected power method: $z^{(t+1)} \leftarrow \text{Project}_{\Delta n} (\mu L z^{(t)})$
Projected power method: $z^{(t+1)} \leftarrow \text{Project}_{\Delta^n} \left( \mu L z^{(t)} \right)$

- $Lz$ is gradient of $\frac{1}{2} z^T L z$
Projected power method: \( z^{(t+1)} \leftarrow \text{Project}_{\Delta^n} (\mu Lz^{(t)}) \)

- \( Lz \) is gradient of \( \frac{1}{2} z^\top Lz \)
Projected power method: \( z^{(t+1)} \leftarrow \text{Project}_{\Delta n} \left( \mu Lz^{(t)} \right) \)

- \( Lz \) is gradient of \( \frac{1}{2} z^\top Lz \)
Projected power method: $z^{(t+1)} \leftarrow \text{Project}_{\Delta n} \left( \mu L z^{(t)} \right)$

- $Lz$ is gradient of $\frac{1}{2} z^\top L z$
Illustration

Projected power method: $z^{(t+1)} \leftarrow \text{Project}_{\Delta^n} \left( \mu Lz^{(t)} \right)$

- $Lz$ is gradient of $\frac{1}{2} z^\top Lz$
Projected power method: $z^{(t+1)} \leftarrow \text{Project}_{\Delta n} \left( \mu Lz^{(t)} \right)$

- $Lz$ is gradient of $\frac{1}{2} z^\top Lz$
Projected power method: $z^{(t+1)} \leftarrow \text{Project}_{\Delta n} \left( \mu L z^{(t)} \right)$

- $Lz$ is gradient of $\frac{1}{2} z^\top L z$
Projected power method: $z^{(t+1)} \leftarrow \text{Project}_{\Delta n} (\mu Lz^{(t)})$

- $Lz$ is gradient of $\frac{1}{2} z^T Lz$
Initialization?

\[ L = \underbrace{\mathbb{E}[L]}_{\text{approx. low-rank}} + L - \mathbb{E}[L] \]
Initialization?

\[ L \approx \hat{L} + (L - \mathbb{E}[L]) \]

**Spectral initialization**

1. \( \hat{L} \leftarrow \text{rank-}m \text{ approximation of } L \)
Initialization?

\[
\begin{align*}
L & = \mathbb{E}[L] + L - \mathbb{E}[L] \\
\hat{L} & \leftarrow \text{rank-}m \text{ approximation of } L \\
\end{align*}
\]

Spectral initialization

1. \( \hat{L} \leftarrow \text{rank-}m \text{ approximation of } L \)
2. \( z^{(0)} \leftarrow \text{Project}_{\Delta^n}(\mu \hat{z}) \), where \( \hat{z} \) is a random column of \( \hat{L} \)
Summary of projected power method (PPM)

1. Spectral initialization

2. For $t = 1, 2, \cdots$

$$z^{(t)} \leftarrow \text{Project}_{\Delta_n} \left( \mu L z^{(t-1)} \right)$$
Random corruption model

\[ y_{i,j} \overset{\text{ind}}{=} \begin{cases} \ x_i - x_j \mod m & \text{with prob. } \pi_0 \\ \text{Uniform}(m) & \text{else} \end{cases} \]
Random corruption model

\begin{equation}
    y_{i,j} \overset{\text{ind}}{=} \begin{cases} 
        x_i - x_j \mod m & \text{with prob. } \pi_0 \\
        \text{Uniform}(m) & \text{else}
    \end{cases}
\end{equation}

**Theorem (Chen-Candès’16)** Fix $m > 0$ and set $\mu \gtrsim 1/\sigma_2(L)$. With high prob., PPM recovers the truth exactly within $O(\log n)$ iterations if

- signal-to-noise ratio (SNR) not too small: $\pi_0 > 2\sqrt{\frac{\log n}{mn}}$
Implications

Theorem (Chen-Candès’16) ⋯ PPM succeeds within $O(\log n)$ iterations if the non-corruption rate $\pi_0 > 2\sqrt{\frac{\log n}{mn}}$

- PPM succeeds even when most (i.e. $1 - O(\sqrt{\frac{\log n}{n}})$) entries are corrupted
Implications

**Theorem (Chen-Candès’16)** ••• PPM succeeds within $O(\log n)$ iterations if

non-corruption rate $\pi_0 > 2\sqrt{\frac{\log n}{mn}}$

- PPM succeeds even when most (i.e. $1 - O\left(\sqrt{\frac{\log n}{n}}\right)$) entries are corrupted
- Nearly linear time algorithm
Implications

**Theorem (Chen-Candès’16)** ··· PPM succeeds within $O(\log n)$ iterations if

$$\text{non-corruption rate } \pi_0 > 2\sqrt{\frac{\log n}{mn}}$$

- PPM succeeds even when most (i.e. $1 - O(\sqrt{\frac{\log n}{n}})$) entries are corrupted
- Nearly linear time algorithm
- Works for any initialization obeying $\|z^{(0)} - x\| < 0.5\|x\|$
Empirical misclassification rate

Misclassification rate when $n$ and $\pi_0$ vary \( (\mu = 10/\sigma_2(L)) \)
More general noise models

\[ y_{i,j} = x_i - x_j + \eta_{i,j} \mod m, \quad \text{where } \eta_{i,j} \sim \text{i.i.d. } P_0 \]
More general noise models

\[ y_{i,j} = x_i - x_j + \eta_{i,j} \mod m, \quad \text{where } \eta_{i,j} \overset{\text{i.i.d.}}{\sim} P_0 \]

Distributions of \( y_{i,j} \) under different hypotheses

\[ P_0 \]

\[ x_i - x_j = 0 \]
More general noise models

\[ y_{i,j} = x_i - x_j + \eta_{i,j} \mod m, \quad \text{where } \eta_{i,j} \sim \text{i.i.d.} P_0 \]

Distributions of \( y_{i,j} \) under different hypotheses

\[ P_0 \quad \begin{array}{c} \text{Distribution of } P_0 \\ \end{array} \]
\[ x_i - x_j = 0 \]

\[ P_1 \quad \begin{array}{c} \text{Distribution of } P_1 \\ \end{array} \]
\[ x_i - x_j = 1 \]
More general noise models

\[ y_{i,j} = x_i - x_j + \eta_{i,j} \mod m, \quad \text{where } \eta_{i,j} \overset{\text{i.i.d.}}{\sim} P_0 \]

Distributions of \( y_{i,j} \) under different hypotheses

\[ P_0 \]
\[ x_i - x_j = 0 \]

\[ P_1 \]
\[ x_i - x_j = 1 \]

\[ P_9 \]
\[ x_i - x_j = 9 \]
More general noise models

\[ y_{i,j} = x_i - x_j + \eta_{i,j} \mod m, \]
where \( \eta_{i,j} \text{i.i.d.} \sim P_0 \)

Distributions of \( y_{i,j} \) under different hypotheses

\[ P_0 \]
\[ x_i - x_j = 0 \]

\[ P_1 \]
\[ x_i - x_j = 1 \]
\[ \downarrow \]
\[ \text{KL}(P_0 \parallel P_1) \]

\[ P_9 \]
\[ x_i - x_j = 9 \]
\[ \downarrow \]
\[ \text{KL}(P_0 \parallel P_9) \]
More general noise models

\[ y_{i,j} = x_i - x_j + \eta_{i,j} \mod m, \quad \text{where } \eta_{i,j} \overset{\text{i.i.d.}}{\sim} P_0 \]

Distributions of \( y_{i,j} \) under different hypotheses

\[
\begin{align*}
P_0 & \quad x_i - x_j = 0 \\
\Downarrow & \\
KL(P_0 \parallel P_1) & \\
\Downarrow & \\
P_9 & \quad x_i - x_j = 9
\end{align*}
\]

**Theorem (Chen-Candès’16)** Fix \( m > 0 \) and set \( \mu \gtrsim 1/\sigma_2(L) \). Under mild conditions, PPM succeeds within \( O(\log n) \) iterations with high prob., provided that

\[
KL_{\min} := \min_{1 \leq l < m} KL(P_0 \parallel P_l) > \frac{4 \log n}{n}
\]
Interpretation: why $\text{KL}_{\text{min}}$ matters

**Theorem (Chen-Candès’16)** ... PPM succeeds within $O(\log n)$ iterations if

\[
\text{KL}_{\text{min}} := \min_{1 \leq l < m} \text{KL}(P_0 \parallel P_l) > \frac{4 \log n}{n}
\]

Suppose $x_1 = \cdots = x_n = 1$
Interpretation: why $\text{KL}_{\text{min}}$ matters

**Theorem (Chen-Candès'16)** ... PPM succeeds within $O(\log n)$ iterations if

$$\text{KL}_{\text{min}} := \min_{1 \leq l < m} \text{KL}(P_0 \parallel P_l) > \frac{4 \log n}{n}$$

Suppose $x_1 = \cdots = x_n = 1$
Interpretation: why $\text{KL}_{\min}$ matters

**Theorem (Chen-Candès’16)** ... PPM succeeds within $O(\log n)$ iterations if

$$\text{KL}_{\min} := \min_{1 \leq l < m} \text{KL}(P_0 \parallel P_l) > \frac{4 \log n}{n}$$

Suppose $x_1 = \cdots = x_n = 1$

- Peaks of $\mathbb{E}[L]$ reveal ground truth $\mathbb{E}[L_{i,j}]$
Interpretation: why $KL_{\min}$ matters

**Theorem (Chen-Candès’16)** ... PPM succeeds within $O(\log n)$ iterations if

$$KL_{\min} := \min_{1 \leq l < m} KL(P_0 \parallel P_l) > \frac{4 \log n}{n}$$

Suppose $x_1 = \cdots = x_n = 1$

- Peaks of $E[L]$ reveal ground truth $E[L_{i,j}]$
- $L \approx E[L]$ if $KL_{\min}$ is sufficiently large
Empirical misclassification rate

Modified Gaussian noise model:

\[ P \{ \eta_{i,j} = z \} \propto \exp \left( -\frac{z^2}{2\sigma^2} \right), \quad |z| \leq \frac{m-1}{2} \]
PPM is information-theoretically optimal

Theorem (Chen-Candes'16)

Fix $m > 0$. No method achieves exact recovery if $KL_{\min} < 4 \log n$. 

PPM works
PPM is information-theoretically optimal

Theorem (Chen-Candès’16) Fix $m > 0$. No method achieves exact recovery if

$$K_{L_{\text{min}}} < \frac{4 \log n}{n}$$
Large-$m$ case: random corruption model

$$y_{i,j} = \begin{cases} 
  x_i - x_j, & \text{with prob. } \pi_0 \\
  \text{Unif}(m), & \text{else}
\end{cases}$$

**Theorem (Chen-Candès’16)** Suppose $\log n \lesssim m \lesssim \text{poly}(n)$. PPM succeeds if

$$\pi_0 \gtrsim \frac{1}{\sqrt{n}}$$
Large-$m$ case: random corruption model

\[ y_{i,j} = \begin{cases} x_i - x_j, & \text{with prob. } \pi_0 \\ \text{Unif}(m), & \text{else} \end{cases} \]

Theorem (Chen-Candès’16) Suppose \( \log n \lesssim m \lesssim \text{poly}(n) \). PPM succeeds if

\[ \pi_0 \gtrsim \frac{1}{\sqrt{n}} \]

- **Spiky model:** when \( m \gg n \), model converges to

\[ x_i \in [0, 1), \quad y_{i,j} = \begin{cases} x_i - x_j, & \text{with prob. } \pi_0 \\ \text{Unif}(0, 1), & \text{else} \end{cases} \]

Singer'09; Wang & Singer'12; Bandeira et al’14; Boumal’16; Liu et al’16, Perry et al’16 ...
Large-\(m\) case: random corruption model

\[
y_{i,j} = \begin{cases} 
  x_i - x_j, & \text{with prob. } \pi_0 \\
  \text{Unif}(m), & \text{else}
\end{cases}
\]

**Theorem (Chen-Candès’16)** Suppose \(\log n \preceq m \preceq \text{poly}(n)\). PPM succeeds if

\[
\pi_0 \gtrsim \frac{1}{\sqrt{n}}
\]

- **Spiky model**: when \(m \gg n\), model converges to

\[
x_i \in [0, 1), \quad y_{i,j} = \begin{cases} 
  x_i - x_j, & \text{with prob. } \pi_0 \\
  \text{Unif}(0, 1), & \text{else}
\end{cases}
\]

Succeeds even if a dominant fraction \(1 - O(1/\sqrt{n})\) of inputs are corrupted

---

*Singer’09; Wang & Singer’12; Bandeira et al’14; Boumal’16; Liu et al’16, Perry et al’16 ...*
Large-$m$ case: random corruption model

\[
y_{i,j} = \begin{cases} 
  x_i - x_j, & \text{with prob. } \pi_0 \\
  \text{Unif}(m), & \text{else}
\end{cases}
\]

**Theorem (Chen-Candès’16)** Suppose \( \log n \lesssim m \lesssim \text{poly}(n) \). PPM succeeds if

\[
\pi_0 \gtrsim \frac{1}{\sqrt{n}}
\]

- “Smooth” noise model if \( m \lesssim \sqrt{n} \)

---

*Singer’09; Wang & Singer’12; Bandeira et al’14; Boumal’16; Liu et al’16, Perry et al’16 ...*
Large-$m$ case: random corruption model

\[ y_{i,j} = \begin{cases} 
  x_i - x_j, & \text{with prob. } \pi_0 \\
  \text{Unif}(m), & \text{else}
\end{cases} \]

Theorem (Chen-Candès’16) Suppose $\log n \lesssim m \lesssim \text{poly}(n)$. PPM succeeds if

\[ \pi_0 \gtrsim \frac{1}{\sqrt{n}} \]

- "Smooth" noise model if $m \lesssim \sqrt{n}$
  - Recovers each $x_i \in [0,1)$ up to a resolution of $\frac{1}{m} \gtrsim \frac{1}{\sqrt{n}}$

Singer’09; Wang & Singer’12; Bandeira et al’14; Boumal’16; Liu et al’16, Perry et al’16 ...
Joint shape alignment: Chair dataset from ShapeNet

20 representative shapes (out of 50)

1We add extra noise to each point of the shapes to make it more challenging.
Joint shape alignment: Chair dataset from ShapeNet\textsuperscript{1}

20 representative shapes (out of 50)

pairwise cost $-\ell_{i,j}(x_i, x_j)$: 
avg nearest-neighbor squared distance

\textsuperscript{1}We add extra noise to each point of the shapes to make it more challenging.
Joint shape alignment: Chair dataset from ShapeNet

20 representative shapes (out of 50)

pairwise cost $-\ell_{i,j}(x_i, x_j)$:
avg nearest-neighbor squared distance

aligned shapes

---

1We add extra noise to each point of the shapes to make it more challenging.
Joint shape alignment: angular estimation errors

We add extra noise to each point of the shapes to make it more challenging.

<table>
<thead>
<tr>
<th></th>
<th>projected power method</th>
<th>semidefinite relaxation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Runtime</td>
<td>2.4 sec</td>
<td>895.6 sec</td>
</tr>
</tbody>
</table>

\(^2\)We add extra noise to each point of the shapes to make it more challenging.
Joint graph matching: CMU House dataset

111 images of a toy house
Joint graph matching: CMU House dataset

111 images of a toy house

input matches

3 representative images
Joint graph matching: CMU House dataset

111 images of a toy house

3 representative images
Dixon imaging in body MRI


2 phasor candidates for field inhomogeneity at each voxel

candidate 1

candidate 2
Dixon imaging in body MRI


2 phasor candidates for field inhomogeneity at each voxel

\[
\begin{align*}
\text{maximize} & \quad \sum \ell(x_i, x_j) \\
\text{subject to} & \quad x_i \in \{1, 2\}
\end{align*}
\]
Dixon imaging in body MRI

Representative cases of water signal recovery


commercial software

projected power method
Things I have not talked about ...

1. General noise model with large $m$

2. Incomplete data
Concluding remarks

A new approach to discrete assignment problems

- Finds MLE in suitable regimes
- Computationally efficient

**Paper:** “The projected power method: an efficient algorithm for joint alignment from pairwise differences”, Y. Chen and E. Candès, 2016