Low-rank Tensor Recovery: Theory and Algorithms

Donald Goldfarb

John Wright

Bo Huang

Cun Mu

Zhiwei Qin

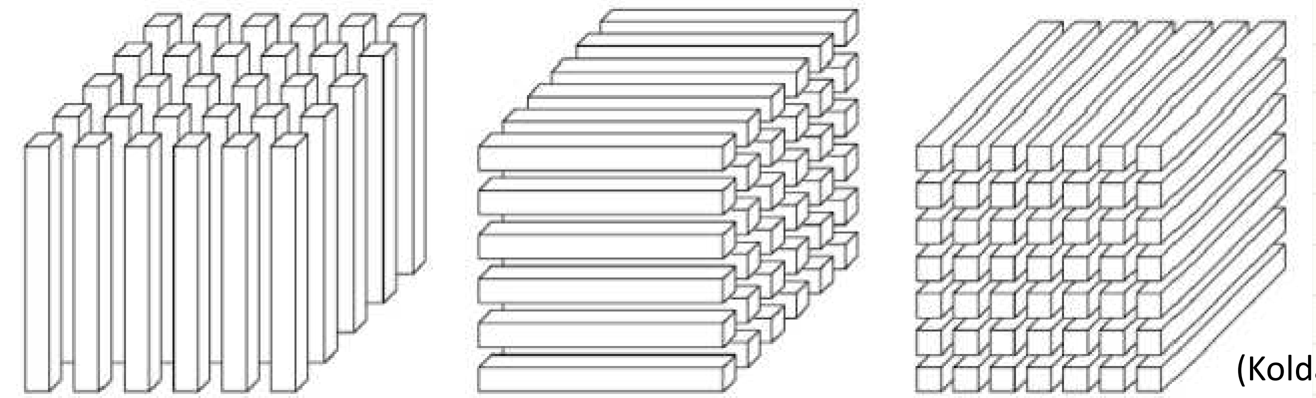
Outline

- Tensor Basics, Decomposition and Rank
- Low-rank Tensor Recovery Models
  - *Tensor Completion*
  - *Tensor Robust Principal Component Analysis*
- Algorithms
  - *Alternating Direction Augmented Lagrangian(ADAL)*
  - *Accelerated Linearized Bregman (ALB)*
- Experiments
- Alternative convex model: Square Deal

Tensor Decomposition and Rank

Tensor Basics

- Modes and Fibers



(a) Mode-1 (column) fibers: $\mathbf{x}_{:jk}$ (b) Mode-2 (row) fibers: $\mathbf{x}_{i:k}$ (c) Mode-3 (tube) fibers: $\mathbf{x}_{ij:}$

(Kolda & Bader, 2009)

- Unfolding (flattening)

$$\mathcal{X} = \begin{bmatrix} 1 & 4 & 7 & 10 \\ 2 & 5 & 8 & 11 \\ 3 & 6 & 9 & 12 \end{bmatrix} \begin{bmatrix} 13 & 16 & 19 & 22 \\ 14 & 17 & 20 & 23 \\ 15 & 18 & 21 & 24 \end{bmatrix}$$

$$\mathbf{X}_{(1)} = \begin{bmatrix} 1 & 4 & 7 & 10 & 13 & 16 & 19 & 22 \\ 2 & 5 & 8 & 11 & 14 & 17 & 20 & 23 \\ 3 & 6 & 9 & 12 & 15 & 18 & 21 & 24 \end{bmatrix}$$

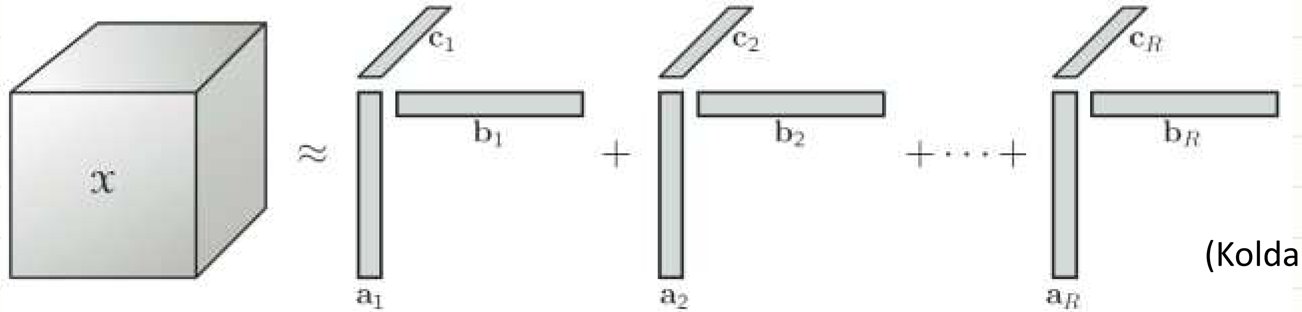
$$\mathbf{X}_{(2)} = \begin{bmatrix} 1 & 2 & 3 & 13 & 14 & 15 \\ 4 & 5 & 6 & 16 & 17 & 18 \\ 7 & 8 & 9 & 19 & 20 & 21 \\ 10 & 11 & 12 & 22 & 23 & 24 \end{bmatrix}$$

$$\mathbf{X}_{(3)} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & \cdots & 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 & 17 & \cdots & 21 & 22 & 23 & 24 \end{bmatrix}$$

Tensor Decomposition and Rank

- CANDECOMP/PARAFAC (CP)

$$\mathcal{X} \approx \sum_{r=1}^{R} \lambda_r \boldsymbol{a}_r^{(1)} \circ \boldsymbol{a}_r^{(2)} \circ \cdots \circ \boldsymbol{a}_r^{(N)}$$



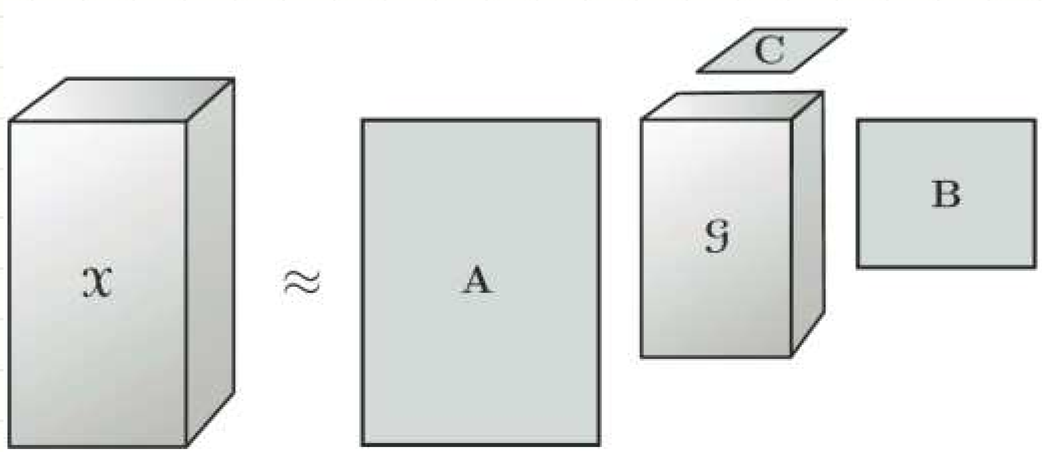
(Kolda & Bader, 2009)

- $\left(\boldsymbol{a}^{(1)} \circ \boldsymbol{a}^{(2)} \circ \cdots \circ \boldsymbol{a}^{(N)}\right)_{i_1 i_2 \cdots i_N} = a_{i_1}^{(1)} a_{i_2}^{(2)} \cdots a_{i_N}^{(N)}$
- rank$(\mathcal{X}) \stackrel{\text{def}}{=}$ smallest R s.t. approximation holds with equality

Tensor Decomposition and Rank

- Tucker Decomposition (Higher-order SVD)

$$\mathcal{X} \approx \mathcal{G} \times_1 A \times_2 B \times_3 C$$



(Kolda & Bader, 2009)

- $\mathcal{Y} = \mathcal{X} \times_n A \Leftrightarrow Y_{(n)} = AX_{(n)}$
- *Tucker (multilinear) Rank*: set of column ranks of the N unfoldings $X_{(n)}, n = 1, \cdots, N$.

Notation:

- True tensor: $\mathcal{X}_0 \in \mathbb{R}^{n_1 \times n_2 \cdots n_K}$

- The ith unfolding: $\mathcal{X}_{(i)} \in \mathcal{R}^{n_i \times \Pi_{j=1,j\neq i}^{k} n_j}$

$$n_{(1)}^{i} := \max(n_i, \Pi_{j=1,j\neq i}^{k} n_j) \quad n_{(2)}^{i} := \min(n_i, \Pi_{j=1,j\neq i}^{k} n_j)$$

- Let r_i be the rank of the ith unfolding $\mathcal{X}_{(i)}$

- Let $U_i \in \mathbb{R}^{n_i \times r_i}$ and $V_i \in \mathbb{R}^{(\Pi_{j=1,j\neq i}^{k} n_j) \times r_i}$ are left and right singular vectors of $\mathcal{X}_{(i)}$

Tensor Completion

Tensor Completion

- Non-convex Model: (Vector optimization)

$$\min_{\text{w.r.t. } \mathbb{R}_+^K} \operatorname{rank}_{\text{tc}}(\boldsymbol{\mathcal{X}}) \quad \mathcal{P}_\Omega[\boldsymbol{\mathcal{X}}] = \mathcal{P}_\Omega[\boldsymbol{\mathcal{X}}_0]$$

- Convex Model:

$$\min_{\boldsymbol{\mathcal{X}}} \sum_{i=1}^{K} \lambda_i \|\boldsymbol{\mathcal{X}}_{(i)}\|_* \quad \mathcal{P}_\Omega[\boldsymbol{\mathcal{X}}] = \mathcal{P}_\Omega[\boldsymbol{\mathcal{X}}_0]$$

Tensor Completion

- Strongly Convex Model (S-TC):

$$\min_{\boldsymbol{\mathcal{X}}} \sum_{i=1}^{K} \lambda_i \|\boldsymbol{\mathcal{X}}_{(i)}\|_* + K\tau \|X\|_F^2 \quad \mathcal{P}_\Omega[\boldsymbol{\mathcal{X}}] = \mathcal{P}_\Omega[\boldsymbol{\mathcal{X}}_0]$$

  – Non-smooth convex model if $\tau = 0$

  – Strong convexity allows more efficient algorithms

  – Exact recovery guaranteed when τ is below threshhold

<u>Tensor Completion: recovery guarantees</u>

- Is exact recovery always possible?
- No! Conditions need to be posed on the tensor structure.
- *Tensor Incoherence Conditions (TICs)*

Tensor Completion: TICs

- ***Tensor Incoherence Conditions (TICs)*** with respect to $\mu_{i,0}$ and $\mu_{i,1}$: (same as matrix case)
- For all i=1,…,K

$$\max_j \|U_i^\top e_j\|^2 \le \frac{\mu_{i,0} r_i}{n_i}, \quad \max_j \|V_i^\top e_j\|^2 \le \frac{\mu_{i,0} r_i}{\prod_{j=1, j\neq i}^K n_j},$$

$$\|U_i V_i^\top\|_\infty \le \sqrt{\frac{\mu_{i,1} r_i}{\prod_{j=1}^K n_j}},$$

Exact recovery requires:

1. Incoherent tensor: Small $\mu_{i,0}$ and $\mu_{i,1}$
2. Low rank: small r_i

Tensor Completion: Main Theorem

Theorem: Suppose $\mathcal{X}_0$ obeys the TICs,

$\mu_i := \max\{\mu_{i,0}, \mu_{i,1}\}$, $|\Omega| = m$, and

$$m \geq 32\beta \max_i \{\mu_i r_i (n_{(1)}^i + n_{(2)}^i) \log^2(8 n_{(1)}^i)\}$$

then (S-TC) is exact for

(i) any choice of $\{\lambda_i\}$, when $\tau = 0$;

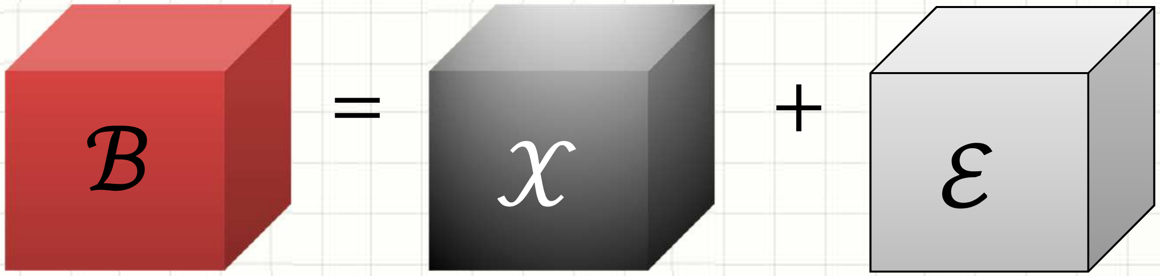
(ii) $\lambda_i \geq \frac{32}{3\rho}\beta^{1/2} \log(n_{(1)}^i) \|\mathcal{P}_\Omega \mathcal{X}_0\|_F \sqrt{2 n_{(1)}^i}$, $\tau = 1$

with probability

$$1 - 6K \max_i \{\log(n_{(1)}^i)(n_{(1)}^i + n_{(2)}^i)^{2-2\beta} - (n_{(1)}^i)^{2-2\beta^{1/2}}\}$$

Tensor Robust Principal Component Analysis (TRPCA)

TRPCA: Tensor Robust Principal Component Analysis



	Noisy	Low-rank	Sparse
NP-hard:		**Tucker-rank**$(\mathcal{X})$	$\|\|\mathcal{E}\|\|_0$
Convex:		$\sum_{i=1}^{K} \lambda_i \|\|\boldsymbol{\mathcal{X}}_{(i)}\|\|_*$	$\|\|\mathcal{E}\|\|_1$

TRPCA: Strongly Convex Formulation

- Similar to tensor completion, we extend TRPCA to allow a strongly convex model:

$$\min_{\boldsymbol{\mathcal{X}},\boldsymbol{\mathcal{E}}} \quad \sum_{i}^{K} \lambda_i ||\boldsymbol{\mathcal{X}}_{(i)}||_* + K||\boldsymbol{\mathcal{E}}||_1 + \frac{\tau}{2}||\boldsymbol{\mathcal{X}}||_F^2 + \frac{\tau}{2}||\boldsymbol{\mathcal{E}}||_F^2$$
$$s.t. \quad \boldsymbol{\mathcal{X}} + \boldsymbol{\mathcal{E}} = \boldsymbol{\mathcal{B}},$$

- Only convex TRPCA if $\tau = 0$

TRPCA: Main Theorem

Theorem: Suppose $\mathcal{X}_0$ obeys the TICs, $\mu_i := \max\{\mu_{i,0}, \mu_{i,1}\}$ and $\lambda_i = \sqrt{n_{(1)}^i}$, then TRPCA is exact when

$$\tau \leq \min\{\frac{K}{10||\mathcal{B}||_\infty}, \frac{K}{5||\mathcal{B}||_F}\}$$

and

$$r_i \leq \rho_{r_i} n_{(2)}^i \mu_i^{-1} (\log n_{(1)}^i)^{-2}, \quad m \leq \rho_s \prod_{j=1}^{K} n_j$$

with probability at least $1 - cKn^{-10}$, where ρ_{r_i} and ρ_s are positive constants.

TRPCA: Grossly corrupted data

- Suppose now that some of the data entrees are missing and the others are corrupted;

- Let Ω be the set of locations where there is data;

- Each entry in Ω is corrupted with probability γ independently of the others;

- Is exact recovery possible in this case? Yes!

TRPCA: Grossly corrupted data

- TRPCA with grossly corrupted data can be formulated as (strongly convex model):

$$\min_{\boldsymbol{\mathcal{X}},\boldsymbol{\mathcal{E}}} \quad \sum_{i=1}^{k} \lambda_i ||\boldsymbol{\mathcal{X}}_{(i)}||_* + K||\boldsymbol{\mathcal{E}}||_1 + \frac{\tau}{2}||\boldsymbol{\mathcal{X}}||_F^2 + \frac{\tau}{2}||\boldsymbol{\mathcal{E}}||_F^2$$

$$s.t. \qquad \mathcal{P}_\Omega \boldsymbol{\mathcal{X}} + \boldsymbol{\mathcal{E}} = \boldsymbol{\mathcal{B}}.$$

$$(\boldsymbol{\mathcal{E}} \in \Omega, \quad \boldsymbol{\mathcal{B}} \in \Omega)$$

TRPCA: Grossly corrupted data

Theorem: Suppose $\mathcal{X}_0$ obeys the TICs, $\mu_i := \max\{\mu_{i,0}, \mu_{i,1}\}$ and let $\lambda_i = \sqrt{\rho n^i_{(1)}}$; then TRPCA with grossly corrupted data can be solved exactly with probability at least

$$1 - cKn^{-10}$$

provided that

$$\tau = \min_i \{ \frac{K\lambda_i (n^i_{(1)} n^i_{(2)})^{-1}}{(1 + \frac{4}{\rho(1-\gamma_s)}) ||\mathcal{P}_\Omega \mathcal{B}||_F} \},$$

$$r_i \leq \rho_{r_i} n^i_{(2)} (\log n^i_1)^{-2}, \ \ \forall i \ \ and \ \ \gamma \leq \gamma_s,$$

where ρ_{r_i} and γ_s are positive constants.

OPTIMIZATION ALGORITHMS

Algorithms: Variable splitting

$$\sum_i \lambda_i \left\| \boldsymbol{\mathcal{X}}_{(\boldsymbol{i})} \right\|_*$$

$$\mathcal{X}$$

$$\mathcal{X}_1 = \mathcal{X}_2 = \cdots \mathcal{X}_N$$

$$\sum_i \lambda_i \left\| \boldsymbol{\mathcal{X}}_{i,(i)} \right\|_*$$

$$s.t. \quad \mathcal{X}_i = \mathcal{W} \quad i = 1, \cdots, N$$

Algorithms: Variable splitting

- **Tensor Completion:**

$$
\begin{aligned}
\min_{\boldsymbol{\mathcal{X}}_i, \boldsymbol{\mathcal{W}}} \quad & \sum_i \left[\lambda \|\boldsymbol{\mathcal{X}}_{i,(i)}\|_* + \tau \|\boldsymbol{\mathcal{X}}\|_F^2\right] + \tau \|\boldsymbol{\mathcal{W}}\|_F^2 \\
\text{s.t.} \quad & \boldsymbol{\mathcal{X}}_i = \boldsymbol{\mathcal{W}}, \;\; i = 1, 2, \cdots K, \\
& \mathcal{P}_\Omega W = \mathcal{P}_\Omega \boldsymbol{\mathcal{X}}_0 .
\end{aligned}
$$

Algorithms: Variable splitting

- **TRPCA with all data observed:**

$$
\begin{aligned}
\min_{\mathcal{X}_i,\, \mathcal{W}} \quad & \sum_i \left[\lambda \|\mathcal{X}_{i,(i)}\|_* + \frac{\tau}{K+1} \|\mathcal{X}\|_F^2 \right] \\
& + \sum_i \|\mathcal{E}\|_1 + \tau \|\mathcal{E}\|_F^2 \\
\text{s.t.} \quad & \mathcal{X}_i + \mathcal{E} = \mathcal{B}, \;\; i = 1, 2, \cdots K.
\end{aligned}
$$

Algorithms: Variable splitting

- **TRPCA with grossly corrupted data**

$$
\begin{aligned}
\min_{\boldsymbol{\mathcal{X}}_i,\, \boldsymbol{\mathcal{W}}} \quad & \sum_i \left[\lambda \|\boldsymbol{\mathcal{X}}_{i,(i)}\|_* + \frac{\tau}{K+1} \|\boldsymbol{\mathcal{X}}\|_F^2 \right] \\
& + \sum_i \|\boldsymbol{\mathcal{E}}\|_1 + \tau \|\boldsymbol{\mathcal{E}}\|_F^2 + \frac{\tau}{K+1} \|\boldsymbol{\mathcal{W}}\|_F^2 \\
\text{s.t.} \quad & \boldsymbol{\mathcal{X}}_i = \boldsymbol{\mathcal{W}}, \;\; i = 1, 2, \cdots K, \\
& \mathcal{P}_\Omega \boldsymbol{\mathcal{W}} + \boldsymbol{\mathcal{E}} = \boldsymbol{\mathcal{B}}.
\end{aligned}
$$

Algorithms: Alternating Direction Augmented Lagrangian (ADAL)

- Consider augmented Lagrangian function for non-strictly convex TRPCA problem ($\tau = 0$):

$$
\begin{aligned}
L(\mathcal{X}, \mathcal{E}) \quad := \quad & \sum_{i=1}^{K} \lambda_i \|\mathcal{X}_{i,(i)}\|_* + K\|\mathcal{E}\|_1 \\
& + \sum_{i=1}^{K} \left(\frac{1}{2\mu} \|\mathcal{X}_i + \mathcal{E} - \mathcal{B}\|_F^2 - \langle \Lambda_i, \mathcal{X}_i + \mathcal{E} - \mathcal{B} \rangle \right)
\end{aligned}
$$

- Hard to solve $\mathcal{X}$ and $\mathcal{E}$ simultaneously!

Algorithms: ADAL

- ADAL framework:

$$
\begin{aligned}
\mathcal{X}_{(i)}^{k+1} &:= \min_{\mathcal{X}} L(\mathcal{X}, \mathcal{E}^k) \\
\mathcal{E}^{k+1} &:= \min_{\mathcal{E}} L(\mathcal{X}^{k+1}, \mathcal{E}) \\
\Lambda_i^{k+1} &:= \Lambda_i^k - \frac{1}{\mu}(\mathcal{X}_i^{k+1} + \mathcal{E}^{k+1} - \mathcal{B})
\end{aligned}
$$

- Each sub-problem is a simple shrinkage operation!

Algorithms: ADAL

Step 1: X-subproblem:

$$\begin{aligned}\boldsymbol{\mathcal{X}}_{(i)}^{k+1} &:= \min_{\boldsymbol{\mathcal{X}}} L(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{E}}^k) \\ &= \boldsymbol{\mathcal{T}}_{\lambda_i \mu}^{m}\left(\boldsymbol{\mathcal{E}}_{(i)}^{k} - \boldsymbol{\mathcal{B}}_{(i)}^{k} - \mu \Lambda_{i,(i)}^{k}\right)\end{aligned}$$

– Singular-value Soft-threshold:

$$\boldsymbol{\mathcal{T}}_{\mu}^{m}(X) := U \operatorname{diag}(\bar{\sigma}) V^{\top}, \qquad X = U \operatorname{diag}(\sigma) V^{\top}$$

$$\bar{\sigma} := \max(\sigma - \mu, 0)$$

Algorithms: ADAL

Step 2: E-subproblem:

$$\begin{aligned} \mathcal{E}^{k+1} &:= \min_{\mathcal{E}} L(\mathcal{X}^{k+1}, \mathcal{E}) \\ &= \min_{\mathcal{E}} \frac{1}{2} \|C(\mathcal{E}) + \mathcal{D}\|_F^2 + \mu K \|\mathcal{E}\|_1 \end{aligned}$$

$$C(\mathcal{E}) := \begin{pmatrix} \mathcal{E} \\ \vdots \\ \mathcal{E} \end{pmatrix}, \qquad \mathcal{D} := \begin{pmatrix} \mathcal{X}_1^{k+1} - \mathcal{B} - \mu \Lambda_1^k \\ \vdots \\ \mathcal{X}_K^{k+1} - \mathcal{B} - \mu \Lambda_K^k \end{pmatrix}$$

– Lasso shrinkage operation, simple!

Algorithms: Linearized Bregman Method

- Solve the linearly constraint problems:

$$\min_x J(x) \quad \text{s.t.} \quad Ax = b$$

- Bregman distance:

$$D_J^p(u,v) := J(u) - J(v) - \langle p, u - v \rangle,$$

$$p \in \partial J(v)$$

Algorithms: Linearized Bregman Method

- ***Bregman Algorithm:***

Algorithm 1 Original Bregman Iterative Method

Input: $x^0 = p^0 = 0$.
for $k = 0, 1, \cdots$ **do**
  $x^{k+1} = \arg\min_x D_J^{p^k}(x, x^k) + \frac{1}{2}\|Ax - b\|^2$;
  $p^{k+1} = p^k - A^\top(Ax^{k+1} - b)$;
end for

- ***Linearized Bregman Algorithm:***

Algorithm 2 Linearized Bregman Method

Input: $x^0 = p^0 = 0$, $\mu > 0$ and $\tau > 0$.
for $k = 0, 1, \cdots$ **do**
  $x^{k+1} = \arg\min_x D_J^{p^k}(x, x^k) + \tau\langle A^\top(Ax^k - b), x\rangle + \frac{1}{2\mu}\|x - x^k\|^2$
  $p^{k+1} = p^k - \tau A^\top(Ax^k - b) - \frac{1}{\mu}(x^{k+1} - x^k)$;
end for

Linearized Bregman Method: Dual Formulation

- The *Linearized Bregman* method is equivalent to the *Dual Gradient Descent* method on:

$$\min_x J(x) + \frac{1}{2\mu}\|x\|_2^2 \quad \text{s.t.} \quad Ax = b$$

- **Dual Formulation:**

Algorithm 3 Linearized Bregman Method (Equivalent Form)

1: Input: $\mu > 0$, $\tau > 0$ and $y^0 = \tau b$.
2: **for** $k = 0, 1, \cdots$ **do**
3: $\quad w^{k+1} := \arg\min_w \{J(w) + \frac{1}{2\mu}\|w\|^2 - \langle y^k, Aw - b\rangle\}$;
4: $\quad y^{k+1} := y^k - \tau(Aw^{k+1} - b)$.;
5: **end for**

Accelerated Linearized Bregman (ALB)

- Nesterov's accelerating technique:

Algorithm 4 Accelerated Linearized Bregman Method

1: Input: $x^0 = \tilde{x}^0 = \tilde{p}^0 = p^0 = 0$, $\mu > 0$, $\tau > 0$.
2: **for** $k = 0, 1, \cdots$ **do**
3: $\quad x^{k+1} = \arg\min_x D_J^{\tilde{p}^k}(x, \tilde{x}^k) + \tau\langle A^\top(A\tilde{x}^k - b), x\rangle + \frac{1}{2\mu}\|x - \tilde{x}^k\|^2$;
4: $\quad p^{k+1} = \tilde{p}^k - \tau A^\top(A\tilde{x}^k - b) - \frac{1}{\mu}(x^{k+1} - \tilde{x}^k)$;
5: $\quad \tilde{x}^{k+1} = \alpha_k x^{k+1} + (1 - \alpha_k)x^k$;
6: $\quad \tilde{p}^{k+1} = \alpha_k p^{k+1} + (1 - \alpha_k)p^k$.
7: **end for**

- Convergence: Optimal $O(1/k^2)$ rate w.r.t. the Lagrangian function.

ALB method on TRPCA

- At iteration k, we solve the following subproblem:

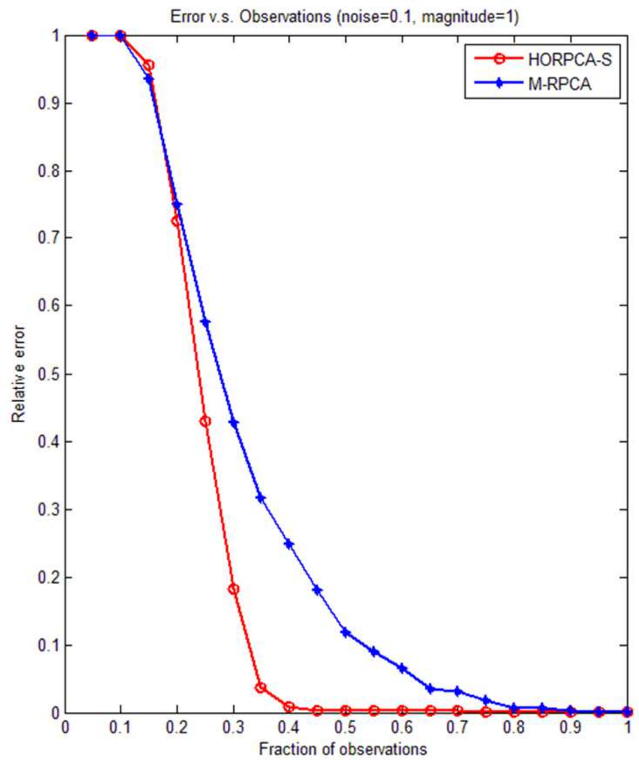
$$(\{\mathcal{X}_i^{k+1}\}, \mathcal{E}^{k+1}) = arg \min_{\{\mathcal{X}_i\},\mathcal{E}} \sum_i \left(\lambda \|\mathcal{X}_{i,(i)}\|_* + \frac{\tau}{K+1} \|\mathcal{X}\|_F^2 - \langle \Lambda_i^k, \mathcal{X}_i \rangle \right)$$
$$+ \sum_i \left(\|\mathcal{E}\|_1 + \tau \|\mathcal{E}\|_F^2 - \langle \Lambda_i^k, \mathcal{E} \rangle \right)$$

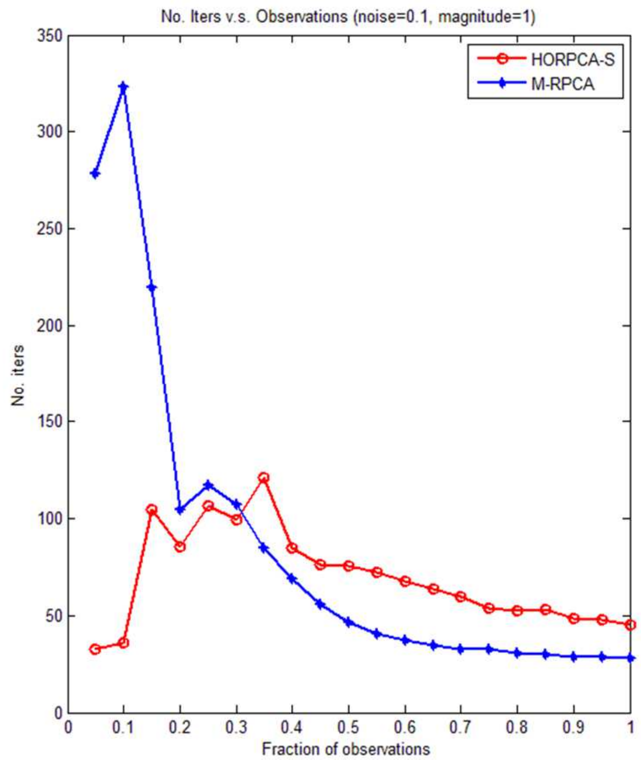
- Decomposable into two subproblems;
- *X-subproblem*: SVD Soft-threshold;
- *E-subproblem*: Lasso shrinkage

EXPERIMENTS

Empirical Recoverability

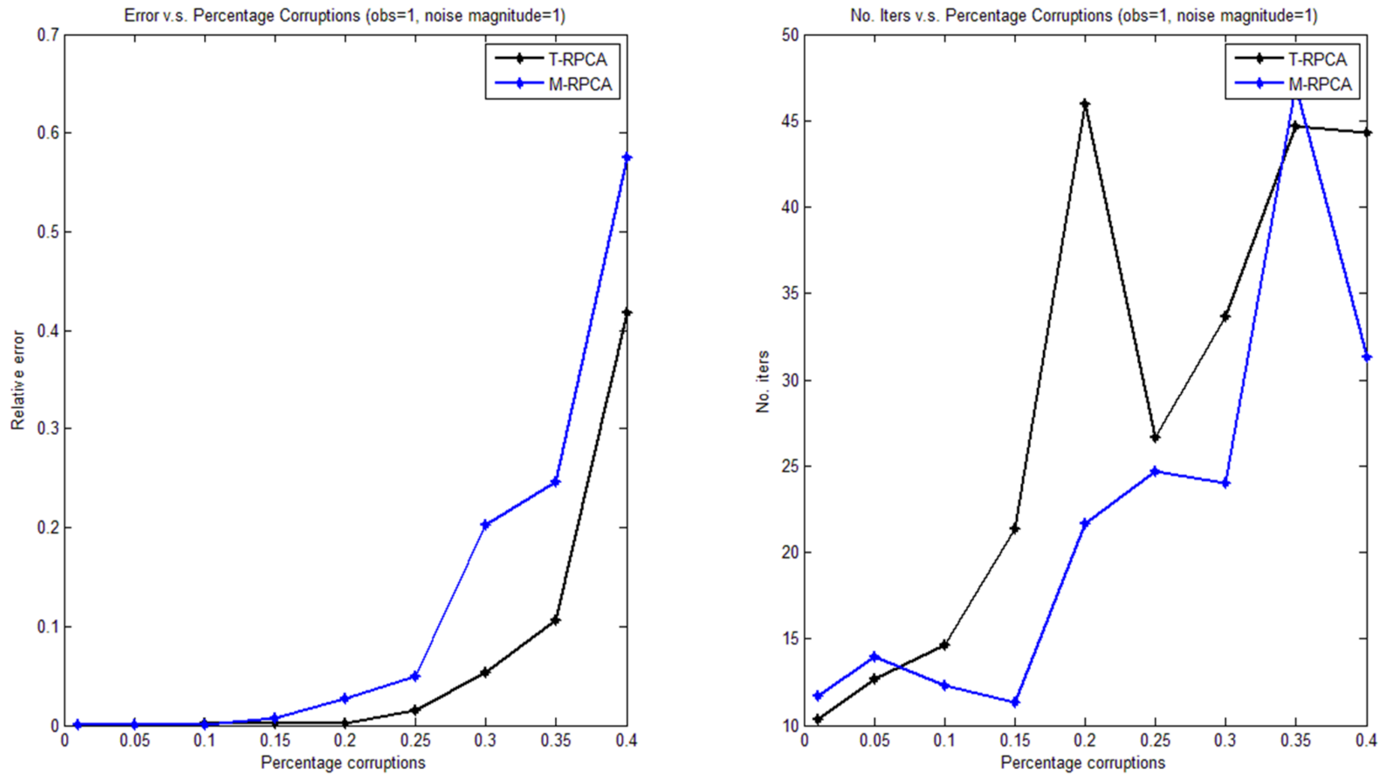
- Random tensors (50, 50, 20), rank-(5,5,5)
- 10% data corrupted, varying % observations…





Empirical Recoverability

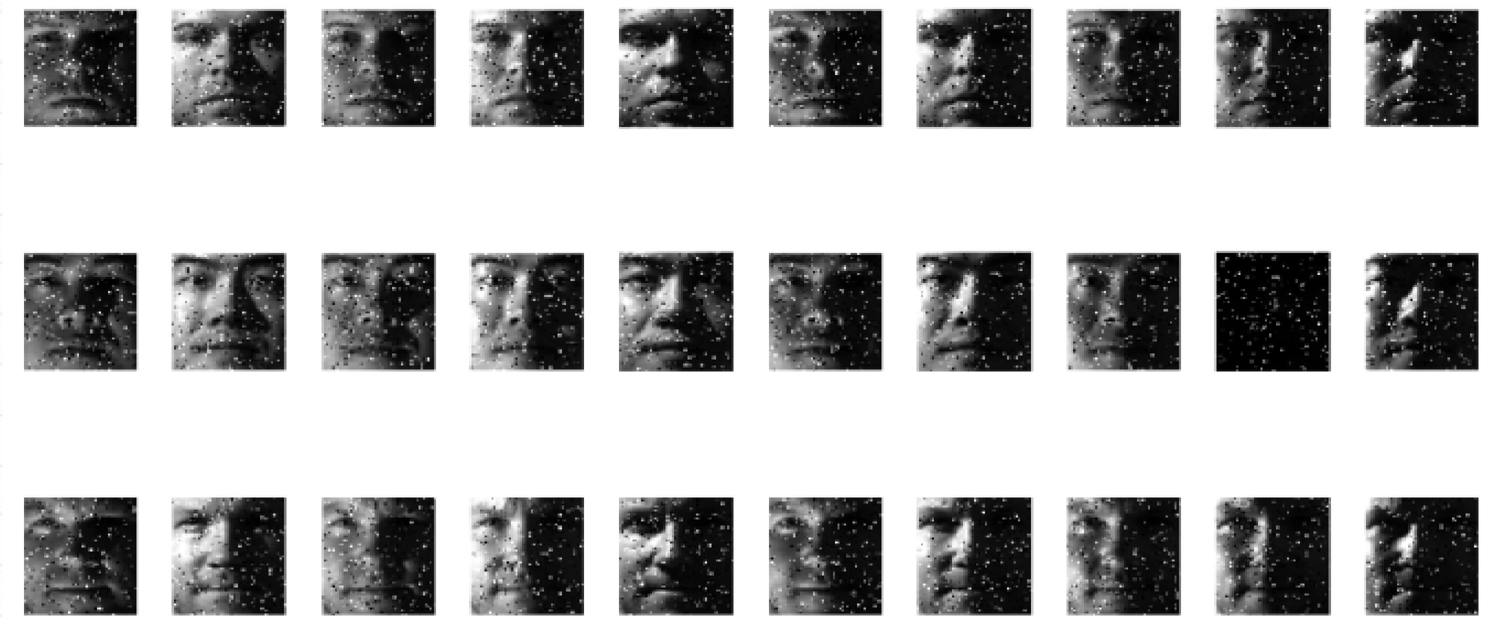
- Random tensors (50, 50, 20), rank-(5,5,5)
- Full observation, varying % corruption…



Face Shadow Reduction

- YaleB face ensemble subset: 5 people, 40 illuminations.
- Each image: 64×56 grey-scale and vectorized.
- Resulting Data: A 3584×40×5 tensor
- 10% pixels corrupted by uniform distributed noise.

Face Shadow Reduction



Original face

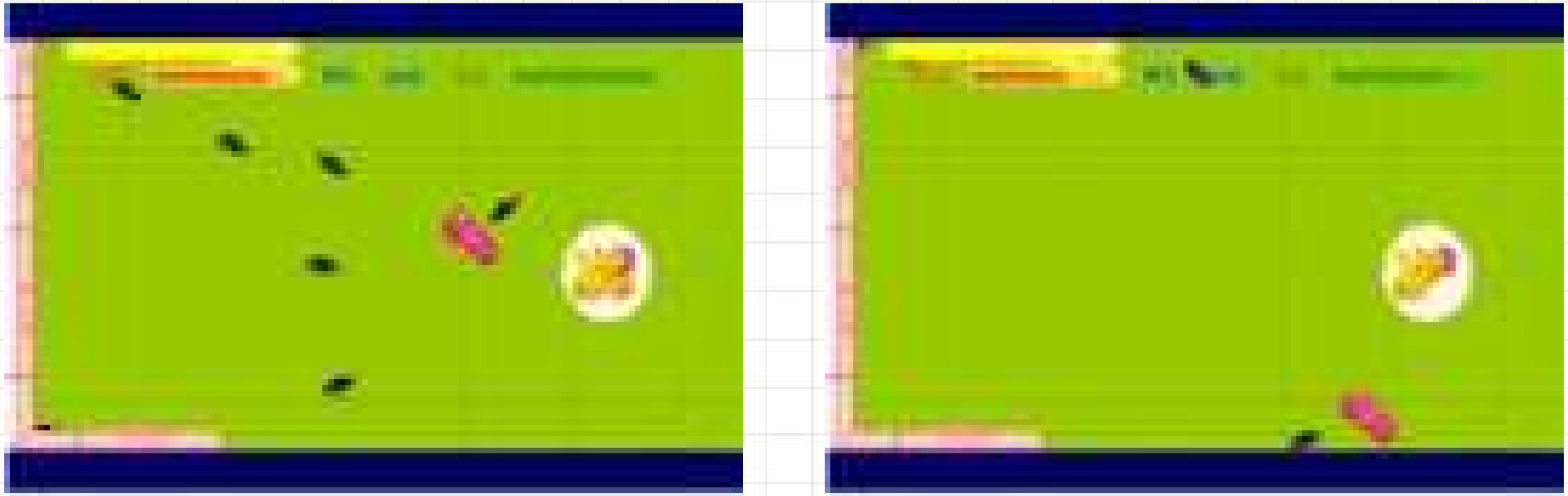
T-RPCA

M-RPCA (mode 2)

Low-rank static background reconstruction

- Game data:
  - 27 colored frames
  - Each has a resolution 86×130
  - Form the tensor data: 86×130×3×27

2 out of 27 frames



Low-rank static background reconstruction

Reconstructed from 20% data

Background from T-RPCA

Background from M-RPCA

SQUARE DEAL – AN IMPROVED CONVEX MODEL FOR HIGH-ORDER TENSORS (K>3)

Square Deal: Gaussian measurement

- Consider low-rank tensor recovery under the Gaussian measurements.

- Vector (non-convex) model: Pareto Opt.

$$\text{minimize}_{(\text{w.r.t. } \mathbb{R}_+^K)} \quad \text{rank}_{\text{tc}}(\boldsymbol{\mathcal{X}}) \quad \text{subject to} \quad \mathcal{G}[\boldsymbol{\mathcal{X}}] = \mathcal{G}[\boldsymbol{\mathcal{X}}_0]$$

- The Sum of Nuclear Norms (SNN) relaxation

$$\text{minimize} \quad \sum_{i=1}^{K} \lambda_i \|\boldsymbol{\mathcal{X}}_{(i)}\|_* \quad \text{subject to} \quad \mathcal{G}[\boldsymbol{\mathcal{X}}] = \mathcal{G}[\boldsymbol{\mathcal{X}}_0]$$

Square Deal: Complexity

- For simplicity, consider the K-way tensor of length n and Tucker rank r.

- For exact recovery, the number of Gaussian measurement needed for Non-convex vector optimization and SNN models are
  - Non-convex: $O(r^K + nrK)$
  - SNN: $O(rn^{K-1})$

- Big gap when n is large!

Square Deal: new convex objective

- *Square Deal:* An improved convex surrogate for the Tucker rank.
- When K=4, instead of unfolding $\mathcal{X}_0$ into a flat matrix, i.e., $\mathcal{X}_{(i)} \in \mathbb{R}^{n \times n^3}$, reshape it into a square matrix $\mathcal{X}_{\square} \in \mathbb{R}^{n^2 \times n^2}$, i.e.,

$$(\mathcal{X}_{\square})_{a+(b-1)n, c+(d-1)n} = (\mathcal{X}_0)_{a,b,c,d}$$

- Same idea applies when K>4: make $\mathcal{X}_{\square}$ as "square" as possible.

Square Deal: Main results

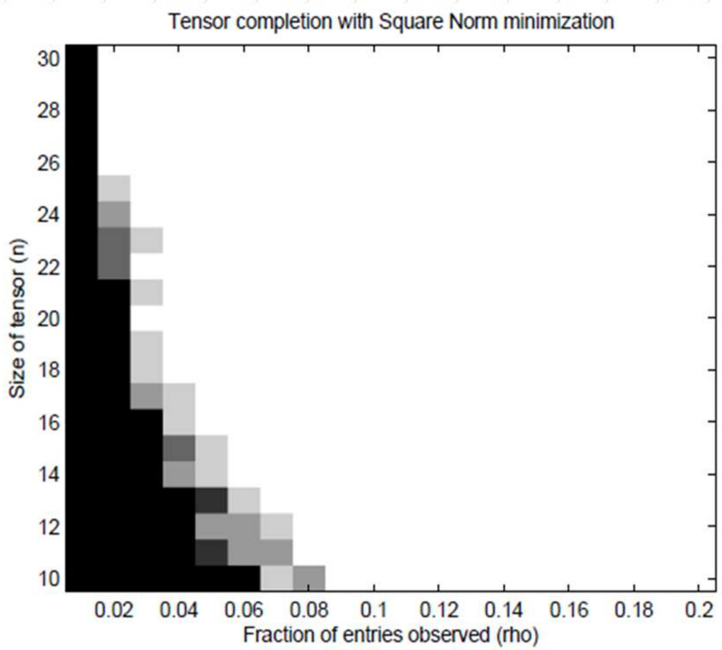
Theorem:

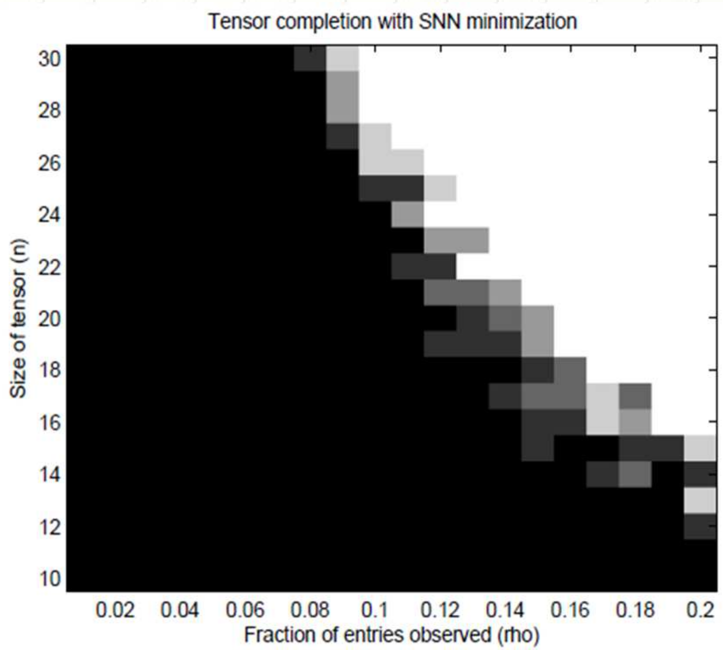
(1) If $\mathcal{X}_0$ has CP rank r, using Square Deal, $m \geq Crn^{\lceil \frac{K}{2} \rceil}$ Gaussian measurements are sufficient to recover $\mathcal{X}_0$;

(2) If $\mathcal{X}_0$ has Tucker rank r, using Square Deal, $m \geq Cr^{\lfloor \frac{K}{2} \rfloor} n^{\lceil \frac{K}{2} \rceil}$ Gaussian measurements are sufficient to recover $\mathcal{X}_0$

Square Deal: Simulation

- Consider the Tensor-Completion problem for a 4-way tensor $\mathcal{X}_0 \in \mathbb{R}^{n\times n\times n\times n}$ with the core tensor $\mathcal{C}_0 \in \mathbb{R}^{1\times 1\times 2\times 2}$, each entree is i.i.d. std. Gaussian, and each element in $\Omega \sim Ber(\rho)$ is chosen randomly.

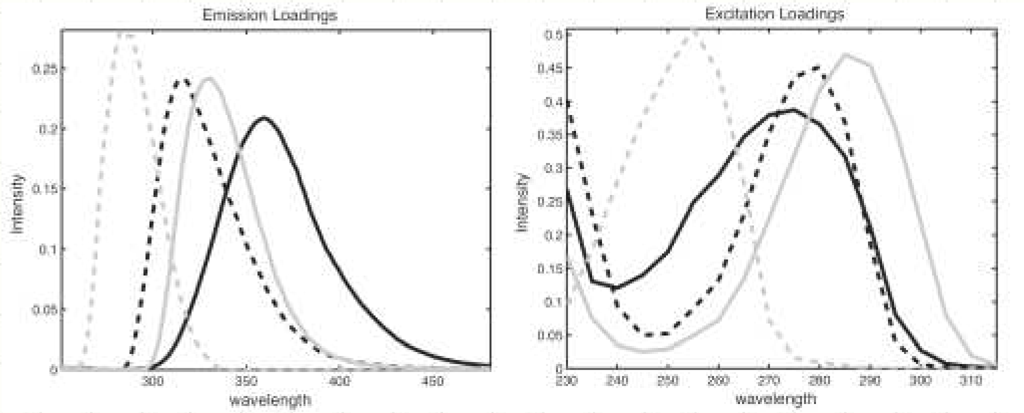




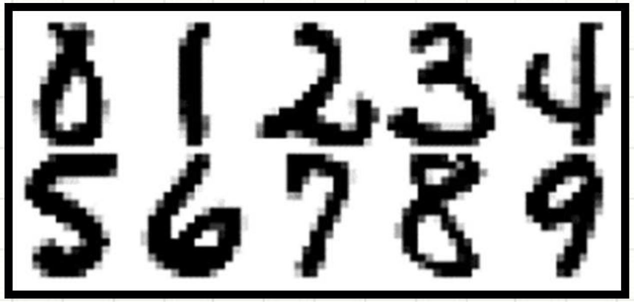
Questions?

Applications

- **Chemometrics – fluorescence excitation-emission**

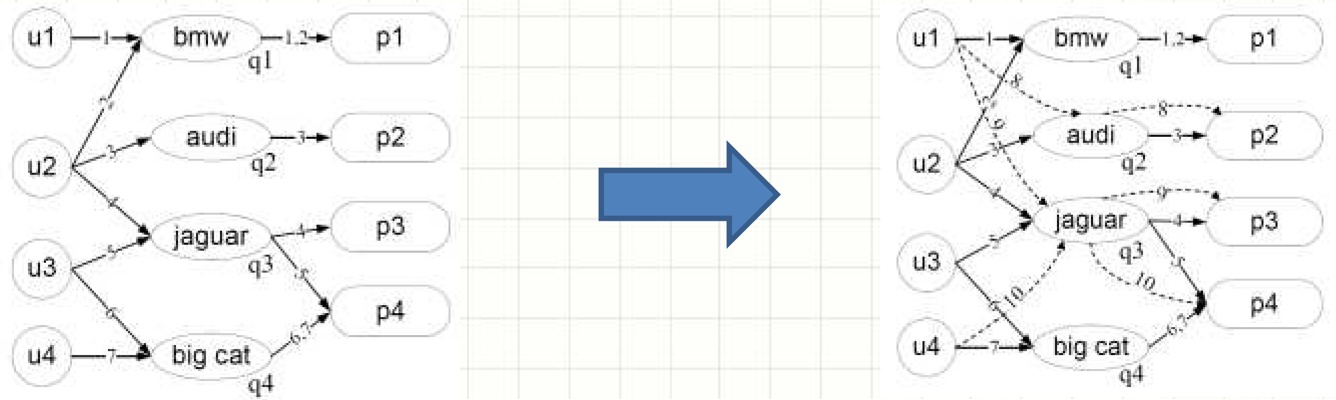


- **Hand-written digits recognition (Savas & Eldén, 2005)**



Applications

- **Personalized Web Search (Sun, et. al., 2005)**



Future works

- Better tensor incoherence conditions;
- Theoretical evidence on the advantage of TRPCA over the regular RPCA;
- More efficient algorithm suitable for strongly convex programming;
- Extend the square deal to the case k<4;
- More interesting applications for low-rank tensor recovery problems.