Online Sparse + Low-Rank Matrix Recovery

Namrata Vaswani

Iowa State University
Web: http://www.ece.iastate.edu/~namrata

(joint work with Wei Lu, Chenlu Qiu and Brian Lois)

Acknowledgements

- This talk is based on joint work with my students
  - Wei Lu and Jinchun Zhan (online sparse matrix recovery – Modified-CS)
  - Chenlu Qiu and Brian Lois (online sparse + low-rank matrix recovery / robust PCA)
- Funded by NSF grants CCF-1117125 and CCF-0917015
- Other collaborators: Han Guo (new student) and Prof. Leslie Hogben (Math, ISU)

Recovery from incomplete data: the question

- In many applications, data acquisition is slow, e.g. in MRI, acquire one Fourier coefficient of the cross-section of interest at a time
  - Question: can we recover the cross-section's image from undersampled data?

Recovery from incomplete data: the question

- In many applications, data acquisition is slow, e.g. in MRI, acquire one Fourier coefficient of the cross-section of interest at a time
  - Question: can we recover the cross-section's image from undersampled data?
  - Yes: if spatially-limited or if exploit sparsity of the image in an appropriate domain

Recovery from incomplete data: the question

- In many applications, data acquisition is slow, e.g. in MRI, acquire one Fourier coefficient of the cross-section of interest at a time
  - Question: can we recover the cross-section's image from undersampled data?
  - Yes: if spatially-limited or if exploit sparsity of the image in an appropriate domain
- In many other applications, data acquisition is fast but cannot see everything, e.g. in video,

$$\text{image} = \text{background} + \text{foreground}$$

  - Question: can we recover two image sequences from one?

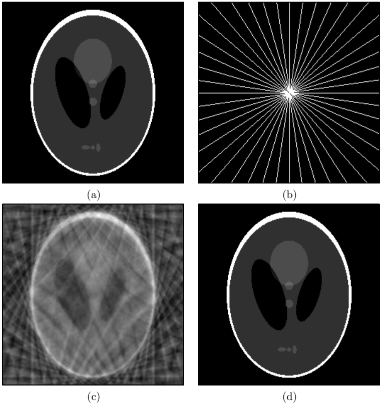
Recovery from incomplete data: the question

- In many applications, data acquisition is slow, e.g. in MRI, acquire one Fourier coefficient of the cross-section of interest at a time
  - Question: can we recover the cross-section's image from undersampled data?
  - Yes: if spatially-limited or if exploit sparsity of the image in an appropriate domain
- In many other applications, data acquisition is fast but cannot see everything, e.g. in video,

$$\text{image} = \text{background} + \text{foreground}$$

  - Question: can we recover two image sequences from one?
  - Yes: if exploit the low-rank structure of the background sequence and sparseness of the foreground

Sparse recovery: Magnetic Resonance Imaging (MRI)



Example taken from [Candes,Romberg,Tao,T-IT, Feb 2006]

- (a) Shepp-Logan phantom: 256×256 image
- (b) MR imaging pattern: 256-point DFT along 22 radial lines
- (c) Inverse-DFT
- (d) Basis Pursuit solution (uses sparsity: gives exact recovery!)

Sparse recovery / Compressive sensing [Mallat et al'93], [Feng,Bresler'96], [Gordinsky,Rao'97], [Chen,Donoho'98], [Candes,Romberg,Tao'05],[Donoho'05]

- Recover a sparse vector x, with support size at most s, from

$$y := Ax + w$$

when A is a known fat matrix and $\|w\|_2 \leq \epsilon$ (small noise).

Sparse recovery / Compressive sensing [Mallat et al'93], [Feng,Bresler'96], [Gordinsky,Rao'97], [Chen,Donoho'98], [Candes,Romberg,Tao'05],[Donoho'05]

- Recover a sparse vector x, with support size at most s, from

$$y := Ax + w$$

when A is a known fat matrix and $\|w\|_2 \leq \epsilon$ (small noise).

Sparse recovery / Compressive sensing [Mallat et al'93], [Feng,Bresler'96], [Gordinsky,Rao'97], [Chen,Donoho'98], [Candes,Romberg,Tao'05],[Donoho'05]

- Recover a sparse vector x, with support size at most s, from

$$y := Ax + w$$

when A is a known fat matrix and $\|w\|_2 \leq \epsilon$ (small noise).

- Solution by convex relaxation: ℓ_1 minimization [Chen,Donoho'98]:

$$\min \|\tilde{x}\|_1 \text{ subject to } \|y - A\tilde{x}\|_2 \leq \epsilon$$

if $\delta_{2s}(A) < 0.4$, error bounded by $C\epsilon$ [Candes et al'05,'06,'08]

  - restricted isometry constant (RIC) $\delta_s(A)$: smallest real # s.t.

$$(1-\delta_s)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1+\delta_s)\|x\|_2^2$$

  for all s-sparse vectors x [Candes,Tao,T-IT'05]

Sparse recovery / Compressive sensing [Mallat et al'93], [Feng,Bresler'96], [Gordinsky,Rao'97], [Chen,Donoho'98], [Candes,Romberg,Tao'05],[Donoho'05]

- Recover a sparse vector x, with support size at most s, from

$$y := Ax + w$$

  when A is a known fat matrix and $\|w\|_2 \leq \epsilon$ (small noise).

- Solution by convex relaxation: ℓ_1 minimization [Chen,Donoho'98]:

$$\min \|\tilde{x}\|_1 \text{ subject to } \|y - A\tilde{x}\|_2 \leq \epsilon$$

  if $\delta_{2s}(A) < 0.4$, error bounded by $C\epsilon$ [Candes et al'05,'06,'08]

  - restricted isometry constant (RIC) $\delta_s(A)$: smallest real # s.t.

$$(1 - \delta_s)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta_s)\|x\|_2^2$$

    for all s-sparse vectors x [Candes,Tao,T-IT'05]

- Applications: projection imaging - MRI, CT, astronomy, single-pixel camera

Low-rank matrix recovery (completion)

- Recover a low-rank matrix from a subset of its entries

$$Y := \mathcal{P}_\Omega(L)$$

Ω is the set of missing entries [Fazel et al, Recht et al,2009]

- Applications: recommendation system design, e.g. Netflix problem; survey data analysis, ...
  - ℓ_k: ratings of movies by user k
  - a given user will rate only a subset of all the movies: missing entries; goal: complete the matrix in order to recommend movies
  - matrix is low-rank: user preferences governed by only a few factors

Sparse + Low-rank matrix recovery

- Separate a low-rank matrix L and a sparse matrix X from

$$Y := X + L$$

  or from a subset of entries of $(X + L)$
  - if L or range(L) is the quantity of interest: robust PCA
  - if X is quantity of interest: robust sparse recovery

- Applications: video analytics (e.g. for surveillance, tracking, mobile video chat, occlusion removal,...) [Candes et al,2009]

$$X = [x_1, x_2 \ldots, x_t, \ldots x_{t_{\max}}], \; L = [\ell_1, \ell_2, \ldots \ell_t, \ldots \ell_{t_{\max}}]$$

  - ℓ_t: bg - usually slow changing, global (dense) changes
  - x_t: fg - sparse, consists of one or more moving objects (technically x_t: (fg-bg) on fg support)

Sparse + Low-rank matrix recovery

- Separate a low-rank matrix L and a sparse matrix X from

$$Y := X + L$$

or from a subset of entries of $(X + L)$
  - if L or range(L) is the quantity of interest: robust PCA
  - if X is quantity of interest: robust sparse recovery

- Applications: video analytics (e.g. for surveillance, tracking, mobile video chat, occlusion removal,...) [Candes et al,2009]

$$X = [x_1, x_2 \ldots, x_t, \ldots x_{t_{\max}}], \; L = [\ell_1, \ell_2, \ldots \ell_t, \ldots \ell_{t_{\max}}]$$

  - ℓ_t: bg - usually slow changing, global (dense) changes
  - x_t: fg - sparse, consists of one or more moving objects (technically x_t: (fg-bg) on fg support)

- Other apps: detecting anomalous connectivity patterns in social networks or in computer networks; functional MRI based brain activity detection; recommendation system design

Our work: the question

- How to solve the above problems for dynamically arriving data?
  - e.g., dynamic or functional MRI, online video analytics, ...
- Option 1: batch methods
  - recover the entire sequence in a batch fashion (e.g. for sparse recovery - use Fourier sparsity along the time axis)
  - slow and memory-intensive

Our work: the question

- How to solve the above problems for dynamically arriving data?
  - e.g., dynamic or functional MRI, online video analytics, ...
- Option 1: batch methods
  - recover the entire sequence in a batch fashion (e.g. for sparse recovery - use Fourier sparsity along the time axis)
  - slow and memory-intensive
- Option 2: do not use past knowledge
  - fast and memory-efficient, but will need more measurements

Our work: the question

- How to solve the above problems for dynamically arriving data?
  - e.g., dynamic or functional MRI, online video analytics, ...
- Option 1: batch methods
  - recover the entire sequence in a batch fashion (e.g. for sparse recovery - use Fourier sparsity along the time axis)
  - slow and memory-intensive
- Option 2: do not use past knowledge
  - fast and memory-efficient, but will need more measurements
- Option 3: design recursive algorithms (our work)
  - use previously recovered images and current observed data to recover the current image
  - fast and memory-efficient and need fewer measurements

Our work: Online (recursive) solutions

- Developed provably accurate recursive solutions for
  - "online" sparse matrix recovery (recursive recovery of sparse signal sequences) [KF-CS, ICIP'08]
    - brief overview
  - "online" sparse + low-rank matrix recovery (online or recursive robust PCA) [Qiu,Vaswani,Allerton 2010]
    - most of this talk
- **The "online" problem as we define it uses extra assumptions**
- **In this talk "recursive" $\Leftrightarrow$ "online" (used interchangeably)**

Recursive recovery of sparse seq's: Problem [Vaswani,ICIP'08][1]

- Given measurements
$$y_t := Ax_t + w_t, \quad \|w_t\|_2 \le \epsilon, \quad t = 0, 1, 2, \ldots$$
  - $A = H\Phi$ (given): $n \times m$, $n < m$
    - H: measurement matrix, Φ: sparsity basis matrix
    - e.g., in MRI: H = partial Fourier, Φ = inverse wavelet
  - y_t: measurements (given)
  - x_t: sparsity basis vector
  - $\mathcal{N}_t$: support set of x_t
  - w_t: small noise
- Goal: recursively reconstruct x_t from $y_0, y_1, \ldots y_t$,
  - i.e. use only y_t and $\hat{x}_{t-1}$ for recovering x_t

[1] N. Vaswani, Kalman Filtered Compressed Sensing, ICIP, 2008

Recursive recovery of sparse seq's: Problem [Vaswani,ICIP'08][1]

- Given measurements
$$y_t := Ax_t + w_t, \quad \|w_t\|_2 \leq \epsilon, \quad t = 0, 1, 2, \ldots$$
  - $A = H\Phi$ (given): $n \times m$, $n < m$
    - H: measurement matrix, Φ: sparsity basis matrix
    - e.g., in MRI: H = partial Fourier, Φ = inverse wavelet
  - y_t: measurements (given)
  - x_t: sparsity basis vector
  - $\mathcal{N}_t$: support set of x_t
  - w_t: small noise
- Goal: recursively reconstruct x_t from $y_0, y_1, \ldots y_t$,
  - i.e. use only y_t and $\hat{x}_{t-1}$ for recovering x_t
- Use slow support change: $|\mathcal{N}_t \setminus \mathcal{N}_{t-1}| \approx |\mathcal{N}_{t-1} \setminus \mathcal{N}_t| \ll |\mathcal{N}_t|$
  - also use slow signal value change when valid

[1] N. Vaswani, Kalman Filtered Compressed Sensing, ICIP, 2008

Recursive recovery of sparse seq's: Solutions [KF-CS, ICIP'08], [LS-CS,T-SP,Aug10]

- Introduced Kalman filtered CS (KF-CS) and Least Squares CS (LS-CS):
  - first recursive algorithms that needed fewer measurements for accurate recovery than simple ℓ_1
  - able to obtain time-invariant error bounds on LS-CS error under weaker RIP assumptions (fewer meas's) than simple ℓ_1
- But these could not achieve *exact* recovery with fewer meas's than what simple ℓ_1 needed
  - solved by Modified-CS

Recursive recovery of sparse seq's: Modified-CS [Modified-CS,ISIT'09,T-SP'10,T-IT'15]

- Idea: support at $t-1$, $\mathcal{N}_{t-1}$, is a good predictor of $\mathcal{N}_t$
- Reformulate: Sparse Recovery with Partial Support Knowledge $\mathcal{T}$
  - $\text{support}(x) = \mathcal{T} \cup \Delta \setminus \Delta_e$: Δ, Δ_e unknown

Recursive recovery of sparse seq's: Modified-CS [Modified-CS,ISIT'09,T-SP'10,T-IT'15]

- Idea: support at $t-1$, $\mathcal{N}_{t-1}$, is a good predictor of $\mathcal{N}_t$
- Reformulate: Sparse Recovery with Partial Support Knowledge $\mathcal{T}$
  - support$(x) = \mathcal{T} \cup \Delta \setminus \Delta_e$: Δ, Δ_e unknown
- Modified-CS: tries to find a vector $\tilde{x}$ that is sparsest outside $\mathcal{T}$ among all vectors satisfying the data constraint

$$\min_{\tilde{x}} \|\tilde{x}_{\mathcal{T}^c}\|_1 \text{ subject to } \|y - A\tilde{x}\|_2 \leq \epsilon$$

Recursive recovery of sparse seq's: Modified-CS [Modified-CS,ISIT'09,T-SP'10,T-IT'15]

- Idea: support at $t-1$, $\mathcal{N}_{t-1}$, is a good predictor of $\mathcal{N}_t$
- Reformulate: Sparse Recovery with Partial Support Knowledge $\mathcal{T}$
  - support$(x) = \mathcal{T} \cup \Delta \setminus \Delta_e$: Δ, Δ_e unknown
- Modified-CS: tries to find a vector $\tilde{x}$ that is sparsest outside $\mathcal{T}$ among all vectors satisfying the data constraint

$$\min_{\tilde{x}} \|\tilde{x}_{\mathcal{T}^c}\|_1 \text{ subject to } \|y - A\tilde{x}\|_2 \leq \epsilon$$

- Provably exact recovery in noise-free case if $\delta_{s+|\Delta|+|\Delta_e|} < 0.4$ [Vaswani,Lu, ISIT'09,T-SP'10]

Recursive recovery of sparse seq's: Modified-CS [Modified-CS,ISIT'09,T-SP'10,T-IT'15]

- Idea: support at $t-1$, $\mathcal{N}_{t-1}$, is a good predictor of $\mathcal{N}_t$
- Reformulate: Sparse Recovery with Partial Support Knowledge $\mathcal{T}$
  - $\text{support}(x) = \mathcal{T} \cup \Delta \setminus \Delta_e$: Δ, Δ_e unknown
- Modified-CS: tries to find a vector $\tilde{x}$ that is sparsest outside $\mathcal{T}$ among all vectors satisfying the data constraint

$$\min_{\tilde{x}} \|\tilde{x}_{\mathcal{T}^c}\|_1 \text{ subject to } \|y - A\tilde{x}\|_2 \leq \epsilon$$

- Provably exact recovery in noise-free case if $\delta_{s+|\Delta|+|\Delta_e|} < 0.4$ [Vaswani,Lu, ISIT'09,T-SP'10]
- For noisy case: time-invariant error bounds under a realistic signal change model and $\delta_{s+ks_a} < 0.4$ [Zhan,Vaswani, ISIT'13, T-IT'15 (to appear)]

Recursive recovery of sparse seq's: Modified-CS [Modified-CS,ISIT'09,T-SP'10,T-IT'15]

- Idea: support at $t-1$, $\mathcal{N}_{t-1}$, is a good predictor of $\mathcal{N}_t$
- Reformulate: Sparse Recovery with Partial Support Knowledge $\mathcal{T}$
  - $\text{support}(x) = \mathcal{T} \cup \Delta \setminus \Delta_e$: Δ, Δ_e unknown
- Modified-CS: tries to find a vector $\tilde{x}$ that is sparsest outside $\mathcal{T}$ among all vectors satisfying the data constraint

$$\min_{\tilde{x}} \|\tilde{x}_{\mathcal{T}^c}\|_1 \text{ subject to } \|y - A\tilde{x}\|_2 \leq \epsilon$$

- Provably exact recovery in noise-free case if $\delta_{s+|\Delta|+|\Delta_e|} < 0.4$ [Vaswani,Lu, ISIT'09,T-SP'10]
- For noisy case: time-invariant error bounds under a realistic signal change model and $\delta_{s+ks_a} < 0.4$ [Zhan,Vaswani, ISIT'13, T-IT'15 (to appear)]
- Regularized modified-CS & modified-CS-residual: also use slow signal value change (when valid);
  - significant advantage over existing work for dynamic MRI

Online Robust PCA: background

- Principal Components' Analysis (PCA): estimate the low-dimensional subspace that best approximates a given dataset
  - SVD on data matrix, compute top left singular vectors
- Robust PCA: PCA in presence of outliers; many useful heuristics in older work, e.g., RSL [De la Torre et al,2003]
- Online robust PCA: start with a good initial estimate of the low-dimensional subspace, keep updating it as more data comes in, while being robust to outliers

Online Robust PCA: background

- Principal Components' Analysis (PCA): estimate the low-dimensional subspace that best approximates a given dataset
  - SVD on data matrix, compute top left singular vectors
- Robust PCA: PCA in presence of outliers; many useful heuristics in older work, e.g., RSL [De la Torre et al,2003]
- Online robust PCA: start with a good initial estimate of the low-dimensional subspace, keep updating it as more data comes in, while being robust to outliers
- [Candes et al,2009] posed robust PCA as: separate low-rank matrix L, sparse X from

$$Y := X + L$$

A practical provably correct solution: PCP

- [Candes et al,2009; Chandrasekharan et al,2009; Hsu et al,2011] introduced and studied a convex opt program called PCP:

$$\min_{\tilde{X},\tilde{L}} \|\tilde{L}\|_* + \lambda\|\tilde{X}\|_1 \text{ s.t. } Y = \tilde{X} + \tilde{L}$$

- If (a) left and right singular vectors of L are dense enough; (b) support of X is generated uniformly at random; (c) rank and sparsity are bounded, then PCP exactly recovers X and L from $Y := X + L$ w.h.p. [Candes et al,2009]
  - [Chandrasekharan et al,2009; Hsu et al,2011]: similar flavor; replace 'unif rand support' by upper bound on # of nonzeros in any row of X.
  - first set of guarantees for a practical robust PCA approach

A practical provably correct solution: PCP

- [Candes et al,2009; Chandrasekharan et al,2009; Hsu et al,2011] introduced and studied a convex opt program called PCP:

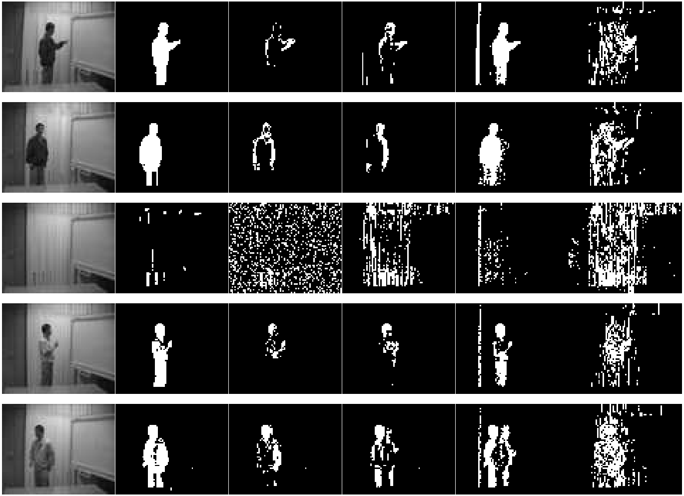
$$\min_{\tilde{X},\tilde{L}} \|\tilde{L}\|_* + \lambda \|\tilde{X}\|_1 \text{ s.t. } Y = \tilde{X} + \tilde{L}$$

- If (a) left and right singular vectors of L are dense enough; (b) support of X is generated uniformly at random; (c) rank and sparsity are bounded, then PCP exactly recovers X and L from $Y := X + L$ w.h.p. [Candes et al,2009]
  - [Chandrasekharan et al,2009; Hsu et al,2011]: similar flavor; replace 'unif rand support' by upper bound on # of nonzeros in any row of X.
  - first set of guarantees for a practical robust PCA approach
- Much later work on the *batch* robust PCA problem w/ guarantees

Need for an online method

- Disadvantages of batch methods:
  - slower especially for online applications;
  - memory intensive;
  - do not allow infrequent/slow support change of columns of X
    - reason: this can result in X being rank deficient
- Video analytics: need online solution; and have occasionally static or slow moving fg objects
- Functional MRI: the activated brain region does not change a lot from frame to frame
- Network anomaly detection: need online solution; anomalous behavior continues for a period of time after begins

Figure : ReProCS: proposed. Frames $t = t_0 + 60, 120, 199, 475, 1148$.



original | ReProCS (online) | PCP (batch) | RSL (batch) | GRASTA (online) | MG (batch)

Figure : ReProCS: proposed. Frames $t = t_0 + 60, 120, 199, 475, 1148$.

"Online" sparse + low-rank recovery / robust PCA problem

[Qiu,Vaswani,Allerton'10,'11] [Guo,Qiu,Vaswani,T-SP'14] [2]

- Given sequentially arriving n-length data vectors y_t satisfying

$$y_t := \ell_t, \quad t = 1, 2, \ldots, t_0$$

and

$$y_t := x_t + \ell_t, \quad t = t_0 + 1, t_0 + 2, \ldots, t_{\max}$$

  - x_t's are sparse vectors with support sets, $\mathcal{T}_t$, of size at most s;
  - its support sets $\mathcal{T}_t$ have *at least some* changes over time

[2] C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010
H. Guo, C. Qiu, N. Vaswani, An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans.SP, Aug 2014

"Online" sparse + low-rank recovery / robust PCA problem

[Qiu,Vaswani,Allerton'10,'11] [Guo,Qiu,Vaswani,T-SP'14] [2]

- Given sequentially arriving n-length data vectors y_t satisfying

$$y_t := \ell_t, \quad t = 1, 2, \ldots, t_0$$

and

$$y_t := x_t + \ell_t, \quad t = t_0 + 1, t_0 + 2, \ldots, t_{\max}$$

  - x_t's are sparse vectors with support sets, $\mathcal{T}_t$, of size at most s;
  - its support sets $\mathcal{T}_t$ have *at least some* changes over time
  - ℓ_t's lie in a slowly-changing low-dimensional subspace of $\mathbb{R}^n$;
    - $\Leftrightarrow \ell_t = P_t a_t$ w/ $\|(I - P_{t-1}P_{t-1}{}')\ell_t\|_2 \ll \|\ell_t\|_2$ (P_t: tall)

[2] C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010
H. Guo, C. Qiu, N. Vaswani, An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans.SP, Aug 2014

"Online" sparse + low-rank recovery / robust PCA problem

[Qiu,Vaswani,Allerton'10,'11] [Guo,Qiu,Vaswani,T-SP'14] [2]

- Given sequentially arriving n-length data vectors y_t satisfying

$$y_t := \ell_t, \quad t = 1, 2, \ldots, t_0$$

and

$$y_t := x_t + \ell_t, \quad t = t_0 + 1, t_0 + 2, \ldots, t_{\max}$$

  - x_t's are sparse vectors with support sets, $\mathcal{T}_t$, of size at most s;
  - its support sets $\mathcal{T}_t$ have *at least some* changes over time
  - ℓ_t's lie in a slowly-changing low-dimensional subspace of $\mathbb{R}^n$;
    - $\Leftrightarrow \ell_t = P_t a_t$ w/ $\|(I - P_{t-1}P_{t-1}{}')\ell_t\|_2 \ll \|\ell_t\|_2$ (P_t: tall)
  - left singular vectors of the matrix $L_t := [\ell_1, \ell_2, \ldots \ell_t]$ are dense

[2] C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010
H. Guo, C. Qiu, N. Vaswani, An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans.SP, Aug 2014

"Online" sparse + low-rank recovery / robust PCA problem

[Qiu,Vaswani,Allerton'10,'11] [Guo,Qiu,Vaswani,T-SP'14] [2]

- Given sequentially arriving n-length data vectors y_t satisfying

$$y_t := \ell_t, \quad t = 1, 2, \ldots, t_0$$

and

$$y_t := x_t + \ell_t, \quad t = t_0 + 1, t_0 + 2, \ldots, t_{\max}$$

  - x_t's are sparse vectors with support sets, $\mathcal{T}_t$, of size at most s;
  - its support sets $\mathcal{T}_t$ have *at least some* changes over time
  - ℓ_t's lie in a slowly-changing low-dimensional subspace of $\mathbb{R}^n$;
    - $\Leftrightarrow \ell_t = P_t a_t$ w/ $\|(I - P_{t-1}P_{t-1}{}')\ell_t\|_2 \ll \|\ell_t\|_2$ (P_t: tall)
  - left singular vectors of the matrix $L_t := [\ell_1, \ell_2, \ldots \ell_t]$ are dense
- Goal: recursively estimate x_t, ℓ_t and range(L_t) at all $t > t_0$.

[2] C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010
H. Guo, C. Qiu, N. Vaswani, An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans.SP, Aug 2014

"Online" sparse + low-rank recovery / robust PCA problem

[Qiu,Vaswani,Allerton'10,'11] [Guo,Qiu,Vaswani,T-SP'14] [3]

- Initial outlier-free seq $y_t = \ell_t$ for first t_0 frames needed to estimate the initial subspace P_{t_0}. Easy to obtain in many apps, e.g.,
  - in video surveillance, easy to get a short background-only training sequence before fg objects start appearing
  - for fMRI, this corresponds to acquiring a short sequence without any activation

[3] C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010
H. Guo, C. Qiu, N. Vaswani, An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans.SP, Aug 2014

"Online" sparse + low-rank recovery / robust PCA problem

[Qiu,Vaswani,Allerton'10,'11] [Guo,Qiu,Vaswani,T-SP'14] [3]

- Initial outlier-free seq $y_t = \ell_t$ for first t_0 frames needed to estimate the initial subspace P_{t_0}. Easy to obtain in many apps, e.g.,
  - in video surveillance, easy to get a short background-only training sequence before fg objects start appearing
  - for fMRI, this corresponds to acquiring a short sequence without any activation
- Note: extension of all our ideas to the undersampled case $y_t = Ax_t + B\ell_t$ is easy (relevant to MRI apps)

[3] C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010
H. Guo, C. Qiu, N. Vaswani, An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans.SP, Aug 2014

Related work

Batch robust PCA and performance guarantees

- Older work, e.g. RSL [de la Torre et al,IJCV'03]; PCP and much later work on provably correct robust PCA solutions

Recursive / incremental / online robust PCA algorithms

- Older work (before PCP): [Li et al, ICIP 2003] iRSL: doesn't work
- [Qiu,Vaswani, Allerton'10, Allerton'11, T-SP'14]: ReProCS (Recursive Projected CS)
- [Balzano et al, CVPR 2012]: GRASTA
- [Mateos et al, JSTSP 2013]: batch, online; online: not enough info, no code

Online robust PCA performance guarantees: almost no work

- [Qiu,Vaswani,Lois,Hogben, ISIT'13, T-IT'14]: partial result;
- [Lois, Vaswani,ICASSP'15,arXiv:1409.3959]: complete correctness result
- [Feng et al,NIPS'13 OR-PCA Stoch Opt]: partial result and only asymptotic

Some definitions

- P is a basis matrix $\Leftrightarrow P'P = I$
- "Estimate P" $\Leftrightarrow$ estimate range(P): subspace spanned by col's of P
- "$\hat{P}$ is an accurate estimate of P" $\Leftrightarrow$ $\text{SE}(\hat{P}, P) := \|(I - \hat{P}\hat{P}')P\|_2 \ll 1$

ReProCS algorithm [Qiu,Vaswani,Allerton'10,Allerton'11],[Guo,Qiu,Vaswani,T-SP'14][4]

Recall: for $t > t_0$, $y_t := x_t + \ell_t$, $\ell_t = P_t a_t$, P_t: tall $n \times r$ basis matrix

[4] C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010
H. Guo, C. Qiu, N. Vaswani, An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans.SP, Aug 2014

ReProCS algorithm [Qiu,Vaswani,Allerton'10,Allerton'11],[Guo,Qiu,Vaswani,T-SP'14][4]

Recall: for $t > t_0$, $y_t := x_t + \ell_t$, $\ell_t = P_t a_t$, P_t: tall $n \times r$ basis matrix

Initialize: compute $\hat{P}_0$ = top left singular vectors of $[\ell_1, \ell_2, \dots \ell_{t_0}]$.

[4] C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010
H. Guo, C. Qiu, N. Vaswani, An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans.SP, Aug 2014

ReProCS algorithm [Qiu,Vaswani,Allerton'10,Allerton'11],[Guo,Qiu,Vaswani,T-SP'14][4]

Recall: for $t > t_0$, $y_t := x_t + \ell_t$, $\ell_t = P_t a_t$, P_t: tall $n \times r$ basis matrix

Initialize: compute $\hat{P}_0$ = top left singular vectors of $[\ell_1, \ell_2, \ldots \ell_{t_0}]$.

For $t > t_0$, do

- Projection: compute $\tilde{y}_t := \Phi_t y_t$, where $\Phi_t := I - \hat{P}_{t-1}\hat{P}'_{t-1}$
  - then $\tilde{y}_t = \Phi_t x_t + \beta_t$, $\beta_t := \Phi_t \ell_t$ is small "noise" because of slow subspace change

[4] C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010
H. Guo, C. Qiu, N. Vaswani, An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans.SP, Aug 2014

ReProCS algorithm [Qiu,Vaswani,Allerton'10,Allerton'11],[Guo,Qiu,Vaswani,T-SP'14][4]

Recall: for $t > t_0$, $y_t := x_t + \ell_t$, $\ell_t = P_t a_t$, P_t: tall $n \times r$ basis matrix

Initialize: compute $\hat{P}_0$ = top left singular vectors of $[\ell_1, \ell_2, \dots \ell_{t_0}]$.

For $t > t_0$, do

- Projection: compute $\tilde{y}_t := \Phi_t y_t$, where $\Phi_t := I - \hat{P}_{t-1}\hat{P}'_{t-1}$
  - then $\tilde{y}_t = \Phi_t x_t + \beta_t$, $\beta_t := \Phi_t \ell_t$ is small "noise" because of slow subspace change
- Noisy Sparse Recovery: ℓ_1 min + support estimate + LS: get $\hat{x}_t$
  - denseness of P_t's $\Rightarrow$ sparse x_t recoverable from $\tilde{y}_t$

[4] C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010
H. Guo, C. Qiu, N. Vaswani, An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans.SP, Aug 2014

ReProCS algorithm [Qiu,Vaswani,Allerton'10,Allerton'11],[Guo,Qiu,Vaswani,T-SP'14][4]

Recall: for $t > t_0$, $y_t := x_t + \ell_t$, $\ell_t = P_t a_t$, P_t: tall $n \times r$ basis matrix

Initialize: compute $\hat{P}_0$ = top left singular vectors of $[\ell_1, \ell_2, \dots \ell_{t_0}]$.

For $t > t_0$, do

- Projection: compute $\tilde{y}_t := \Phi_t y_t$, where $\Phi_t := I - \hat{P}_{t-1}\hat{P}'_{t-1}$
  - then $\tilde{y}_t = \Phi_t x_t + \beta_t$, $\beta_t := \Phi_t \ell_t$ is small "noise" because of slow subspace change
- Noisy Sparse Recovery: ℓ_1 min + support estimate + LS: get $\hat{x}_t$
  - denseness of P_t's $\Rightarrow$ sparse x_t recoverable from $\tilde{y}_t$
- Recover ℓ_t: compute $\hat{\ell}_t = y_t - \hat{x}_t$

[4] C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010
H. Guo, C. Qiu, N. Vaswani, An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans.SP, Aug 2014

ReProCS algorithm [Qiu,Vaswani,Allerton'10,Allerton'11],[Guo,Qiu,Vaswani,T-SP'14][4]

Recall: for $t > t_0$, $y_t := x_t + \ell_t$, $\ell_t = P_t a_t$, P_t: tall $n \times r$ basis matrix

Initialize: compute $\hat{P}_0$ = top left singular vectors of $[\ell_1, \ell_2, \dots \ell_{t_0}]$.

For $t > t_0$, do

- Projection: compute $\tilde{y}_t := \Phi_t y_t$, where $\Phi_t := I - \hat{P}_{t-1}\hat{P}'_{t-1}$
  - then $\tilde{y}_t = \Phi_t x_t + \beta_t$, $\beta_t := \Phi_t \ell_t$ is small "noise" because of slow subspace change
- Noisy Sparse Recovery: ℓ_1 min + support estimate + LS: get $\hat{x}_t$
  - denseness of P_t's $\Rightarrow$ sparse x_t recoverable from $\tilde{y}_t$
- Recover ℓ_t: compute $\hat{\ell}_t = y_t - \hat{x}_t$
- Subspace update: update $\hat{P}_t$ every α frames by projection-PCA

[4] C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010
H. Guo, C. Qiu, N. Vaswani, An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans.SP, Aug 2014

Why ReProCS works [Qiu,Vaswani,Lois,Hogben,T-IT,2014] [5]

- Slow subspace change: noise β_t seen by sparse recovery step is small
- Denseness of columns of $P_t \Rightarrow$ RIC of $\Phi_t = I - \hat{P}_{t-1}\hat{P}'_{t-1}$ is small
  - denseness assump: $(2s)\max_t \max_i \|(P_{t-1})_{i,:}\|_2^2 \le 0.09$
  - easy to show [Qiu,Vaswani,Lois,Hogben,T-IT,2014]:

$$\delta_{2s}(\Phi_t) = \max_{|T|\le 2s} \|I_T{}'\hat{P}_{t-1}\|_2^2 \le (2s)\max_i \|(\hat{P}_{t-1})_{i,:}\|_2^2 \le 0.09{+}0.05$$

(here: 0.05 is due to the small error b/w $\hat{P}_{t-1}$ and P_{t-1})

- Above two facts + any result for ℓ_1 min: x_t is accurately recovered; and hence $\ell_t = y_t - x_t$ is accurately recovered

[5] C. Qiu, N. Vaswani, B. Lois and L. Hogben, Recursive Robust PCA or Recursive Sparse Recovery in Large but Structured Noise, IEEE Trans. IT, 2014

Why ReProCS works [Qiu,Vaswani,Lois,Hogben,T-IT,2014] [5]

- Slow subspace change: noise β_t seen by sparse recovery step is small
- Denseness of columns of $P_t \Rightarrow$ RIC of $\Phi_t = I - \hat{P}_{t-1}\hat{P}'_{t-1}$ is small
  - denseness assump: $(2s) \max_t \max_i \|(P_{t-1})_{i,:}\|_2^2 \leq 0.09$
  - easy to show [Qiu,Vaswani,Lois,Hogben,T-IT,2014]:

$$\delta_{2s}(\Phi_t) = \max_{|T| \leq 2s} \|{I_T}' \hat{P}_{t-1}\|_2^2 \leq (2s) \max_i \|(\hat{P}_{t-1})_{i,:}\|_2^2 \leq 0.09+0.05$$

    (here: 0.05 is due to the small error b/w $\hat{P}_{t-1}$ and P_{t-1})
- Above two facts + any result for ℓ_1 min: x_t is accurately recovered; and hence $\ell_t = y_t - x_t$ is accurately recovered
- Most of the work: show accurate subspace recovery $\hat{P}_t \approx P_t$
  - std PCA results not applicable: $e_t := \ell_t - \hat{\ell}_t = x_t - \hat{x}_t$ correlated w/ ℓ_t

[5] C. Qiu, N. Vaswani, B. Lois and L. Hogben, Recursive Robust PCA or Recursive Sparse Recovery in Large but Structured Noise, IEEE Trans. IT, 2014

ReProCS algorithm: why projection-PCA needed

- let $e_t := \ell_t - \hat{\ell}_t = \hat{x}_t - x_t$
- perturbation seen by standard PCA,

$$\frac{1}{\alpha}\sum_t \hat{\ell}_t \hat{\ell}_t' - \frac{1}{\alpha}\sum_t \ell_t \ell_t' = \frac{1}{\alpha}\sum_t \ell_t e_t' + \left(\frac{1}{\alpha}\sum_t \ell_t e_t'\right)' + \frac{1}{\alpha}\sum_t e_t e_t'$$

- when e_t and ℓ_t uncorrelated & e_t zero mean: first two terms are close to zero w.h.p.

ReProCS algorithm: why projection-PCA needed

- let $e_t := \ell_t - \hat{\ell}_t = \hat{x}_t - x_t$
- perturbation seen by standard PCA,

$$\frac{1}{\alpha}\sum_t \hat{\ell}_t\hat{\ell}_t' - \frac{1}{\alpha}\sum_t \ell_t\ell_t' = \frac{1}{\alpha}\sum_t \ell_t e_t' + \left(\frac{1}{\alpha}\sum_t \ell_t e_t'\right)' + \frac{1}{\alpha}\sum_t e_t e_t'$$

- when e_t and ℓ_t uncorrelated & e_t zero mean: first two terms are close to zero w.h.p.
- in ReProCS, e_t is correlated with ℓ_t;

ReProCS algorithm: why projection-PCA needed

- let $e_t := \ell_t - \hat{\ell}_t = \hat{x}_t - x_t$
- perturbation seen by standard PCA,

$$\frac{1}{\alpha}\sum_t \hat{\ell}_t \hat{\ell}_t' - \frac{1}{\alpha}\sum_t \ell_t \ell_t' = \frac{1}{\alpha}\sum_t \ell_t e_t' + \left(\frac{1}{\alpha}\sum_t \ell_t e_t'\right)' + \frac{1}{\alpha}\sum_t e_t e_t'$$

- when e_t and ℓ_t uncorrelated & e_t zero mean: first two terms are close to zero w.h.p.
- in ReProCS, e_t is correlated with ℓ_t; thus first two terms are the dominant ones; if condition # of $\frac{1}{\alpha}\sum_t \ell_t \ell_t'$ large: perturbation not be small compared to its min eigenvalue

ReProCS algorithm: why projection-PCA needed

- let $e_t := \ell_t - \hat{\ell}_t = \hat{x}_t - x_t$
- perturbation seen by standard PCA,

$$\frac{1}{\alpha}\sum_t \hat{\ell}_t\hat{\ell}_t' - \frac{1}{\alpha}\sum_t \ell_t\ell_t' = \frac{1}{\alpha}\sum_t \ell_t e_t' + \left(\frac{1}{\alpha}\sum_t \ell_t e_t'\right)' + \frac{1}{\alpha}\sum_t e_t e_t'$$

- when e_t and ℓ_t uncorrelated & e_t zero mean: first two terms are close to zero w.h.p.
- in ReProCS, e_t is correlated with ℓ_t; thus first two terms are the dominant ones; if condition # of $\frac{1}{\alpha}\sum_t \ell_t\ell_t'$ large: perturbation not be small compared to its min eigenvalue
- by $\sin\theta$ theorem [Davis,Kahan, 1970],

$$\|(I - \hat{P}\hat{P}')P\|_2 \lesssim \frac{\|\text{perturbation}\|_2}{\lambda_{\min}(\frac{1}{\alpha}\sum_t \ell_t\ell_t') - \|\text{perturbation}\|_2}$$

(P: eigenvec's with nonzero eigenval's of $\frac{1}{\alpha}\sum_t \ell_t\ell_t'$)

ReProCS algorithm: why projection-PCA needed

- let $e_t := \ell_t - \hat{\ell}_t = \hat{x}_t - x_t$
- perturbation seen by standard PCA,

$$\frac{1}{\alpha}\sum_t \hat{\ell}_t\hat{\ell}_t' - \frac{1}{\alpha}\sum_t \ell_t\ell_t' = \frac{1}{\alpha}\sum_t \ell_t e_t' + \left(\frac{1}{\alpha}\sum_t \ell_t e_t'\right)' + \frac{1}{\alpha}\sum_t e_t e_t'$$

- when e_t and ℓ_t uncorrelated & e_t zero mean: first two terms are close to zero w.h.p.
- in ReProCS, e_t is correlated with ℓ_t; thus first two terms are the dominant ones; if condition # of $\frac{1}{\alpha}\sum_t \ell_t\ell_t'$ large: perturbation not be small compared to its min eigenvalue
- by $\sin\theta$ theorem [Davis,Kahan, 1970],

$$\|(I - \hat{P}\hat{P}')P\|_2 \lesssim \frac{\|\text{perturbation}\|_2}{\lambda_{\min}(\frac{1}{\alpha}\sum_t \ell_t\ell_t') - \|\text{perturbation}\|_2}$$

(P: eigenvec's with nonzero eigenval's of $\frac{1}{\alpha}\sum_t \ell_t\ell_t'$)

ReProCS correctness result [Lois,Vaswani, arXiV:1409.3959],[Qiu,Vaswani,Lois,Hogben,T-IT'14][6]

For most videos (i.e. w.p. at least $1 - n^{-10}$),

- the region occupied by the foreground objects (support of x_t) is exactly recovered at all times, and

[6] B. Lois and N. Vaswani, A Correctness Result for Online Robust PCA, arXiV:1409.3959.
C. Qiu, N. Vaswani, B. Lois and L. Hogben, Recursive Robust PCA or Recursive Sparse Recovery in Large but Structured Noise, IEEE Trans. IT, 2014.

ReProCS correctness result [Lois,Vaswani, arXiV:1409.3959],[Qiu,Vaswani,Lois,Hogben,T-IT'14][6]

For most videos (i.e. w.p. at least $1 - n^{-10}$),

- the region occupied by the foreground objects (support of x_t) is exactly recovered at all times, and
- foreground and background images are accurately recovered at all times ($\|x_t - \hat{x}_t\|_2 = \|\ell_t - \hat{\ell}_t\|_2 \leq b$)

[6] B. Lois and N. Vaswani, A Correctness Result for Online Robust PCA, arXiV:1409.3959.
C. Qiu, N. Vaswani, B. Lois and L. Hogben, Recursive Robust PCA or Recursive Sparse Recovery in Large but Structured Noise, IEEE Trans. IT, 2014.

ReProCS correctness result [Lois,Vaswani, arXiV:1409.3959],[Qiu,Vaswani,Lois,Hogben,T-IT'14][6]

For most videos (i.e. w.p. at least $1 - n^{-10}$),

- the region occupied by the foreground objects (support of x_t) is exactly recovered at all times, and
- foreground and background images are accurately recovered at all times ($\|x_t - \hat{x}_t\|_2 = \|\ell_t - \hat{\ell}_t\|_2 \leq b$)
- the background subspace recovery error decays to a small value within a short delay of a subspace change time,

if

[6] B. Lois and N. Vaswani, A Correctness Result for Online Robust PCA, arXiV:1409.3959.
C. Qiu, N. Vaswani, B. Lois and L. Hogben, Recursive Robust PCA or Recursive Sparse Recovery in Large but Structured Noise, IEEE Trans. IT, 2014.

- an initial background-only training sequence is available (to get an accurate initial subspace estimate)

- an initial background-only training sequence is available (to get an accurate initial subspace estimate)
- the background images change slowly (ℓ_t lies in a slowly changing low-dimensional subspace)

- an initial background-only training sequence is available (to get an accurate initial subspace estimate)
- the background images change slowly (ℓ_t lies in a slowly changing low-dimensional subspace)
- background changes (w.r.t. a mean background image) are dense,

- an initial background-only training sequence is available (to get an accurate initial subspace estimate)
- the background images change slowly (ℓ_t lies in a slowly changing low-dimensional subspace)
- background changes (w.r.t. a mean background image) are dense,
- there is some motion of the foreground objects at least once every so often (there is some change in the support of x_t's)

Details follow in the next few slides . . .

ReProCS correctness result: Support change - examples

1. *(random motion)* all support sets mutually disjoint
   - this satisfies our model as long as $s \in O(\frac{n}{\log n})$

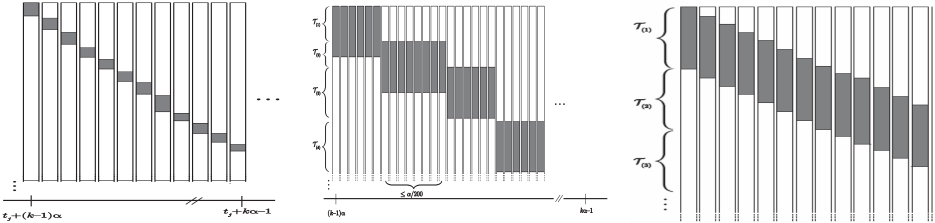
ReProCS correctness result: Support change - examples

1. *(random motion)* all support sets mutually disjoint
   - this satisfies our model as long as $s \in O(\frac{n}{\log n})$
2. *(infrequent motion)* a 1D object of length s that moves at least once every β frames; and, when it moves, it moves down by at least s/ϱ pixels
   - and by no more than $b_2 s$ indices
   - this satisfies our model as long as $s \in O(\frac{n}{\log n})$ and $\varrho^2 \beta \leq 0.01\alpha$

ReProCS correctness result: Support change - examples

1. *(random motion)* all support sets mutually disjoint
   - this satisfies our model as long as $s \in O(\frac{n}{\log n})$
2. *(infrequent motion)* a 1D object of length s that moves at least once every β frames; and, when it moves, it moves down by at least s/ϱ pixels
   - and by no more than $b_2 s$ indices
   - this satisfies our model as long as $s \in O(\frac{n}{\log n})$ and $\varrho^2 \beta \leq 0.01\alpha$
3. *(slow motion)* an object of length s moves down by at least one pixel in every frame
   - this satisfies our model as long as $s \in O(\log n)$

ReProCS correctness result: Support change - examples



(a) disjoint supports (b) infrequent motion (c) slow moving

Figure : In any of these we could have randomly selected pixels (need not be a block) at a given time and also random ordering across time

ReProCS correctness result: Subspace change model

ℓ_t's are zero mean, bounded and mutually independent r.v.'s with covariance matrix Σ_t that is low-rank and "slowly changing"

- $\Sigma_t \overset{EVD}{=} P_t \Lambda_t P_t'$ where $P_t = P_j$ for $t \in [t_j, t_{j+1} - 1]$, $j = 1, 2, \ldots J$
- P_j is a tall $n \times r_j$ basis matrix that changes as

$$P_j = [P_{j-1} \setminus P_{j,\text{old}}, \ \ P_{j,\text{new}}]$$

- "slow change": $\lambda_{\text{new}}^+(d) := \max_{t \in [t_j, t_j + d]} \lambda_{\max}(\Lambda_{t,\text{new}})$ is small and $t_{j+1} - t_j$ is large

ReProCS correctness result: Subspace change model

ℓ_t's are zero mean, bounded and mutually independent r.v.'s with covariance matrix Σ_t that is low-rank and "slowly changing"

- $\Sigma_t \overset{EVD}{=} P_t \Lambda_t P_t'$ where $P_t = P_j$ for $t \in [t_j, t_{j+1} - 1]$, $j = 1, 2, \dots J$
- P_j is a tall $n \times r_j$ basis matrix that changes as

$$P_j = [P_{j-1} \setminus P_{j,\text{old}}, \ \ P_{j,\text{new}}]$$

- "slow change": $\lambda^+_{\text{new}}(d) := \max_{t \in [t_j, t_j + d]} \lambda_{\max}(\Lambda_{t,\text{new}})$ is small and $t_{j+1} - t_j$ is large

Define

- $c := \max_j \text{rank}(P_{(j),\text{new}})$, $\gamma_{\text{new}}(d) := \max_{t \in [t_j, t_j + d]} \|a_{t,\text{new}}\|_\infty$
- $r := r_0 + Jc$, $\lambda^+ := \max_t \lambda_{\max}(\Lambda_t)$, $\gamma := \max_t \|a_t\|_\infty$

Theorem

Consider ReProCS. Pick a $\zeta \leq \min\left(\frac{10^{-4}\lambda_0^-}{(r_0+Jc)^2\lambda^+}, \frac{1}{(r_0+Jc)^3\gamma^2}\right)$. *If ReProCS algorithm parameters* α, K, ξ, ω *are set appropriately, and if*

Theorem

Consider ReProCS. Pick a $\zeta \leq \min\left(\frac{10^{-4}\lambda_0^-}{(r_0+Jc)^2\lambda^+}, \frac{1}{(r_0+Jc)^3\gamma^2}\right)$. *If ReProCS algorithm parameters* α, K, ξ, ω *are set appropriately, and if*

1. *initial subspace accurately estimated:* $\|(I - \hat{P}_0\hat{P}_0')P_0\|_2 \leq r_0\zeta$
2. *"slow subspace change" holds:*
   - *projection of* ℓ_t *along new direc's small for first* d *frames after* t_j*: for a* $d \geq (K+2)\alpha$, $\lambda_{new}^+(d) \leq 3\lambda_0^-$ *and* $\gamma_{new}(d) \leq 0.05 x_{\min}$
   - *and delay between change times is large:* $(t_{j+1} - t_j) > d$,

Theorem

Consider ReProCS. Pick a $\zeta \leq \min\left(\frac{10^{-4}\lambda_0^-}{(r_0+Jc)^2\lambda^+}, \frac{1}{(r_0+Jc)^3\gamma^2}\right)$. *If ReProCS algorithm parameters* α, K, ξ, ω *are set appropriately, and if*

1. *initial subspace accurately estimated:* $\|(I - \hat{P}_0\hat{P}_0')P_0\|_2 \leq r_0\zeta$
2. *"slow subspace change" holds:*
   - *projection of* ℓ_t *along new direc's small for first* d *frames after* t_j*: for a* $d \geq (K+2)\alpha$, $\lambda_{new}^+(d) \leq 3\lambda_0^-$ *and* $\gamma_{new}(d) \leq 0.05 x_{\min}$
   - *and delay between change times is large:* $(t_{j+1} - t_j) > d$,
3. *subspace basis matrices are dense enough:*

$$(2s)\max_i \|(P_{j,new})_{i,:}\|_2^2 \leq 0.0004 \text{ and } (2s)\max_i \|(P_J)_{i,:}\|_2 \leq 0.09$$

Theorem

Consider ReProCS. Pick a $\zeta \leq \min\left(\frac{10^{-4}\lambda_0^-}{(r_0+Jc)^2\lambda^+}, \frac{1}{(r_0+Jc)^3\gamma^2}\right)$. *If ReProCS algorithm parameters* α, K, ξ, ω *are set appropriately, and if*

1. *initial subspace accurately estimated:* $\|(I - \hat{P}_0\hat{P}_0')P_0\|_2 \leq r_0\zeta$
2. *"slow subspace change" holds:*
   - *projection of* ℓ_t *along new direc's small for first* d *frames after* t_j*: for a* $d \geq (K+2)\alpha$, $\lambda_{new}^+(d) \leq 3\lambda_0^-$ *and* $\gamma_{new}(d) \leq 0.05 x_{\min}$
   - *and delay between change times is large:* $(t_{j+1} - t_j) > d$,
3. *subspace basis matrices are dense enough:*

   $$(2s)\max_i \|(P_{j,new})_{i,:}\|_2^2 \leq 0.0004 \text{ and } (2s)\max_i \|(P_J)_{i,:}\|_2 \leq 0.09$$

4. *support of* x_t *has size smaller than* s *and changes enough,*
   - *e.g., moves down by at least* $s/10$ *pixels at least once every* $\alpha/500$ *frames,*

then, with probability at least $1 - n^{-10}$,

1. *support*(x_t) *is exactly recovered at all times,*
2. $SE_t := \|(I - \hat{P}_t \hat{P}_t')P_t\|_2$ *reduces to* $(r + c)\zeta$ *within* $(K + 2)\alpha$ *frames after* t_j,
3. $\|\ell_t - \hat{\ell}_t\|_2 = \|x_t - \hat{x}_t\|_2 \leq b \ll \|x_t\|_2$

Notice: no bound needed on λ^+ *or on* γ*: the result allows large but structured* ℓ_t

Details:

- *B. Lois and N. Vaswani, A Correctness Result for Online Robust PCA, ICASSP 2015, arXiV:1409.3959.*
- *C. Qiu, N. Vaswani, B. Lois and L. Hogben, Recursive Robust PCA or Recursive Sparse Recovery in Large but Structured Noise, IEEE Trans. IT, 2014.*

Discussion: Contributions

- To our knowledge, first correctness result for online robust PCA
  - or online sparse + low-rank recovery / online sparse recovery in large but structured noise
  - online algorithm: faster; less storage needed: only $O(n \log n)$ instead of $O(nt_{\max})$

Discussion: Contributions

- To our knowledge, first correctness result for online robust PCA
  - or online sparse + low-rank recovery / online sparse recovery in large but structured noise
  - online algorithm: faster; less storage needed: only $O(n \log n)$ instead of $O(nt_{\max})$
- Allows significantly more correlated support change than PCP
  - ReProCS allows the fraction of nonzeros per row of X to be $O(1)$;
  - PCP only allows this to be $O(\frac{1}{\text{rank}(L)})$ [Hsu et al'2011] or needs uniformly random support [Candes et al]

Discussion: Contributions

- To our knowledge, first correctness result for online robust PCA
  - or online sparse + low-rank recovery / online sparse recovery in large but structured noise
  - online algorithm: faster; less storage needed: only $O(n \log n)$ instead of $O(nt_{\max})$
- Allows significantly more correlated support change than PCP
  - ReProCS allows the fraction of nonzeros per row of X to be $O(1)$;
  - PCP only allows this to be $O(\frac{1}{\text{rank}(L)})$ [Hsu et al'2011] or needs uniformly random support [Candes et al]
- New proof techniques needed: useful for various other problems
  - almost all existing robust PCA results are for batch approaches
  - previous PCA results require $e_t := \hat{\ell}_t - \ell_t$ uncorrelated w/ ℓ_t

Discussion: Limitations

- Needs knowledge of bounds on γ_{new} and c to set algorithm parameters

Discussion: Limitations

- Needs knowledge of bounds on γ_{new} and c to set algorithm parameters
- Needs a tighter bound on rank and sparsity compared to PCP
  - let $s_{\text{mat}} := |\text{support}(X)|$ and $r_{\text{mat}} := \text{rank}(L)$
  - we allow $s_{\text{mat}} \in O(\frac{nt_{\max}}{\log n})$ and $r_{\text{mat}} \in O(\log n)$
  - PCP allows $s_{\text{mat}} \in O(nt_{\max})$ and $r_{\text{mat}} \in O(\frac{n}{\log^2 n})$

Discussion: Limitations

- Needs knowledge of bounds on γ_{new} and c to set algorithm parameters
- Needs a tighter bound on rank and sparsity compared to PCP
  - let $s_{\text{mat}} := |\text{support}(X)|$ and $r_{\text{mat}} := \text{rank}(L)$
  - we allow $s_{\text{mat}} \in O(\frac{nt_{\max}}{\log n})$ and $r_{\text{mat}} \in O(\log n)$
  - PCP allows $s_{\text{mat}} \in O(nt_{\max})$ and $r_{\text{mat}} \in O(\frac{n}{\log^2 n})$
  - result for ReProCS-deletion relaxes above (ongoing)

Discussion: Limitations

- Needs knowledge of bounds on γ_{new} and c to set algorithm parameters
- Needs a tighter bound on rank and sparsity compared to PCP
  - let $s_{\text{mat}} := |\text{support}(X)|$ and $r_{\text{mat}} := \text{rank}(L)$
  - we allow $s_{\text{mat}} \in O(\frac{nt_{\max}}{\log n})$ and $r_{\text{mat}} \in O(\log n)$
  - PCP allows $s_{\text{mat}} \in O(nt_{\max})$ and $r_{\text{mat}} \in O(\frac{n}{\log^2 n})$
  - result for ReProCS-deletion relaxes above (ongoing)
- Needs
  - initial subspace knowledge and slow subspace change
    - both are usually practically valid
  - zero-mean & mutually independent assump. on ℓ_t's over t
    - models independent random variations around a fixed bg mean
    - can replace it by a more practical AR model (ongoing)

Discussion: Limitations

- Needs knowledge of bounds on γ_{new} and c to set algorithm parameters
- Needs a tighter bound on rank and sparsity compared to PCP
  - let $s_{\text{mat}} := |\text{support}(X)|$ and $r_{\text{mat}} := \text{rank}(L)$
  - we allow $s_{\text{mat}} \in O(\frac{nt_{\max}}{\log n})$ and $r_{\text{mat}} \in O(\log n)$
  - PCP allows $s_{\text{mat}} \in O(nt_{\max})$ and $r_{\text{mat}} \in O(\frac{n}{\log^2 n})$
  - result for ReProCS-deletion relaxes above (ongoing)
- Needs
  - initial subspace knowledge and slow subspace change
    - both are usually practically valid
  - zero-mean & mutually independent assump. on ℓ_t's over t
    - models independent random variations around a fixed bg mean
    - can replace it by a more practical AR model (ongoing)
- Only ensures accurate recovery of x_t, ℓ_t, not exact

Some Generalizations

- Direct application to online matrix completion
- Easy extension to $y_t = Ax_t + B\ell_t$

Some Generalizations

- Direct application to online matrix completion
- Easy extension to $y_t = Ax_t + B\ell_t$
- Relax independence assumption on ℓ_t's, replace by AR model (ongoing) – almost exactly same result

Some Generalizations

- Direct application to online matrix completion
- Easy extension to $y_t = Ax_t + B\ell_t$
- Relax independence assumption on ℓ_t's, replace by AR model (ongoing) – almost exactly same result
- Result for ReProCS-deletion – ReProCS that also deletes direc's (ongoing):
  - needs an extra clustering assumption on the eigenvalues for a certain period of time after subspace change has stabilized;
  - but relaxes denseness requirement and so allows $r_{\text{mat}} \in O(n)$ instead of $r_{\text{mat}} \in O(\log n)$

Online Matrix Completion

- Can provide a provably accurate solution for online matrix completion; that also allows highly correlated set of unknown entries
  - but requires slow subspace change and initial subspace knowledge
- Low-rank matrix completion is a special case w/ known $T_t = \text{support}(x_t)$
  - in MC: T_t is the set of unknown entries of ℓ_t at time t
- ReProCS for online matrix completion:
  - Assume: accurate initial subspace knowledge, $\hat{P}_0$.
  - Compute $\Phi_t := (I - \hat{P}_{t-1}\hat{P}_{t-1}')$
  - Given T_t, get an estimate of ℓ_t as

$$\hat{\ell}_t = (I - I_{T_t}(\Phi_t)_{T_t}{}^{\dagger}\Phi_t)y_t$$

  - Use projection-PCA as before to update the subspace estimate

ReProCS algorithm - recap [Qiu,Vaswani,Allerton'10,Allerton'11][7]

Initialize: given $\hat{P}_0$ with range$(\hat{P}_0) \approx$ range$([\ell_1, \ell_2, \dots \ell_{t_0}])$
For $t > t_0$,

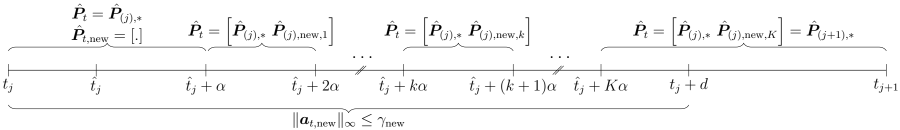
- Projection: compute $\tilde{y}_t := \Phi_t y_t$, where $\Phi_t := I - \hat{P}_{t-1}\hat{P}_{t-1}'$
  - then $\tilde{y}_t = \Phi_t x_t + \beta_t$, $\beta_t := \Phi_t \ell_t$ is small "noise"
- Noisy Sparse Recovery: ℓ_1 min + support estimate + LS: get $\hat{x}_t$
  - $\hat{x}_{t,cs} = \arg\min_x \|x\|_1$ s.t. $\|\tilde{y}_t - \Phi_t x\|_2 \leq \xi$
  - $\hat{\mathcal{T}}_t = \{i : |(\hat{x}_{t,cs})_i| > \omega\}$
  - $\hat{x}_t = I_{\hat{\mathcal{T}}_t}({A_{\hat{\mathcal{T}}_t}}' A_{\hat{\mathcal{T}}_t})^{-1} {A_{\hat{\mathcal{T}}_t}}' y_t$
- Get $\hat{\ell}_t = y_t - \hat{x}_t$
- Subspace update: update $\hat{P}_t$ every α frames by projection-PCA

---

[7] C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010
C. Qiu and N. Vaswani, Recursive Sparse Recovery in Large but Correlated Noise, Allerton 2011

ReProCS algorithm: projection PCA

Assume $t_{j+1} - t_j > (K+2)\alpha$; recall: t_j: subspace change times



let $\hat{P}_{j,*} := \hat{P}_{j-1}$ be an (accurate) estimate of the previous subspace

at $t = \hat{t}_j + k\alpha,\ k = 1, 2, \dots K$,

- $\hat{P}_{j,\text{new},k} \leftarrow SVD\left((I - \hat{P}_{j,*}\hat{P}'_{j,*})[\hat{\ell}_{\hat{t}_j+(k-1)\alpha+1}, \dots \hat{\ell}_{\hat{t}_j+k\alpha}], thresh\right)$
- update $\hat{P}_t = [\hat{P}_{j,*}, \hat{P}_{j,\text{new},k}]$

Proof idea: Why projection PCA works?

- Before the first proj-PCA, i.e. for $t \in [t_j, \hat{t}_j + \alpha]$,
  - $P_t = [P_*, P_{\text{new}}]$, $\hat{P}_{t-1} = [\hat{P}_*] \Rightarrow \beta_t$ (noise seen by sparse rec step) and hence $e_t = \hat{x}_t - x_t = \ell_t - \hat{\ell}_t$ is largest
  - e_t still not too large due to slow subspace change; and e_t is sparse and supported on $\mathcal{T}_t$
  - at $t = \hat{t}_j + \alpha$, get $\hat{P}_{\text{new},1}$: estimate is good because of above: $\text{SE}(P_{\text{new}}, \hat{P}_{\text{new},1}) := \|(I - \hat{P}_{\text{new},1}\hat{P}_{\text{new},1}{}')P_{\text{new}}\|_2 < 0.6$

Proof idea: Why projection PCA works?

- Before the first proj-PCA, i.e. for $t \in [t_j, \hat{t}_j + \alpha]$,
  - $P_t = [P_*, P_{\text{new}}]$, $\hat{P}_{t-1} = [\hat{P}_*] \Rightarrow \beta_t$ (noise seen by sparse rec step) and hence $e_t = \hat{x}_t - x_t = \ell_t - \hat{\ell}_t$ is largest
  - e_t still not too large due to slow subspace change; and e_t is sparse and supported on $\mathcal{T}_t$
  - at $t = \hat{t}_j + \alpha$, get $\hat{P}_{\text{new},1}$: estimate is good because of above: $\text{SE}(P_{\text{new}}, \hat{P}_{\text{new},1}) := \|(I - \hat{P}_{\text{new},1}\hat{P}_{\text{new},1}{}')P_{\text{new}}\|_2 < 0.6$
- For $t \in [\hat{t}_j + \alpha + 1, \hat{t}_j + 2\alpha]$,
  - $P_t = [P_*, P_{\text{new}}]$, $\hat{P}_{t-1} = [\hat{P}_*, \hat{P}_{\text{new},1}] \Rightarrow \beta_t$ and hence e_t smaller; and e_t is sparse and supported on $\mathcal{T}_t$
  - at $t = \hat{t}_j + 2\alpha$, get $\hat{P}_{\text{new},2}$; estimate better because of above
- Continuing this way, show $\text{SE}(P_{\text{new}}, \hat{P}_{\text{new},k}) < 0.6^k + 0.4c\zeta$; pick K so $\text{SE}(P_{\text{new}}, \hat{P}_{\text{new},K}) < c\zeta$

Proof Outline: k-th projection-PCA interval

Conditioned on accurate recovery so far,

- slow subspace change, denseness assumption, appropriate support threshold and LS ensure that $e_t := x_t - \hat{x}_t = \hat{\ell}_t - \ell_t$ satisfies

$$e_t = I_{\mathcal{T}_t}[\Phi_{\mathcal{T}_t}{}'\Phi_{\mathcal{T}_t}]^{-1} I_{\mathcal{T}_t}{}'\Phi \ell_t \text{ where } \Phi := I - \hat{P}_{t-1}\hat{P}_{t-1}{}'$$

  and

$$\|[\Phi_{\mathcal{T}_t}{}'\Phi_{\mathcal{T}_t}]^{-1}\|_2 \leq 1.2$$

- by $\sin\theta$ theorem [Davis,Kahan,1970],

$$\text{SE}(\hat{P}_{\text{new},k}, P_{\text{new}}) \lesssim \frac{\|\text{perturbation}\|_2}{\lambda_{\text{new}}^- - \|\text{perturbation}\|_2}$$

$$\|\text{perturbation}\|_2 \lesssim 2\Big\|\frac{1}{\alpha}\sum_t (I - \hat{P}_*\hat{P}_*{}')\ell_t e_t'\Big\|_2 + \Big\|\frac{1}{\alpha}\sum_t e_t e_t'\Big\|_2$$

- use matrix Hoeffding ineq [Tropp,2012] to bound these terms w.h.p.

Proof Outline: k-th projection-PCA interval – 2

Conditioned on accurate recovery so far,

- the dominant perturbation term

$$\mathrm{dom} := \mathbb{E}\left[\frac{1}{\alpha}\sum_{t=\hat{t}_j+(k-1)\alpha}^{\hat{t}_j+k\alpha}(I-\hat{P}_*\hat{P}_*{}')\ell_t e_t'\right] \approx \frac{1}{\alpha}\sum_t A_t B_t'$$

where $A_t := P_{\text{new}}\Lambda_{t,\text{new}}P_{\text{new}}'$ and $B_t := I_{\mathcal{T}_t}[\Phi_{\mathcal{T}_t}{}'\Phi_{\mathcal{T}_t}]^{-1}I_{\mathcal{T}_t}{}'$

- use slow subspace change to get

$$\|\frac{1}{\alpha}\sum_t A_t A_t'\|_2 \le \max_t \|A_t\|_2^2 \le \lambda_{\text{new}}^+(d)^2 \le 9\lambda_0^{-2}$$

- use model on $\mathcal{T}_t$ to show that

$$\|\frac{1}{\alpha}\sum_t B_t B_t'\|_2 = \|\frac{1}{\alpha}\sum_t I_{\mathcal{T}_t}[\Phi_{\mathcal{T}_t}{}'\Phi_{\mathcal{T}_t}]^{-2}I_{\mathcal{T}_t}{}'\|_2 \le \frac{1}{\alpha}1.2^2\varrho^2\beta \le 0.02$$

Proof Outline: k-th projection-PCA interval – 2

Conditioned on accurate recovery so far,

- the dominant perturbation term

$$\mathrm{dom} := \mathbb{E}\left[\frac{1}{\alpha}\sum_{t=\hat{t}_j+(k-1)\alpha}^{\hat{t}_j+k\alpha}(I-\hat{P}_*\hat{P}_*{}')\ell_t e_t'\right] \approx \frac{1}{\alpha}\sum_t A_t B_t'$$

where $A_t := P_{\mathrm{new}}\Lambda_{t,\mathrm{new}}P_{\mathrm{new}}'$ and $B_t := I_{\mathcal{T}_t}[\Phi_{\mathcal{T}_t}{}'\Phi_{\mathcal{T}_t}]^{-1}I_{\mathcal{T}_t}{}'$

- use slow subspace change to get

$$\|\frac{1}{\alpha}\sum_t A_t A_t'\|_2 \le \max_t \|A_t\|_2^2 \le \lambda_{\mathrm{new}}^{+}(d)^2 \le 9\lambda_0^{-2}$$

- use model on $\mathcal{T}_t$ to show that

$$\|\frac{1}{\alpha}\sum_t B_t B_t'\|_2 = \|\frac{1}{\alpha}\sum_t I_{\mathcal{T}_t}[\Phi_{\mathcal{T}_t}{}'\Phi_{\mathcal{T}_t}]^{-2}I_{\mathcal{T}_t}{}'\|_2 \le \frac{1}{\alpha}1.2^2\varrho^2\beta \le 0.02$$

- use Cauchy-Schwartz to get $\|\mathrm{dom}\|_2 \lesssim \sqrt{0.02}\cdot 3\lambda_0^-$

Proof Outline: Overall idea

- Define subspace error, $\text{SE}(P, \hat{P}) := \|(I - \hat{P}\hat{P}')P\|_2$.
- Start with $\text{SE}(P_{j-1}, \hat{P}_{j-1}) \leq r_{j-1}\zeta \ll 1$ at $t = t_j - 1$.
  1. First show that $t_j \leq \hat{t}_j \leq t_j + 2\alpha$
  2. Analyze projected sparse recovery for $t \in [\hat{t}_j, \hat{t}_j + \alpha)$
  3. Analyze proj-PCA at $t = \hat{t}_j + \alpha$: $\text{SE}(P_{j,\text{new}}, \hat{P}_{j,\text{new},1}) \leq 0.6$
  4. Repeat for each of the K projection-PCA intervals: show that
  $$\text{SE}(P_{j,\text{new}}, \hat{P}_{j,\text{new},k}) \leq 0.6^k + 0.4c\zeta$$
  5. Pick K s.t. $0.6^K + 0.4c\zeta \leq c\zeta$. Set $\hat{P}_j = [\hat{P}_{(j-1)}, \hat{P}_{j,\text{new},K}]$
- Thus, at $t = \hat{t}_j + K\alpha - 1$,

  $\text{SE}(P_j, \hat{P}_j) \leq \text{SE}(P_{j-1}, \hat{P}_{j-1}) + \text{SE}(P_{j,\text{new}}, \hat{P}_{j,\text{new},K}) \leq r_{j-1}\zeta + c\zeta = r_j\zeta$
- $t_{j+1} - t_j > (K+2)\alpha$ implies $\text{SE}(P_j, \hat{P}_j) \leq r_j\zeta$ at $t = t_{j+1} - 1$

Experiments [Guo,Qiu,Vaswani,TSP'14][8]

1. Real background simulated foreground: background of moving lake water video with a simulated moving rectangular object overlaid on it; object intensity similar to background intensity and object moving slowly (making it a difficult seq)

2. Real videos

[8] H. Guo, C. Qiu, N. Vaswani, An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans. SP, Aug 2014

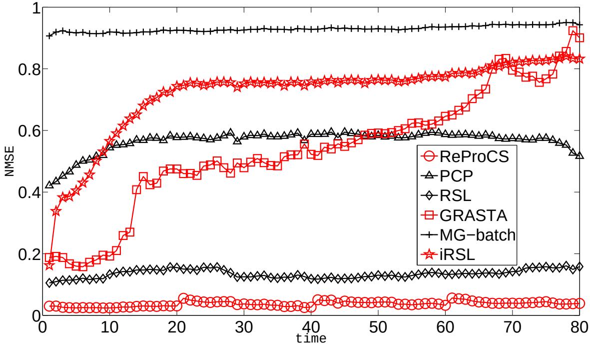


Figure : Recovery error (Monte Carlo over 100 realiz's). Black: batch methods, Red: online methods, Red Circles: ReProCS

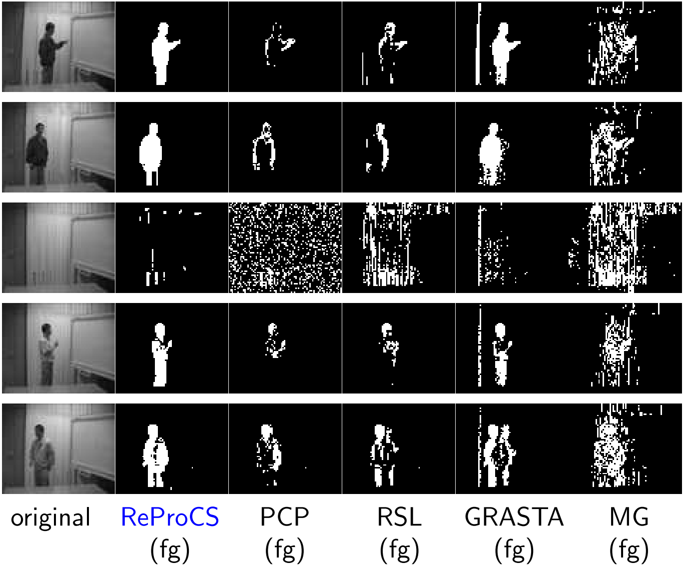


Figure : Online: ReProCS (proposed method) and GRASTA, Batch:

original | ReProCS (bg) | PCP (bg) | RSL (bg) | GRASTA (bg) | MG (bg)

Figure : Background layer recovery at $t = t_{train} + 60, 120, 199, 475, 1148$.

Algorithm parameters

Recall that $\zeta \leq \min(\frac{10^{-4}}{(r_0+Jc)^2 f}, \frac{1}{(r_0+Jc)^3\gamma_*^2})$.

- $\xi = \sqrt{c}\gamma_{\text{new}} + \sqrt{\zeta}(\sqrt{r_0 + Jc} + \sqrt{c})$;
- ω satisfies $7\xi \leq \omega \leq x_{\min} - 7\xi$;
- $K = \left\lceil \frac{\log(0.16c\zeta)}{\log(0.4)} \right\rceil$;
- $\alpha = C(\log(6KJ) + 11\log(n)), \; C \geq C_{add} := 20^2 \cdot 8 \cdot 96^2 \frac{(1.2\xi)^4}{(c\zeta\lambda^-)^2}$
- If we assume that min and max eigenvalues are seen in the training data, then can estimate λ^-, λ^+, γ_* from training data

Summary

- To the best of our knowledge, this is the first correctness result for online sparse + low-rank recovery
  - equivalently also for online robust PCA / recursive sparse recovery in large but structured noise
- Advantages
  - online algorithm: faster; less storage needed; removes a key limitation of PCP: allows more correlated support change
- New proof techniques needed to obtain our results
  - almost all existing robust PCA results are for batch approaches
  - previous finite sample PCA results are not useful: assume $e_t := \hat{\ell}_t - \ell_t$ is uncorrelated with ℓ_t

References

1. B. Lois and N. Vaswani, "A Correctness Result for Online Robust PCA", ICASSP 2015, arXiv:1409.3959[cs.IT]
2. C. Qiu, N. Vaswani, B. Lois and L. Hogben, "Recursive Robust PCA or Recursive Sparse Recovery in Large but Structured Noise", IEEE Trans. IT, August 2014
3. H. Guo, C. Qiu, N. Vaswani, "An Online Algorithm for Separating Sparse and Low-Dimensional Signal Sequences From Their Sum", IEEE Trans. Sig. Proc, August 2014
4. C. Qiu and N. Vaswani, Real-time Robust Principal Components' Pursuit, Allerton, 2010