Real-time Principal Components’ Pursuit

Chenlu Qiu

Department of Electrical and Computer Engineering
Iowa State University
Web: http://www.ece.iastate.edu/~chenlu
Robust Principal Component Analysis

- Principal Component Analysis (PCA)
  - Find the “principal components’ space” with the smallest dimension that spans a given dataset.
  - Optimal in MSE sense.
  - Sensitive to outliers and corruptions.

- Solutions:
  - A lot of existing work on robustifying PCA, most of which
    - first detect the corrupted points and then
    - either fill them with some heuristics, or just remove them.
  - Recent work: Robust Principal Component Analysis [E. Candes, et.al], [P. A. Parrilo et.al, 09]
Robust Principal Component Analysis [E. Candes, et.al] [P. A. Parrilo et.al, 09]

Given a data matrix $M$ composed of low rank component $L$ and sparse component $S$, recover $L$ and $S$ from $M$.

- Nonzero entries of $S$ can have arbitrary large magnitude.

This can be solved by Principal Component Pursuit (PCP)\(^1\) as

$$
\min_{L,S} \|L\|_* + \lambda \|S\|_1 \quad \text{s.t. } L + S = M
$$

(1)

with high probability provided that:

(i) the singular vectors of $L$ are spread out (not sparse)\(\implies L\) is not sparse

(ii) the support of $S$ are uniformly random \(\implies S\) is not low rank

(iii) rank($L$) and $|\text{supp}(S)|$ (size of the support of $S$) are both sufficiently small

---

\(^1\)Let $\|L\|_*$ denote the nuclear norm of $L$, i.e. sum of singular values of $L$; and let $\|S\|_1$ denotes the $\ell_1$ norm of $S$ seen as a long vector
Robust Principal Component Analysis (RPCA)

- Applications: video surveillance, face recognition, etc.

For e.g., given a sequence of surveillance video frames,
- stack each image frame as a column vector of the data matrix $M$
- the background variations lying on a low dimensional subspace is modeled as low rank component $L$;
- the “moving objects” are modeled as the sparse part $S$. 
RPCA in video surveillance

- $L$ lies on a low dimensional subspace.
- The support of $S_t$ is uniformly random with $|\text{supp}(S)| = 32$.
- RPCA distinguishes $L$ and $S$ correctly with $\frac{\|S - \hat{S}\|_F^2}{\|S\|_F^2} = 9.8 \times 10^{-7}$. $^2$

---

$^2$ Let $\|X\|_F$ denote the Frobenius norm of matrix $X$, $\|X\|_F^2 = \sum_i \sum_j (X_{i,j})^2$. 

Chenlu Qiu
RR-PCP
5/22
RPCA in video surveillance: limitations

- A offline method
  - Surveillance application usually requires online approach.
- Require the support of $S$ to be uniformly random.
  - Objects occupy a block of pixels and move as a block $\Rightarrow$ elements of $S_t$ are spatially correlated.
  - Objects move slowly and/or with approximately constant velocity $\Rightarrow$ $S_t$ and $S_{t+1}$ are time correlated.
- Require rank($L$) and $|\text{supp}(S)|$ to be sufficiently small.
RPCA when objects move in a correlated fashion

- Eight moving objects in $S_t$.
- Each object occupies a $2 \times 2$ nonzero pixel block.
- $|\text{supp}(S)| = 32$
- Each object moves one pixel step towards top/bottom/left/right w.p. 0.05 and stays static w.p. 0.8.
- RPCA still works with $\frac{\|S - \hat{S}\|_F^2}{\|S\|_F^2} = 2.4 \times 10^{-4}$. 

Chenlu Qiu
RPCA fails when objects move in a heavily correlated fashion

- Two moving objects in $S_t$.
- Each object occupies a $4 \times 4$ nonzero pixel block.
- $|\text{supp}(S)| = 32$
- Each object moves one pixel step towards top/bottom/left/right w.p. 0.05 and stays static w.p. 0.8.
- RPCA fails with $\frac{\|S - \hat{S}\|^2}{\|S\|^2}$ above 38%.

Chenlu Qiu
Problem Formulation and Signal Model

- Our problem: given $M_t$, recover $L_t$ and $S_t$ sequentially.
  - Do this even if the support of $S_t$ are not randomly distributed.
Matrix $L$ has low rank $\iff L_t = Ux_t$ for some unknown orthonormal matrix $U$ and a sparse vector $x_t$.

$x_t$, and hence $L_t$, follows a piecewise stationary model with nonstationary transients when switching pieces.

- The sparse vector $x_t$ has piecewise constant support $N_t := \text{supp}(x_t)$.
- The changes of $N_t$ (additions and/or deletions) happens for every $d$ frames.
- For each piece, $x_t$ follows a stationary AR-1 model with parameter $0 < f < 1$, i.e.,
  \[
  x_t = f \, x_{t-1} + \nu_t \\
  x_t \sim \mathcal{N}(0, \Sigma), \; \nu_t \sim \mathcal{N}(0, (1 - f^2)\Sigma)
  \]
- When switching pieces, $x_t$ is nonstationary transient.
In $S_t$,

- There are several nonzero pixel blocks. All other pixels are zero.
- Each nonzero block can either be static w.p. $p$, or move one pixel step to left/right/top/bottom w.p. $(1 - p)/4$.

Therefore, the support of $S_t$ is spatial and time correlated.
Recall that

\[ M_t = L_t + S_t = [U \ I] \begin{bmatrix} x_t \\ S_t \end{bmatrix} \]

where \( L_t = Ux_t \) with \( U \) an unknown orthonormal matrix and \( x_t \) a sparse vector with support \( N_t \).

At first glance, a straight way is let \( A = [U \ I] \) and \( u = [x_t \ S_t]^T \) and solve

\[
\min \|u\|_1 \quad \text{s.t.} \quad M_t = Au
\]

This method is termed as Pursuit of Justices (PJ) [John Wright, et.al. 09], [J. N. Laska, et. al. 09].

However, the above method can not be used because \( U \) is unknown.
Other related works include

- **Decoding by Linear Programming** [E. Candes, T. Tao’ 2005].
- **Robust Linear Regression** [Y. Jin, B.Rao’ 2010].
- **Bayesian Sparse Robust Regression** [K. Mitra, et. al 2010].
- All above methods require $(U)_{N_t}$ known.
Main Idea

- Let $P_t := (U)_{N_t}$ be a submatrix of $U$, composed by the columns of $U$ indexed by $N_t$.
- Columns of $P_t$ spans the principal components' subspace of $L_t$.
- Using $\hat{P}_t$, an estimate of $P_t$, we have

$$L_t = \hat{P}_t \alpha_t + \hat{P}_{t,\perp} \beta_t,$$

therefore,

$$M_t = \hat{P}_t \alpha_t + \hat{P}_{t,\perp} \beta_t + S_t$$

with

- $\hat{P}_{t,\perp}$ an orthogonal complement of $\hat{P}_t$
- $\alpha_t = (\hat{P}_t)^T L_t$
- $\beta_t = (\hat{P}_{t,\perp})^T L_t$
Main Idea

Let $y_t := (\hat{P}_{t,\perp})^T M_t = (\hat{P}_{t,\perp})^T S_t + \beta_t$.

- Note that $\beta_t = (\hat{P}_{t,\perp})^T L_t = (\hat{P}_{t,\perp})^T U x_t$.
  Therefore, when $\text{span}(P_t) \subseteq \text{span}(\hat{P}_t)$, $\beta_t = 0$;
  when $\text{span}(P_t) \nsubseteq \text{span}(\hat{P}_t)$, $\beta_t \neq 0$.

- $\beta_t = (\hat{P}_{t,\perp})^T U x_t$ is the “noise” from resulting from the inaccuracies of $\hat{P}_t$, $x_t \sim N(0, \Sigma)$.

- Recall that $L_t$ follows a AR-1 model with parameter $f$, we can reduce the “noise” by using

$$\tilde{y}_t := (\hat{P}_{t,\perp})^T (M_t - f L_{t-1})$$

for which the “noise” is $\hat{P}_{t,\perp}^T U (x_t - fx_{t-1})$, $x_t - fx_{t-1} \sim N(0, (1 - f^2)\Sigma)$.
Stepwise Algorithm

- We estimate \( \hat{S}_t \) by

\[
\min \|s\|_1 \text{ s.t. } \|\hat{P}_{t, \perp}^T (M_t - s - f\hat{L}_{t-1})\|_2^2 \leq \epsilon
\]  

(2)

- We do support thresholding followed by LS estimate to reduce the error [The Dantzig selector, 2005]:

\[
\hat{T}_t = \{i : (\hat{S}_t)_i \geq \gamma\}
\]

\[
(\hat{S}_t)_{\hat{T}_t} = ((\hat{P}_{t, \perp}^T)_{\hat{T}_t})^\dagger (y_t - f\hat{P}_{t, \perp}^T \hat{L}_{t-1}), (\hat{S}_t)_{\hat{T}_c} = 0
\]

- Let \( \hat{L}_t = M_t - \hat{S}_t \).

- Update \( \hat{P}_{t, \perp} \) using the past sequence's estimate \( \hat{L}_t \) when \( \| (\hat{P}_{t, \perp})^T \hat{L}_t \|_2^2 \) is large (with complexity \( O(m^2 r), \hat{P}_t \in \mathbb{R}^{m \times r} \)).
An overview of our method is given as

\[ \tilde{y}_t = (\hat{P}_{t,\perp})^T (M_t - f \hat{L}_{t-1}) \]

\[ \min \|s\|_1 \]
\[ \text{s.t.} \quad \|\tilde{y}_t - (\hat{P}_{t,\perp})^T s\|_2^2 \leq \epsilon \]
Recursive PCP: at each time $t$, do PCP using all available frames, i.e. solve (1) with $M = [M_1, \cdots, M_t]$.

Since the support of $S_t$ is spatial and time correlated, Recursive PCP does not work.

Our method, RR-PCP, can distinguish $L$ and $S$ very well.
Result Comparison

Normalized Squared Error against time is plotted below.

\[
\frac{\|S_t - \hat{S}_t\|^2}{\|S_t\|^2}
\]

\[
\frac{\|S_t - \hat{S}_t\|^2}{\|S_t\|^2}
\]
When there are five moving “objects”, RR-PCP can distinguish $L$ and $S$ very well.
Discussion

- **Summary of RR-PCP**
  - When a new image frame $M_t$ is available, reconstruct $S_t$ from its perpendicular projection $(\hat{P}_{t,\perp})^T(M_t - f\hat{L}_{t-1})$.
  - Update $\hat{P}_t$ and $\hat{P}_{t,\perp}$ every-so-often.
  - Support of $S_t$ need not to be random.

- **Limitation of RR-PCP**
  - Need a training sequence without sparse component to get an initial estimate $\hat{P}_0$
Future Work

- **Large Scale Data**
  - Memory and computational expensive to compute $\hat{P}_t, \perp$
  - Solution:
    \[
    \min \|s\|_1 \text{ s.t. } \|(I - \hat{P}_t \hat{P}_t^T)(M_t - s - f\hat{L}_{t-1})\|_2^2 \leq \epsilon
    \]

- **Incorporate motion prediction to improve further**
  - Using the motion model and the estimate of last frame, $\hat{S}_{t-1}$, predict the object(s)' location and get $T_{\text{pred}}$, an estimate of $\text{supp}(S_t)$, and solve modified-CS [N. Vaswani et.al. 2010] as
    \[
    \min_{s} \|s_{T_{\text{pred}}}^c\|_1 \text{ s.t. } \|\hat{P}_T^T (M_t - s - f\hat{L}_{t-1})\|_2^2 \leq \epsilon
    \]

- **Undersampled $M_t$?**