Algebraic, Sparse and Low Rank Subspace Clustering

René Vidal
Center for Imaging Science
Johns Hopkins University
High-Dimensional Data

- In many areas, we deal with high-dimensional data
  - Computer vision
  - Medical imaging
  - Medical robotics
  - Signal processing
  - Bioinformatics

The Language of Surgery

Modeling the skills of human expert surgeons to train a new generation of students. (more)
NUMBER OF PHOTOS TAKEN EACH YEAR

High-Dimensional Data in Computer Vision

Daily Number of Photos Uploaded & Shared on Select Platforms, 2005 – 2014YTD

- Flickr
- Snapchat
- Instagram
- Facebook
- WhatsApp (2013, 2014 only)

High-Dimensional Data in Computer Vision

- 140 billion images
- 350 million new photos/day

- 120 million videos
- 300 hours of video/minute

- 3.8 trillion of photographs
- 10% in the past 12 months

- 90% of the internet traffic will be video by the end of 2017

http://www.buzzfeed.com/hunterschwarz/how-many-photos-have-been-taken-ever-6zgv
Low-Dimensional Manifolds

- Face clustering and classification
- Lossy image representation
- Motion segmentation
- DT segmentation
- Video segmentation
Two Fundamental Tasks

• Clustering of data in low-dimensional manifolds

• Classification of data in low-dimensional manifolds
Talk Outline

• Introduction to Subspace Clustering

• Generalized Principal Component Analysis (GPCA)
  – Polynomial fitting and factorization

• Sparse Subspace Clustering (SSC)
  – Matrix of coefficients is sparse

• Low Rank Subspace Clustering (LRSC)
  – Matrix of coefficients is low-rank

• Applications:
  – Face clustering
  – Motion/video segmentation

Introduction to Subspace Clustering

René Vidal
Principal Component Analysis (PCA)

- Given a set of points lying in one subspace, identify
  - Geometric PCA: find a subspace $S$ passing through them
  - Statistical PCA: find projection directions that maximize the variance

- **Solution** (Beltrami’1873, Jordan’1874, Hotelling’33, Eckart-Householder-Young’36)

$$U \Sigma V^\top = [x_1 \ x_2 \ \cdots \ x_N] \in \mathbb{R}^{D \times N}$$

- **Applications:**
  - Signal/image processing, computer vision (eigenfaces), machine learning, genomics, neuroscience (multi-channel neural recordings)
Subspace Clustering Problem

- Given a set of points lying in multiple subspaces, identify
  - The number of subspaces and their dimensions
  - A basis for each subspace
  - The segmentation of the data points

- Challenges
  - Model selection
  - Nonconvex
  - Combinatorial

- More challenges
  - Noise
  - Outliers
  - Missing entries
Subspace Clustering Problem: Challenges

- Even more challenges
  - Angles between subspaces are small
  - Nearby points are in different subspaces

![Graphs showing percentage of subspace pairs and percentage of data points](image-url)
Prior Work: Iterative-Probabilistic Methods

• Approach
  – Given segmentation, estimate subspaces
  – Given subspaces, segment the data
  – Iterate till convergence

• Representative methods
  – **Mixtures of PPCA** (Tipping-Bishop ’99, Grubber-Weiss ’04, Kanatani ’04, Archambeau et al. ’08, Chen ’11)

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages / Open Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple, intuitive</td>
<td>Known number of subspaces and dimensions</td>
</tr>
<tr>
<td>Missing data</td>
<td>Sensitive to initialization and outliers</td>
</tr>
</tbody>
</table>
Prior Work: Algebraic-Geometric Methods

• **Approach**
  – Number of subspaces = degree of polynomial
  – Subspaces = factors of polynomial

• **Representative methods**
  – **Factorization** (Boult-Brown’91, Costeira-Kanade’98, Gear’98, Kanatani et al.’01, Wu et al.’01, Sekmen’13)
  – **GPCA** (Shizawa-Maze ’91, Vidal et al. ’03 ’04 ’05, Huang et al. ’05, Yang et al. ’05, Derksen ’07, Ma et al. ’08, Ozay et al. ‘10)

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages / Open Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Closed form</td>
<td>Complexity</td>
</tr>
<tr>
<td>Arbitrary dimensions</td>
<td>Sensitive to noise, outliers, missing entries</td>
</tr>
</tbody>
</table>
Prior Work: Spectral-Clustering Methods

• Approach
  – Data points = graph nodes
  – Pairwise similarity = edge weights
  – Segmentation = graph cut

• Representative methods
  – Local (Zelnik-Manor ’03, Yan-Pollefeys ’06, Fan-Wu ’06, Goh-Vidal ’07, Sekmen’12)
  – Global (Govindu ’05, Agarwal et al. ’05, Chen-Lerman ’08, Lauer-Schnorr ’09, Zhang et al. ’10)

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages / Open Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Efficient</td>
<td>Known number of subspaces and dimensions</td>
</tr>
<tr>
<td>Robust</td>
<td>Global methods are complex</td>
</tr>
</tbody>
</table>
Prior Work: Sparse and Low-Rank Methods

• Approach
  – Data are self-expressive
  – Global affinity by convex optimization

• Representative methods
  – Sparse Subspace Clustering (SSC) (Elhamifar-Vidal ’09 ’10 ’13, Candes ’12 ’13)
  – Low-Rank Subspace Clustering (LRSC) (Liu et al. ’10 ’13, Favaro-Vidal ’11 ’13)
  – Sparse + Low-Rank (Wang ’13)

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages / Open Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Efficient, Convex</td>
<td>Low-dimensional subspaces</td>
</tr>
<tr>
<td>Robust</td>
<td>Missing entries</td>
</tr>
</tbody>
</table>
Prior Work on Subspace Clustering

Subspace Clustering

Applications in motion segmentation and face clustering
Generalized Principal Component Analysis (GPCA)

René Vidal, Yi Ma and Shankar Sastry
GPCA: Representing One Subspace

- One plane
  \[ b^T x = b_1 x_1 + b_2 x_2 + b_3 x_3 = 0 \]

- One line
  \[ b_1^T x = b_1 x_1 + b_2 x_2 + b_3 x_3 = 0 \]
  \[ b_2^T x = b_4 x_1 + b_5 x_2 + b_6 x_3 = 0 \]

- One subspace can be represented with
  - Set of linear equations
  - Set of polynomials of degree 1
  \[ S = \{ x : B^T x = 0 \} \]
GPCA: Representing a Union of Subspaces

- One subspace

\[ b^T x = b_1 x_1 + b_2 x_2 + b_3 x_3 = 0 \]

- Two subspaces

\[ (b_1^T x = 0) \text{ or } (b_2^T x = 0) \]

\[ p_2(x) = (b_1^T x)(b_2^T x) = 0 \]

- A union of n subspaces can be represented with a set of homogeneous polynomials of degree n
GPCA: Representing \( n \) Subspaces

- Two planes \((b_1^T x = 0) \) or \((b_2^T x = 0)\)
  
  \[ p_2(x) = (b_1^T x)(b_2^T x) = 0 \]

- One plane and one line
  
  - Plane: \( S_1 = \{x : b^T x = 0\} \)
  
  - Line: \( S_2 = \{x : b_1^T x = b_2^T x = 0\} \)

  \[ S_1 \cup S_2 = \{x : (b^T x = 0) \text{ or } (b_1^T x = b_2^T x = 0)\} \]

  De Morgan’s rule

  \[ S_1 \cup S_2 = \{x : (b^T x)(b_1^T x) = 0 \text{ and } (b^T x)(b_2^T x) = 0\} \]

- A union of \( n \) subspaces can be represented with a set of homogeneous polynomials of degree \( n \)

Vidal, Ma, Piazzi, Sastry. A new GPCA Algorithm for Clustering Subspaces by Fitting, Differentiating and Dividing Polynomials, CVPR 04.

GPCA: Fitting Polynomials to Data Points

- Polynomials are linear in their coefficients
  \[(b_1^\top \mathbf{x})(b_2^\top \mathbf{x}) = c_1 x_1^2 + c_2 x_1 x_2 + c_3 x_2^2 = c^\top \nu_n(\mathbf{x}) = 0\]

- Coefficients can be computed linearly from the nullspace of the embedded data matrix
  - Solve using least squares
  - \(N = \#\text{data points}\)

\[
L_n c = \begin{bmatrix}
    \nu_n(\mathbf{x}_1)^\top \\
    \vdots \\
    \nu_n(\mathbf{x}_N)^\top
\end{bmatrix} c = 0
\]

- Number of subspaces can be found from rank of embedded data matrix
  \[n = \min\{i : L_i \text{ drops rank}\}\]
GPCA Algorithm by Polynomial Factorization

• Basis for each subspace

\[ c^T \nu_n(x) = (b_1^T x) \cdots (b_n^T x) \]

\[ c \in \mathbb{R}^{M_n} \]

\[ b_1 \quad b_2 \quad \cdots \quad b_n \]

• Polynomial Factorization Algorithm
  – Find roots of polynomial of degree n in one variable
  – Solve D-2 linear systems in n variables

• Problems
  – Computing roots may be sensitive to noise
  – The estimated polynomial may not perfectly factor with noisy data

To learn a mixture of subspaces we just need one positive example per class.
GPCA Algorithm Polynomial Differentiation

- With noise and outliers
  - Polynomials may not be a perfect union of subspaces

\[ b_1^T x = 0 \]
\[ p_n(x) = 0 \]
\[ b_2^T x = 0 \]
\[ y_1 \]
\[ y_2 \]

- Normals can estimated correctly by choosing points optimally

- Distance to closest subspace without knowing segmentation?

\[ \| x - \tilde{x} \| = \sqrt{\frac{|p_n(x)|}{\|Dp_n(x)\|}} + O(\|x - \tilde{x}\|^2) \]

Vidal, Ma, Piazzi, Sastry. A new GPCA Algorithm for Clustering Subspaces by Fitting, Differentiating and Dividing Polynomials, CVPR 04.
GPCA: Algorithm for Hyperplane Clustering

- Coefficients of the polynomial can be computed from null space of embedded data matrix
  - Solve using least squares
  - \( N = \#\text{data points} \)

\[
L_n c = \begin{bmatrix}
    \nu_n(x_1)^T \\
    \vdots \\
    \nu_n(x_N)^T
\end{bmatrix} c = 0
\]

- Number of subspaces can be computed from the rank of embedded data matrix

\[
n = \min\{i : \text{rank}(L_i) = M_i - 1\}
\]

- Normal to the subspaces \( b_1, b_2, \ldots, b_n \) can be computed from the derivatives of the polynomial

\[
b_i = Dp_n(x)|_{x=y_i} \quad y_i \in S_i
\]
The Society Raffles

©December 7, 1905
American Mutoscope & Biograph Company
Temporal Video Segmentation by GPCA

- Empty living room
- Middle-aged man enters
- Woman enters
- Young man enters, introduces the woman and leaves
- Middle-aged man flirts with woman and steals her tiara
- Middle-aged man checks the time, rises and leaves
- Woman walks him to the door
- Woman returns to her seat
- Woman misses her tiara
- Woman searches her tiara
- Woman sits and dismays

Fig. 5. Temporal segmentation of a scene from the movie *The society raffles*. The top row shows several key frames from the scene displaying different events. The bottom row shows the temporal evolution of the parameter $c_t$ as a function of time.
Sparse Subspace Clustering (SSC)

Ehsan Elhamifar and René Vidal
Sparse Subspace Clustering: Spectral Clustering

- Spectral clustering
  - Represent data points as nodes in graph $G$
  - Connect nodes $i$ and $j$ with weight $c_{ij}$
  - Infer clusters from Laplacian of $G$

- How to define a good affinity matrix $C$ for subspaces?
  - points in the same subspace: $c_{ij} \neq 0$
  - points in different subspaces: $c_{ij} = 0$
Sparse Subspace Clustering: Spectral Clustering

- Spectral curvature clustering (SCC) (Chen-Lerman ’08)
  - Define multiway similarity as normalized volume of d+1 points

- Local subspace affinity (LSA) (Yan-Pollefeys ’06)
  - Use the angles between locally fitted subspaces as similarity
Sparse Subspace Clustering: Intuition

- Data in a union of subspaces are self-expressive

\[ y_i = \sum_{j=1}^{N} c_{ji} y_j \implies y_i = Y c_i \implies Y = Y C \]

- Union of subspaces admits subspace-sparse representation

- Under what conditions on the subspaces and the data
  - \( L0 = \text{subspace sparse?} \)
  - \( L1 = \text{subspace sparse?} \)

\[ P_1 : \min \| c_i \|_1 \quad \text{s.t.} \quad y_i = Y c_i, \quad c_{ii} = 0 \]
**Sparse Subspace Clustering: Noiseless Data**

- **Theorem 1:** $P_1$ recovers a subspace-sparse representation if
  - Subspaces are independent:
    \[
    \dim\left(\bigoplus_{i=1}^{n} S_i\right) = \sum_{i=1}^{n} \dim(S_i)
    \]

\[P_1: \min \|c_i\|_1 \text{ s.t. } y_i = Yc_i, \quad c_{ii} = 0\]

Sparse Subspace Clustering: Noiseless Data

• **Theorem 2:** $P_1$ recovers a subspace-sparse representation if
  
  – Subspaces are disjoint: $S_i \cap S_j = \{0\}$
  
  – Subspaces are sufficiently well separated and data are sufficiently well distributed

\[
\max_{\text{rank}(\bar{Y}_i) = d_i} \sigma_{d_i}(\bar{Y}_i) > \sqrt{d_i} \max_{j \neq i} \cos(\theta_{ij})
\]

• $\theta_{ij}$ is the smallest subspace angle between subspaces $i$ and $j$
  
  – Subspace angles decrease \(\rightarrow\) harder recovery

• $\sigma_{d_i}(\bar{Y}_i)$ is the smallest singular value in each subspace
  
  – Data closer to a degenerate subspace \(\rightarrow\) harder recovery

\[
P_1 : \min \| c_i \|_1 \quad \text{s.t.} \quad y_i = Y c_i, \quad c_{ii} = 0
\]

Sparse Subspace Clustering: Noiseless Data

- **Theorem 3:**
  - $n$ $d$-dimensional subspaces chosen independently, uniformly at random.
  - $rd + 1$ points per subspace chosen independently, uniformly at random.
  - $P_1$ recovers a subspace-sparse representation with high probability if

\[
d < \frac{c^2(r) \log \rho}{12 \log N} D
\]

\[
P_1 : \min \| c_i \|_1 \quad \text{s.t.} \quad y_i = Yc_i, \quad c_{ii} = 0
\]

Sparse Subspace Clustering: Data with Outliers

• **Assumptions**
  - $n$ $d$-dimensional subspaces chosen independently, uniformly at random
  - $rd + 1$ inliers per subspace chosen independently, uniformly at random
  - $N_{\text{outliers}}$ outliers chosen independently and uniformly at random
  - Declare point $i$ as an outlier if the solution to $P_1$ satisfies
    \[
    \|c_i\|_1 > \lambda(\gamma) \sqrt{D}
    \]

• **Theorem 4:**
  - $P_1$ correctly detects all outliers with high probability if
    \[
    N_{\text{outliers}} < \frac{1}{D} e^{c\sqrt{D}} - N_{\text{inliers}}
    \]
  - $P_1$ does not detect any inlier as an outlier if
    \[
    P_1 : \min \|c_i\|_1 \text{ s.t. } y_i = Yc_i, \quad c_{ii} = 0
    \]

Sparse Subspace Clustering: Corrupted Data

• When the data are corrupted with noise \( \tilde{y} = y + e \)
\[
\min \| c_i \|_1 + \mu \| y_i - Y c_i \|_2
\]

• When the data have missing entries
  – Let \( I \subset \{1, \ldots, D\} \) be the indices of the missing entries in \( y \in \mathbb{R}^D \)
  – Form \( \tilde{y} \in \mathbb{R}^{D-|I|} \) and \( \tilde{Y} \in \mathbb{R}^{D-|I| \times N} \) by eliminating rows of \( y \) and \( Y \) indexed by \( I \), and solve the same optimization problems

• When the data are corrupted with outlying entries
  – Let \( \tilde{y} = Y c + e = [Y \quad I_D] \begin{bmatrix} c \\ e \end{bmatrix} \) be corrupted by a vector \( e \in \mathbb{R}^D \)
  – The vector \( [c^T \quad e^T]^T \) is still sparse and can be recovered from
\[
\min \| \begin{bmatrix} c \\ e \end{bmatrix} \|_1 + \mu \| \tilde{y} - [Y \quad I_D] \begin{bmatrix} c \\ e \end{bmatrix} \|_2
\]
Sparse Subspace Clustering: Algorithm

- Represent data points as nodes in graph $G$
- Find the sparse coefficient vectors $\{c_i\}_{i=1}^N$
\[
\min ||c_i||_1 + \mu ||y_i - Yc_i||_2
\]
- Connect nodes $i$ and $j$ by an edge with weight
\[
|c_{ij}| + |c_{ji}|
\]
- Spectral clustering: apply K-means to the smallest eigenvectors of the Laplacian of $G$
Low Rank Subspace Clustering (LRSC)

Paolo Favaro and René Vidal
Sparse Subspace Clustering: Spectral Clustering

- Spectral clustering
  - Represent data points as nodes in graph \( G \)
  - Connect nodes \( i \) and \( j \) with weight \( c_{ij} \)
  - Infer clusters from Laplacian of \( G \)

- How to define a good affinity matrix \( C \) for subspaces?
  - points in the same subspace: \( c_{ij} \neq 0 \)
  - points in different subspaces: \( c_{ij} = 0 \)
Sparse Subspace Clustering: Intuition

- Data in a union of subspaces are self-expressive

\[ y_i = \sum_{j=1}^{N} c_{ji} y_j \implies y_i = Y c_i \implies Y = YC \]

- Union of subspaces admits subspace-sparse representation

\[
P_1 : \min_{\|c_i\|_1} \quad \text{s.t.} \quad y_i = Y c_i, \quad c_{ii} = 0
\]
Subspace Clustering by Matrix Factorization

- Data from i-th subspace can be factorized as $Y_i = U_i V_i^T$

$$Y \Gamma = [Y_1, Y_2, \ldots, Y_n] = [U_1, U_2, \ldots, U_n] \begin{bmatrix} V_1^T & V_2^T & \cdots & V_n^T \end{bmatrix}$$

- Segmentation of the data can be obtained from
  - Leading singular vector of $Y = U \Sigma V^T$ (Boult and Brown ’91)
  - Shape interaction matrix $C = V V^T$ (Costeira & Kanade ’95, Gear ’94)

- $C_{ij} = 0$ if points i and j lie in two independent subspaces (Kanatani et al. ’01, Vidal et al. ’08)
Data in a union of subspaces are self-expressive

\[ y_i = \sum_{j=1}^{N} c_{ji} y_j \implies y_j = Y c_i \implies Y = Y C \]

\(-\quad C \text{ is sparse}\)
\(-\quad C \text{ is low-rank}\)

Low Rank Subspace Clustering (noiseless case)

\[
\min_{C} \|C\|_* \quad \text{s.t.} \quad Y = Y C \implies Y = U \Sigma \nu^T
\]
\[
C = \nu \nu^T
\]

Low Rank Subspace Clustering (noisy case)

\[
\min_{C} \|C\|_* + \frac{\tau}{2} \|Y - Y C\|_F^2 \implies C = \nu (I - \frac{1}{\tau} \Sigma^{-2}) \nu^T
\]
Applications in Computer Vision
Experiments on 3D Motion Segmentation

- Motion segmentation problem
  - Input: multiple images of a scene with multiple rigid-body motions
  - Output: number of motions, motion model parameters, segmentation

- Motion of a rigid-body: 4D subspace (Boult and Brown '91, Tomasi and Kanade '92)
  - $P = \#\text{points}$
  - $F = \#\text{frames}$

\[
\begin{bmatrix}
\vec{x}_{11} & \cdots & \vec{x}_{1P} \\
\vdots & \ddots & \vdots \\
\vec{x}_{F1} & \cdots & \vec{x}_{FP}
\end{bmatrix}_{2F \times P} = \\
\begin{bmatrix}
\vec{A}_1 \\
\vdots \\
\vec{A}_F
\end{bmatrix}_{2F \times 4} \times \\
\begin{bmatrix}
\vec{X}_1 & \cdots & \vec{X}_P
\end{bmatrix}_{4 \times P}
\]
Experiments on 3D Motion Segmentation

- Misclassification rates on Hopkins 155 database

Experiments on Video Segmentation

- Model each video segment as a low-dimensional subspace
- Cluster video frames into multiple segments

Advantages
- SSC easily detects sharp transitions in the video
- SSC can handle camera motion and scene variations
Experiments on Video Segmentation

- Model each video segment as a low-dimensional subspace
- Cluster video frames into multiple segments

**Advantages**
- SSC easily detects sharp transitions in the video
- SSC can handle camera motion and scene variations
Experiments on Face Clustering

- Faces under varying illumination
  - 9D subspace
- Extended Yale B dataset
  - 38 subjects
  - 64 images per subject
- Clustering error
  - SSC < 2.0% error for 2 subjects
  - SSC < 11.0% error for 10 subjects

Conclusions

• Many problems in computer vision can be posed as subspace clustering and classification problems
  – Spatial and temporal video segmentation
  – Face clustering under varying illumination
  – Face classification

• These problems can be solved using
  – Generalized Principal Component Analysis (GPCA)
  – Sparse Subspace Clustering (SSC)
  – Low Rank Subspace Clustering (LRSC)

• This algorithms is provably correct when
  – Subspaces are sufficiently separated
  – Data are well distributed within each subspace
What’s Next

- **Big Data** (Peng ’13, Dyer ’13, You ’15)

<table>
<thead>
<tr>
<th></th>
<th>GPCA</th>
<th>SSC</th>
<th>OMP</th>
<th>?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimension of the data</td>
<td>10</td>
<td>10,000</td>
<td>10,000</td>
<td>1M</td>
</tr>
<tr>
<td>Number of data points</td>
<td>1000</td>
<td>10,000</td>
<td>100,000</td>
<td>1M</td>
</tr>
</tbody>
</table>

- **Missing Data:** (Grubber ’04, Eriksson ’12, Balzano ’12, Pimentel ’14, Candes ’14, Yang’15)

Matrix of corrupted observations + Underlying low-rank matrix = Sparse error matrix

- Chong You
- Congyuan Yang
Acknowledgements

• **Algebraic Methods**
  – Y. Ma, S. Sastry, M. Tsakiris

• **Sparse and Low Rank**
  – E. Elhamifar, P. Favaro, C. You

• **Funding**
  – Sloan Research Fellowship
  – ONR Young Investigator Award
  – NSF CAREER Award 0447739

• **More information/code**
  – Vision Lab @ Johns Hopkins University [http://www.vision.jhu.edu](http://www.vision.jhu.edu)

Thank You!