CUR Decomposition and Subspace Clustering

Keaton Hamm

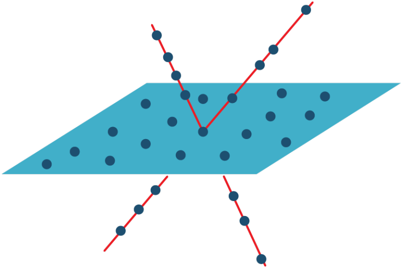
Vanderbilt University

ICCHA7

Joint with A. Aldroubi, A. Sekmen, and A. B. Koku

The Subspace Clustering Problem

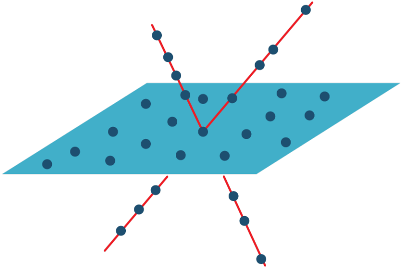
$$\mathscr{U} := \bigcup_{i=1}^{L} S_i, \quad S_i \subset \mathbb{R}^m$$



The Subspace Clustering Problem

$$\mathscr{U} := \bigcup_{i=1}^{L} S_i, \quad S_i \subset \mathbb{R}^m$$

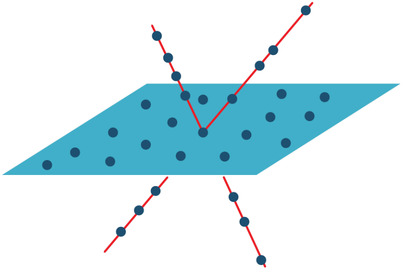
$$W = [w_1 \dots w_n] \in \mathbb{R}^{m \times n}, \quad w_i \in \mathscr{U}$$



The Subspace Clustering Problem

Goals:

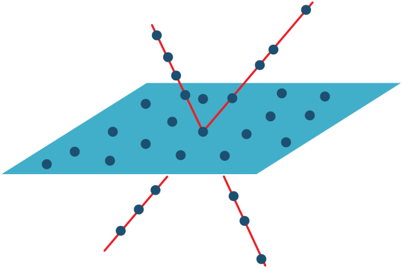
- # of Subspaces?



The Subspace Clustering Problem

Goals:

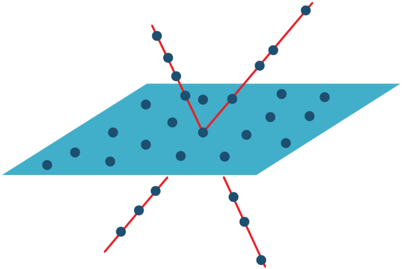
- # of Subspaces?
- $\dim(S_i)$?



The Subspace Clustering Problem

Goals:

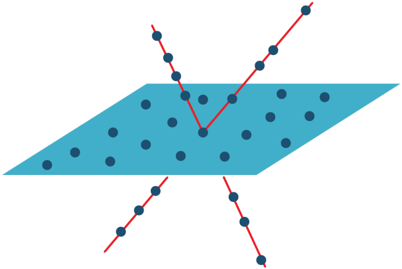
- # of Subspaces?
- $\dim(S_i)$?
- Basis for S_i?



The Subspace Clustering Problem

Goals:

- # of Subspaces?
- $\dim(S_i)$?
- Basis for S_i?
- Cluster data $\{w_i\}_{i=1}^n$.



The Subspace Clustering Problem

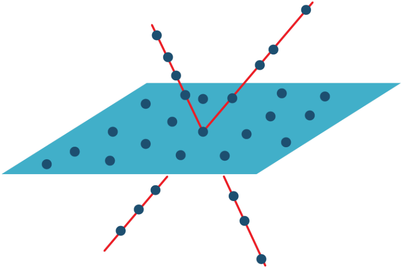
Goals:

- # of Subspaces?
- $\dim(S_i)$?
- Basis for S_i?
- Cluster data $\{w_i\}_{i=1}^n$.

Assumptions:

- Assume $\{S_i\}$ are *independent*:

$$\dim\left(\sum S_i\right) = \sum \dim(S_i)$$



The Subspace Clustering Problem

Goals:

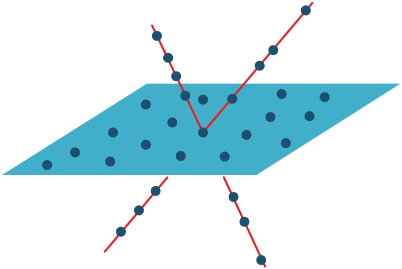
- # of Subspaces?
- $\dim(S_i)$?
- Basis for S_i?
- Cluster data $\{w_i\}_{i=1}^n$.

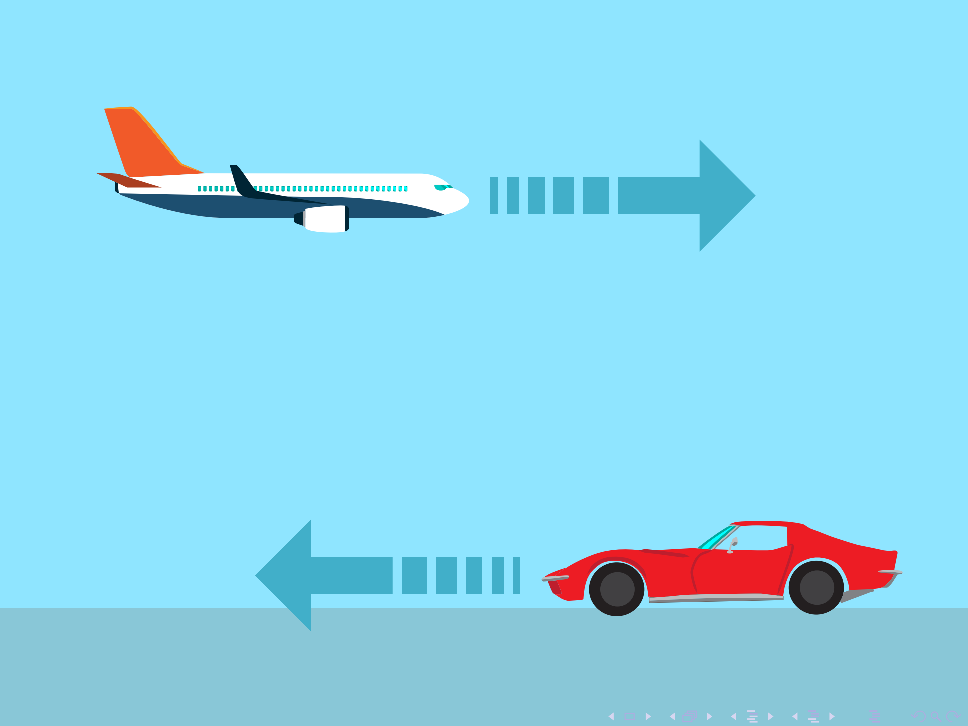
Assumptions:

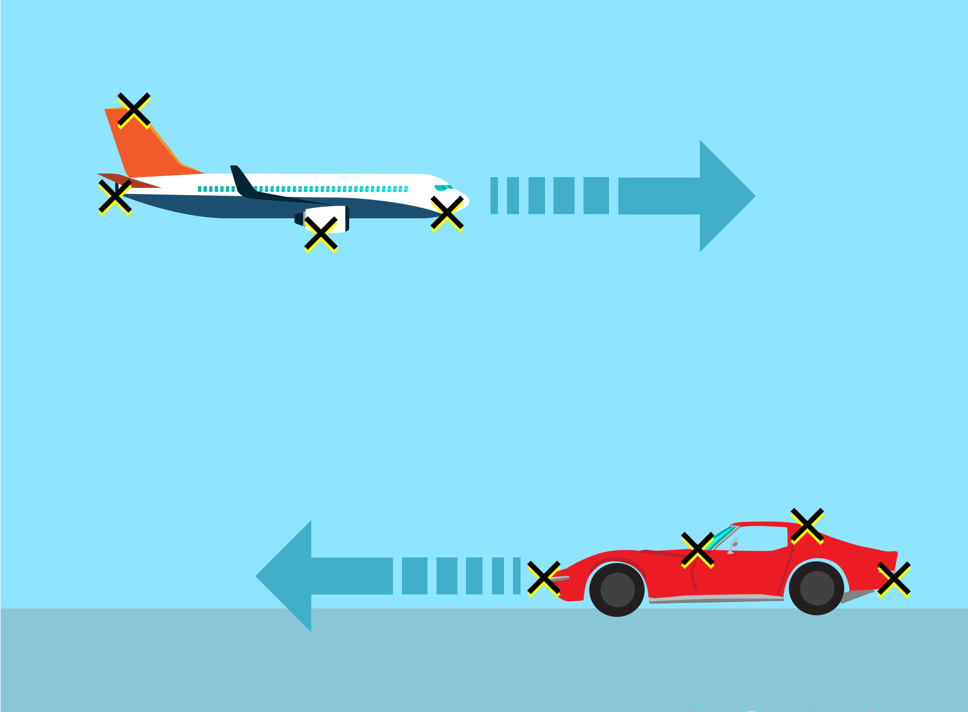
- Assume $\{S_i\}$ are *independent*:

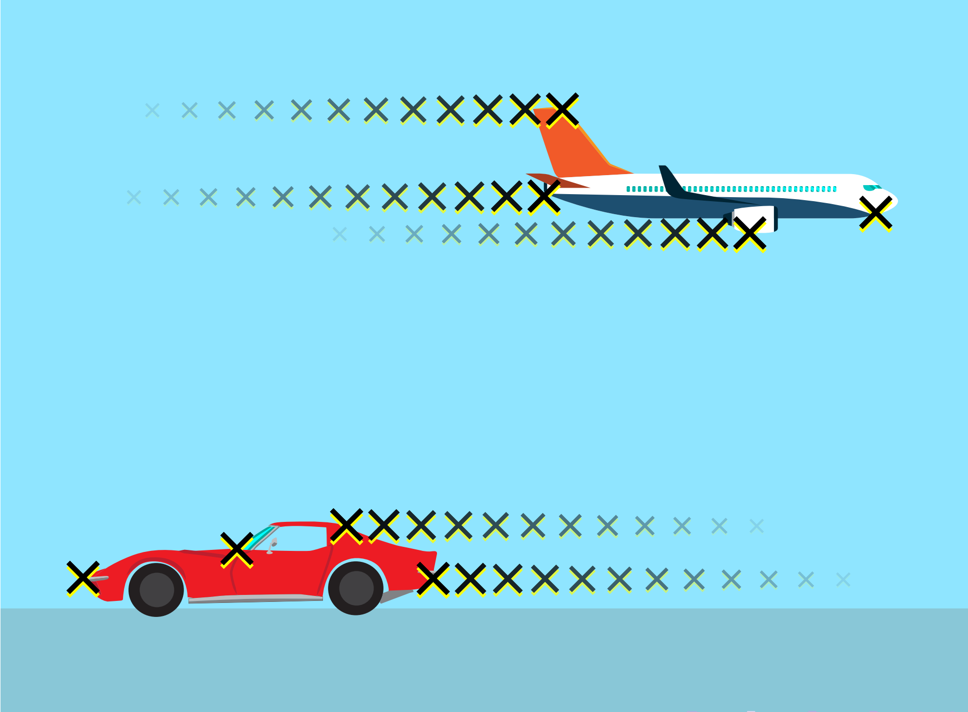
$$\dim\left(\sum S_i\right) = \sum \dim(S_i)$$

- Data from each subspace is *generic*

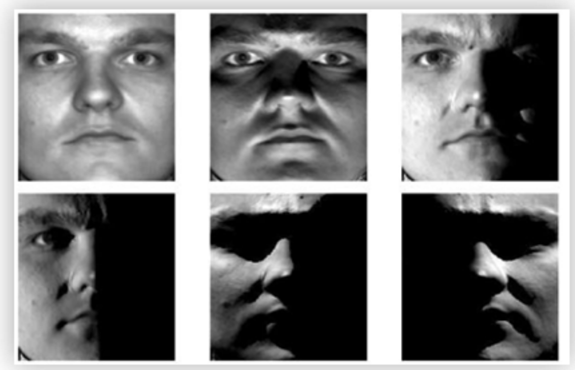








Example Application



Yale Face Database B: Georghiades, A.S. and Belhumeur, P.N. and Kriegman, D.J. From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose. IEEE Trans. Pattern Anal. Mach. Intelligence 23(6):643-660 (2001).

Similarity Matrix Methods

Goal: From W, find a similarity matrix S_W such that

Similarity Matrix Methods

Goal: From W, find a similarity matrix S_W such that

$$\begin{cases} S_W(i,j) \neq 0, & w_i, w_j \text{ are in the same subspace} \\ S_W(i,j) = 0, & \text{otherwise.} \end{cases}$$

Similarity Matrix Methods

Goal: From W, find a similarity matrix S_W such that

$$\begin{cases} S_W(i,j) \neq 0, & w_i, w_j \text{ are in the same subspace} \\ S_W(i,j) = 0, & \text{otherwise.} \end{cases}$$

Example: Shape Interaction Matrix (SIM) [1]

[1] J. P. Costeira and T. Kanade, A Multibody Factorization Method for Independently Moving Objects, *International Journal of Computer Vision*, **29**(3) (1998), 159-179.

Similarity Matrix Methods

Goal: From W, find a similarity matrix S_W such that

$$\begin{cases} S_W(i,j) \neq 0, & w_i, w_j \text{ are in the same subspace} \\ S_W(i,j) = 0, & \text{otherwise.} \end{cases}$$

Example: Shape Interaction Matrix (SIM) [1]

$$\text{If } W = U_r \Sigma_r V_r^T, \qquad \text{SIM}(W) = V_r V_r^T$$

[1] J. P. Costeira and T. Kanade, A Multibody Factorization Method for Independently Moving Objects, *International Journal of Computer Vision*, **29**(3) (1998), 159-179.

Similarity Matrix Methods

Goal: From W, find a similarity matrix S_W such that

$$\begin{cases} S_W(i,j) \neq 0, & w_i, w_j \text{ are in the same subspace} \\ S_W(i,j) = 0, & \text{otherwise.} \end{cases}$$

Example: Shape Interaction Matrix (SIM) [1]

$$\text{If } W = U_r \Sigma_r V_r^T, \qquad \text{SIM}(W) = V_r V_r^T$$

Problems:

1. Not always a similarity matrix[2]

[1] J. P. Costeira and T. Kanade, A Multibody Factorization Method for Independently Moving Objects, *International Journal of Computer Vision*, **29**(3) (1998), 159-179.

[2] A. Aldroubi, A. Sekmen, A.B. Koku, A. F. Cakmak, Similarity Matrix Framework for Data from Union of Subspaces, *ACHA*, In Press.

Similarity Matrix Methods

Goal: From W, find a similarity matrix S_W such that

$$\begin{cases} S_W(i,j) \neq 0, & w_i, w_j \text{ are in the same subspace} \\ S_W(i,j) = 0, & \text{otherwise.} \end{cases}$$

Example: Shape Interaction Matrix (SIM) [1]

$$\text{If } W = U_r \Sigma_r V_r^T, \qquad \text{SIM}(W) = V_r V_r^T$$

Problems:

1. Not always a similarity matrix[2]
2. Not robust to noise

[1] J. P. Costeira and T. Kanade, A Multibody Factorization Method for Independently Moving Objects, *International Journal of Computer Vision*, **29**(3) (1998), 159-179.

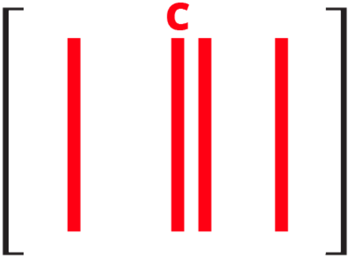
[2] A. Aldroubi, A. Sekmen, A.B. Koku, A. F. Cakmak, Similarity Matrix Framework for Data from Union of Subspaces, *ACHA*, In Press.

CUR Decomposition

- A $(m \times n)$,

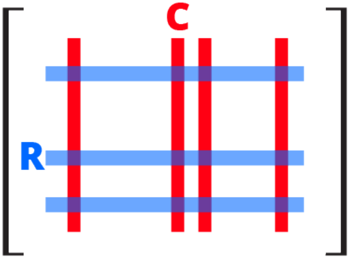
CUR Decomposition

- A $(m \times n)$,
- C $(m \times k)$ k columns of A



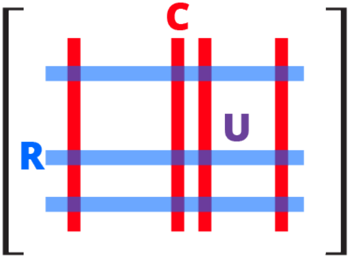
CUR Decomposition

- A $(m \times n)$,
- C $(m \times k)$ k columns of A
- R $(s \times n)$, s columns of A



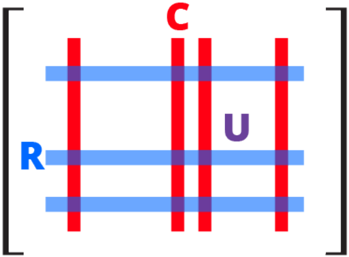
CUR Decomposition

- A $(m \times n)$,
- C $(m \times k)$ k columns of A
- R $(s \times n)$, s columns of A
- U $(s \times k)$, the intersection of C and R



CUR Decomposition

- A $(m \times n)$,
- C $(m \times k)$ k columns of A
- R $(s \times n)$, s columns of A
- U $(s \times k)$, the intersection of C and R



Theorem

If $rank(U) = rank(A)$, then

$$A = CU^{\dagger}R.$$

Proof

Case $s = k = \text{rank}(A)$. By assumption, U is invertible.

Proof

Case $s = k =$ rank(A). By assumption, U is invertible.
Columns of C form a basis for $col(A)$

Proof

Case $s = k =$ rank(A). By assumption, U is invertible.
Columns of C form a basis for $col(A)$

$$A = CX$$

Proof

Case $s = k = \text{rank(A)}$. By assumption, U is invertible.
Columns of C form a basis for $col(A)$

$$A = CX$$

P_r - row selection matrix: $P_r A = R$

Proof

Case $s = k =$ rank(A). By assumption, U is invertible.
Columns of C form a basis for $col(A)$

$$A = CX$$

P_r - row selection matrix: $P_r A = R$

$$\begin{aligned} A = CX &\Leftrightarrow P_r A = P_r CX \\ &\Leftrightarrow R = UX \end{aligned}$$

Proof

Case $s = k =$ rank(A). By assumption, U is invertible.
Columns of C form a basis for $col(A)$

$$A = CX$$

P_r - row selection matrix: $P_rA = R$

$$\begin{aligned} A = CX &\Leftrightarrow P_rA = P_rCX \\ &\Leftrightarrow R = UX \end{aligned}$$

$X = U^{-1}R$ is a solution to $R = UX$

Proof

Case $s = k = \text{rank(A)}$. By assumption, U is invertible.
Columns of C form a basis for $col(A)$

$$A = CX$$

P_r - row selection matrix: $P_r A = R$

$$\begin{aligned} A = CX &\Leftrightarrow P_r A = P_r CX \\ &\Leftrightarrow R = UX \end{aligned}$$

$X = U^{-1}R$ is a solution to $R = UX$
Therefore $A = CU^{-1}R$.

Relation to Subspace Clustering

$d_{\max} := \max \dim(S_i)$

Relation to Subspace Clustering

$d_{\max} := \max \dim(S_i)$

$|A|(i,j) = |A(i,j)|$

Relation to Subspace Clustering

$d_{\max} := \max \dim(S_i)$

$|A|(i,j) = |A(i,j)|$

$$\mathrm{bin}(A)(i,j) = \begin{cases} 1, & A(i,j) \neq 0 \\ 0, & A(i,j) = 0. \end{cases}$$

Relation to Subspace Clustering

$d_{\max} := \max \dim(S_i)$

$|A|(i,j) = |A(i,j)|$

$$\mathrm{bin}(A)(i,j) = \begin{cases} 1, & A(i,j) \neq 0 \\ 0, & A(i,j) = 0. \end{cases}$$

Theorem (Aldroubi, H, Koku, Sekmen, '17)

Suppose W is drawn from $\mathscr{U}$. If $W = CU^\dagger R$, let $Y = U^\dagger R$, and $Q = |Y^T Y|$ or $Q = \mathrm{bin}(Y^T Y)$. Then $Q^{d_{\max}}$ is a similarity matrix for W.

Special Cases

Theorem (Aldroubi, H, Koku, Sekmen, '17)

Suppose W is drawn from $\mathscr{U}$. If $W = CU^\dagger R$, let $Y = U^\dagger R$, and $Q = |Y^T Y|$ or $Q = \mathrm{bin}(Y^T Y)$. Then $Q^{d_{\max}}$ is a similarity matrix for W.

Special Cases

Theorem (Aldroubi, H, Koku, Sekmen, '17)

Suppose W is drawn from $\mathscr{U}$. If $W = CU^\dagger R$, let $Y = U^\dagger R$, and $Q = |Y^T Y|$ or $Q = bin(Y^T Y)$. Then $Q^{d_{\max}}$ is a similarity matrix for W.

Corollary

If $Q = |VV^T| = |SIM(W)|$, then $Q^{d_{\max}}$ is a similarity matrix for W.

Special Cases

Theorem (Aldroubi, H, Koku, Sekmen, '17)

Suppose W is drawn from $\mathscr{U}$. If $W = CU^{\dagger}R$, let $Y = U^{\dagger}R$, and $Q = |Y^TY|$ or $Q = bin(Y^TY)$. Then $Q^{d_{\max}}$ is a similarity matrix for W.

Corollary

If $Q = |VV^T| = |SIM(W)|$, then $Q^{d_{\max}}$ is a similarity matrix for W.

1. Choose $C = R = W$, then $Q = |W^{\dagger}W|$

Special Cases

Theorem (Aldroubi, H, Koku, Sekmen, '17)

Suppose W is drawn from $\mathscr{U}$. If $W = CU^\dagger R$, let $Y = U^\dagger R$, and $Q = |Y^T Y|$ or $Q = bin(Y^T Y)$. Then $Q^{d_{\max}}$ is a similarity matrix for W.

Corollary

If $Q = |VV^T| = |SIM(W)|$, then $Q^{d_{\max}}$ is a similarity matrix for W.

1. Choose $C = R = W$, then $Q = |W^\dagger W|$

Proof of Corollary: Choose $C = R = W$, then $Q = |W^\dagger W| = |V_r V_r^T|$

Consequences

- Provides many similarity matrices from data, W

Consequences

- Provides many similarity matrices from data, W
- Can handle missing data (unlike SIM)

Consequences

- Provides many similarity matrices from data, W
- Can handle missing data (unlike SIM)
- CUR preserves data structure (e.g. sparsity)

Consequences

- Provides many similarity matrices from data, W
- Can handle missing data (unlike SIM)
- CUR preserves data structure (e.g. sparsity)
- More robust to noise

Consequences

- Provides many similarity matrices from data, W
- Can handle missing data (unlike SIM)
- CUR preserves data structure (e.g. sparsity)
- More robust to noise
- Relates similarity matrices with Compressed Sensing-inspired minimization techniques

Relation to Minimization Problems

Low-Rank Representation[3]

$$\min_Z \operatorname{rank} Z \quad \text{s.t.} \quad W = WZ$$

[3] G. Liu, Z. Lin, Y. Yu, Robust subspace segmentation by low-rank representation, in: International Conference on Machine Learning, 2010, pp. 663-670.

Relation to Minimization Problems

Low-Rank Representation[3]

$$\min_{Z} \operatorname{rank} Z \quad \text{s.t.} \quad W = WZ$$

A minimizer is $Z = W^\dagger W = V_r V_r^T$

[3] G. Liu, Z. Lin, Y. Yu, Robust subspace segmentation by low-rank representation, in: International Conference on Machine Learning, 2010, pp. 663-670.

Relation to Minimization Problems

Low-Rank Representation[3]

$$\min_Z \operatorname{rank} Z \quad \text{s.t.} \quad W = WZ$$

A minimizer is $Z = W^\dagger W = V_r V_r^T$
Equivalent to

$$\min_Z \|\sigma(Z)\|_0 \quad \text{s.t.} \quad W = WZ$$

[3] G. Liu, Z. Lin, Y. Yu, Robust subspace segmentation by low-rank representation, in: International Conference on Machine Learning, 2010, pp. 663-670.

 Relation to Minimization Problems

Low-Rank Representation[3]

$$\min_Z \operatorname{rank} Z \quad \text{s.t.} \quad W = WZ$$

A minimizer is $Z = W^\dagger W = V_r V_r^T$
Equivalent to

$$\min_Z \|\sigma(Z)\|_0 \quad \text{s.t.} \quad W = WZ$$

Compressed Sensing Inspiration: Relax to

$$\min_Z \|\sigma(Z)\|_1 \quad \text{s.t.} \quad W = WZ$$

[3] G. Liu, Z. Lin, Y. Yu, Robust subspace segmentation by low-rank representation, in: International Conference on Machine Learning, 2010, pp. 663-670.

Relation to Minimization Problems

Low-Rank Representation[3]

$$\min_Z \operatorname{rank} Z \quad \text{s.t.} \quad W = WZ$$

A minimizer is $Z = W^\dagger W = V_r V_r^T$
Equivalent to

$$\min_Z \|\sigma(Z)\|_0 \quad \text{s.t.} \quad W = WZ$$

Compressed Sensing Inspiration: Relax to

$$\min_Z \|\sigma(Z)\|_1 \quad \text{s.t.} \quad W = WZ$$

Theorem (Liu, et. al.)

$V_r V_r^T$ (hence $W^\dagger W$) is the unique minimizer of the ℓ_1 problem above.

[3] G. Liu, Z. Lin, Y. Yu, Robust subspace segmentation by low-rank representation, in: International Conference on Machine Learning, 2010, pp. 663-670.

Hopkins155 Motion Dataset[4]

[4] R. Tron and R. Vidal, A Benchmark for the Comparison of 3-D Motion Segmentation Algorithms. IEEE International Conference on Computer Vision and Pattern Recognition, June 2007.

Experiments

Hopkins155 Motion Dataset[4]

Method	GPCA	RANSAC	MSL	ALC	SSC	RSIM	NLS	SIM	CUR
Average	10.34%	9.76%	5.03%	3.56%	1.45%	0.89%	0.76%	7.08%	$\sim$ 1.97%

Table: % Classification errors for all 155 sequences in Hopkins155 database.

[4]R. Tron and R. Vidal, A Benchmark for the Comparison of 3-D Motion Segmentation Algorithms. IEEE International Conference on Computer Vision and Pattern Recognition, June 2007.

Proto-algorithm for CUR Motion Clustering

- Find CUR decomposition, $W = CU^\dagger R$

Proto-algorithm for CUR Motion Clustering

- Find CUR decomposition, $W = CU^{\dagger}R$
- $Y = U^{\dagger}R$

Proto-algorithm for CUR Motion Clustering

- Find CUR decomposition, $W = CU^\dagger R$
- $Y = U^\dagger R$
- Threshold Y

Proto-algorithm for CUR Motion Clustering

- Find CUR decomposition, $W = CU^{\dagger}R$
- $Y = U^{\dagger}R$
- Threshold Y
- $Q = |Y^T Y|$

Proto-algorithm for CUR Motion Clustering

- Find CUR decomposition, $W = CU^{\dagger}R$
- $Y = U^{\dagger}R$
- Threshold Y
- $Q = |Y^T Y|$
- Repeat n times, and take $S_W =$ average of n trials of Q

Proto-algorithm for CUR Motion Clustering

- Find CUR decomposition, $W = CU^{\dagger}R$
- $Y = U^{\dagger}R$
- Threshold Y
- $Q = |Y^T Y|$
- Repeat n times, and take $S_W =$ average of n trials of Q
- Cluster columns of S_W

Future Directions

- Most effective use of CUR in real applications?

Future Directions

- Most effective use of CUR in real applications?
- Non-independent subspaces?

Thanks!

A. Aldroubi, K. Hamm, A. B. Koku, and A. Sekmen, CUR Decompositions, Similarity Matrices, and Subspace Clustering, arXiv: 1711.04178