Random Consensus Robust PCA

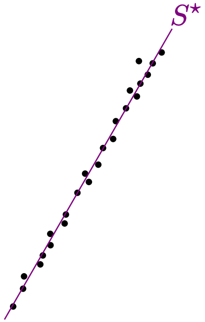
Daniel L. Pimentel-Alarcón,
Nigel Boston and Robert Nowak

University of Wisconsin-Madison

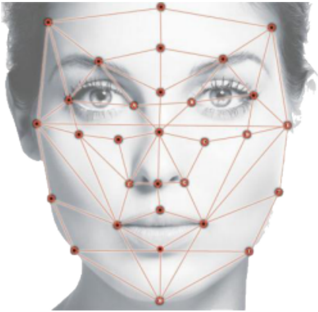
SILO Seminar, 2016

In many Applications we want to Learn Subspaces

$$\mathbf{X} = \begin{bmatrix} 1 & 4 & 1 & 3 & 3 & 1 & 2 & 1 & 2 & 1 \\ 2 & 4 & 2 & 6 & 3 & 2 & 2 & 2 & 4 & 1 \\ 3 & 4 & 3 & 9 & 3 & 3 & 2 & 3 & 6 & 1 \\ 1 & 8 & 1 & 3 & 6 & 1 & 4 & 1 & 2 & 2 \\ 2 & 8 & 2 & 6 & 6 & 2 & 4 & 2 & 4 & 2 \\ 3 & 8 & 3 & 9 & 6 & 3 & 4 & 3 & 6 & 2 \end{bmatrix}$$



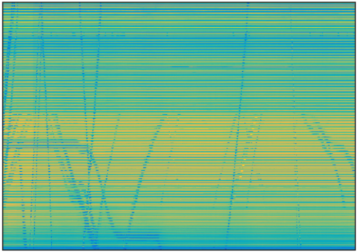
In many Applications we want to Learn Subspaces



In many Applications we want to Learn Subspaces

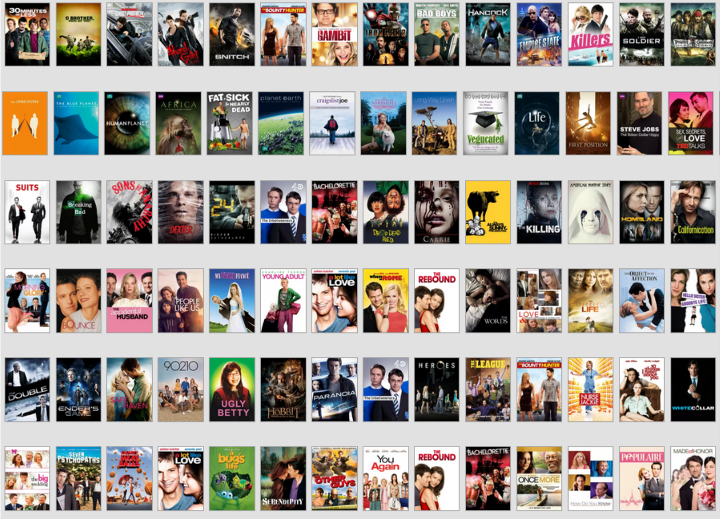
In many Applications we want to Learn Subspaces

In many Applications we want to Learn Subspaces

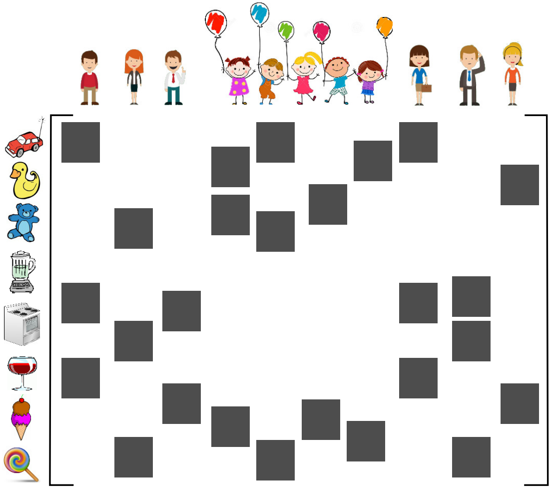


In many Applications we want to Learn Subspaces

In many Applications we want to Learn Subspaces

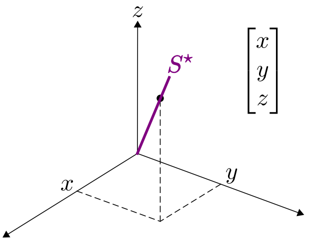


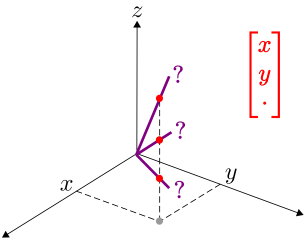
In many Applications we want to Learn Subspaces



We need to Learn Subspaces by Pieces

$$\mathbf{X} = \begin{bmatrix} 1 & 4 & 1 & 3 & 3 & 1 & 2 & 1 & 2 & 1 \\ 2 & 4 & 2 & 6 & 3 & 2 & 2 & 2 & 4 & 1 \\ 3 & 4 & 3 & 9 & 3 & 3 & 2 & 3 & 6 & 1 \\ 1 & 8 & 1 & 3 & 6 & 1 & 4 & 1 & 2 & 2 \\ 2 & 8 & 2 & 6 & 6 & 2 & 4 & 2 & 4 & 2 \\ 3 & 8 & 3 & 9 & 6 & 3 & 4 & 3 & 6 & 2 \end{bmatrix} \qquad \mathbf{X}_{\boldsymbol{\Omega}} = \begin{bmatrix} 1 & \cdot & \cdot & 3 & \cdot & 3 & \cdot & 1 & 2 & \cdot \\ 2 & \cdot & 2 & \cdot & \cdot & 6 & \cdot & \cdot & 4 & \cdot \\ \cdot & \cdot & 3 & \cdot & \cdot & 9 & \cdot & 3 & 6 & \cdot \\ 1 & \cdot & 1 & 3 & 6 & \cdot & 4 & 1 & 2 & 2 \\ \cdot & 8 & \cdot & \cdot & 6 & \cdot & 4 & \cdot & \cdot & \cdot \\ \cdot & 8 & \cdot & \cdot & \cdot & \cdot & 4 & \cdot & \cdot & 2 \end{bmatrix}$$

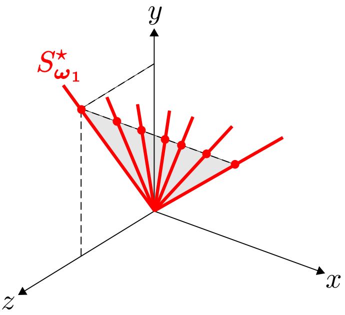




Learning Subspaces by Pieces

A column with $r+1$ observations imposes one restriction on what $S^\star$ may be.

$$\mathbf{X}_{\boldsymbol{\Omega}} = \begin{array}{c} \boldsymbol{x}_{\boldsymbol{\omega}_1} \\ \begin{bmatrix} \cdot \\ 1 \\ 1 \end{bmatrix} \end{array}$$



- A subspace S *agrees* with $\boldsymbol{x}_{\boldsymbol{\omega}_1} \iff \underbrace{f_1(S) = 0}_{\text{degree-}r\text{ polynomial}}$.

Learning Subspaces by Pieces

More precisely:

- Take a basis of S:

$$S \;=\; \mathrm{span}\left.\underbrace{\begin{bmatrix} \\ \mathbf{U} \\ \\ \end{bmatrix}}_{r}\right\} d.$$

- Then $\boldsymbol{x}_{\boldsymbol{\omega}_i} \in S$ is equivalent to:

$$r+1\left\{\begin{bmatrix} \boldsymbol{x}_{\boldsymbol{\omega}_i} \end{bmatrix}\right. = \begin{bmatrix} \mathbf{U}_{\boldsymbol{\omega}_i} \end{bmatrix} \boldsymbol{\theta_i}.$$

Learning Subspaces by Pieces

- We can split this as:

$$\begin{array}{r} r\left\{ \vphantom{\begin{matrix} a\\ b\\ c \end{matrix}} \right. \\ 1\left\{ \vphantom{a} \right. \end{array} \left[\begin{array}{c} \boldsymbol{x}_{\boldsymbol{\Delta}_i} \\ \hline \boldsymbol{x}_{\boldsymbol{\nabla}_i} \end{array}\right] = \left[\begin{array}{c} \mathbf{U}_{\boldsymbol{\Delta}_i} \\ \hline \mathbf{U}_{\boldsymbol{\nabla}_i} \end{array}\right] \boldsymbol{\theta}_i.$$

- We can use the top block to solve for $\boldsymbol{\theta}_i$:

$$\boldsymbol{\theta}_i = \mathbf{U}_{\boldsymbol{\Delta}_i}^{-1} \boldsymbol{x}_{\boldsymbol{\Delta}_i}.$$

- Plug this in the last row:

$$\boldsymbol{x}_{\boldsymbol{\nabla}_i} = \mathbf{U}_{\boldsymbol{\nabla}_i} \mathbf{U}_{\boldsymbol{\Delta}_i}^{-1} \boldsymbol{x}_{\boldsymbol{\Delta}_i}.$$

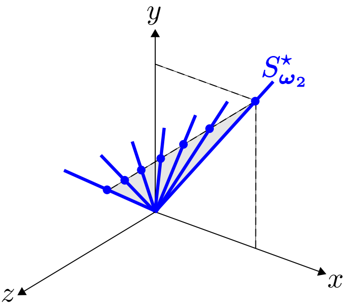
- Or equivalently

$$\underbrace{\boldsymbol{x}_{\boldsymbol{\nabla}_i} - \mathbf{U}_{\boldsymbol{\nabla}_i} \mathbf{U}_{\boldsymbol{\Delta}_i}^{-1} \boldsymbol{x}_{\boldsymbol{\Delta}_i}}_{f_i(\mathbf{U}_{\boldsymbol{\omega}_i} | \boldsymbol{x}_{\boldsymbol{\omega}_i})} = 0.$$

Learning Subspaces by Pieces

An other column with $r+1$ samples imposes an other restriction.

$$\mathbf{X}_{\mathbf{\Omega}} = \begin{array}{c} \boldsymbol{x}_{\boldsymbol{\omega}_2} \\ \left[\begin{array}{c} 2 \\ 2 \\ \cdot \end{array}\right] \end{array}$$

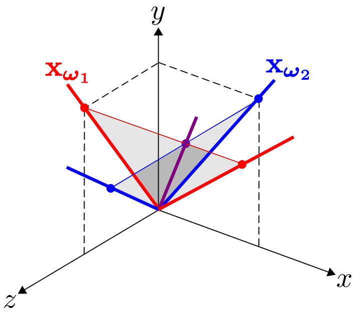


- A subspace S *agrees* with $\boldsymbol{x}_{\boldsymbol{\omega}_2} \iff f_2(\mathbf{U}_{\boldsymbol{\omega}_2}|\boldsymbol{x}_{\boldsymbol{\omega}_2}) = 0$.

Learning Subspaces by Pieces

Each column with $r+1$ samples imposes one restriction.

$$\mathbf{X_\Omega} = \begin{array}{cc} \boldsymbol{x}_{\boldsymbol{\omega}_1} & \boldsymbol{x}_{\boldsymbol{\omega}_2} \\ \end{array}$$
$$\mathbf{X_\Omega} \;=\; \left[\begin{array}{cc} \cdot & 2 \\ 1 & 2 \\ 1 & \cdot \end{array}\right]$$



- A subspace S *agrees* with $\mathbf{X_\Omega}$ $\iff \left\{ \begin{array}{l} f_1(\mathbf{U}_{\boldsymbol{\omega}_1} | \boldsymbol{x}_{\boldsymbol{\omega}_1}) = 0 \\ f_2(\mathbf{U}_{\boldsymbol{\omega}_2} | \boldsymbol{x}_{\boldsymbol{\omega}_2}) = 0 \end{array} \right.$.

Learning Subspaces by Pieces

- We thus obtain a set of *generic* polynomials:

$$f_1(\mathbf{U}_{\boldsymbol{\omega}_1}), f_2(\mathbf{U}_{\boldsymbol{\omega}_2}), \ldots, f_N(\mathbf{U}_{\boldsymbol{\omega}_N}).$$

- Polynomial f_i only involves the variables indicated in $\boldsymbol{\omega}_i$.
- Construct $\boldsymbol{\Omega} = [\boldsymbol{\omega}_1 \ \boldsymbol{\omega}_2 \ \cdots \ \boldsymbol{\omega}_N]$.
  - Each column of $\boldsymbol{\Omega}$ corresponds to one polynomial.
  - Its nonzero rows indicate the variables involved.

Learning Subspaces by Pieces

- Polynomials are a algebraically independent iff

$$\underbrace{n(\mathbf{\Omega}')}_{equations} \leq \underbrace{r(m(\mathbf{\Omega}') - r)}_{unknowns} \qquad \forall\ \mathbf{\Omega}' \subset \mathbf{\Omega}.$$

After this, deep algebraic geometry results do the heavy lifting:

$\Leftrightarrow$ Polynomials are a regular sequence.

$\Leftrightarrow$ Polynomials define a zero-dimensional variety.

$\Leftrightarrow$ At most finitely many solutions (subspaces) will agree with $\mathbf{X}_{\mathbf{\Omega}}$.

Learning Subspaces by Pieces

Theorem (P.-A., Boston, Nowak)

For almost every $\mathbf{X}$*, at most finitely many* r*-dimensional subspaces can agree with* $\mathbf{X}_{\mathbf{\Omega}}$ *if and only if every matrix* $\mathbf{\Omega}'$ *formed with a subset of the columns in* $\mathbf{\Omega}$ *satisfies*

$$m(\mathbf{\Omega}') \;\geq\; n(\mathbf{\Omega}')/r + r.$$

Learning Subspaces by Pieces

Theorem (P.-A., Boston, Nowak)

For almost every $\mathbf{X}$*, at most finitely many* r*-dimensional subspaces can agree with* $\mathbf{X}_{\mathbf{\Omega}}$ *if and only if every matrix* $\mathbf{\Omega}'$ *formed with a subset of the columns in* $\mathbf{\Omega}$ *satisfies*

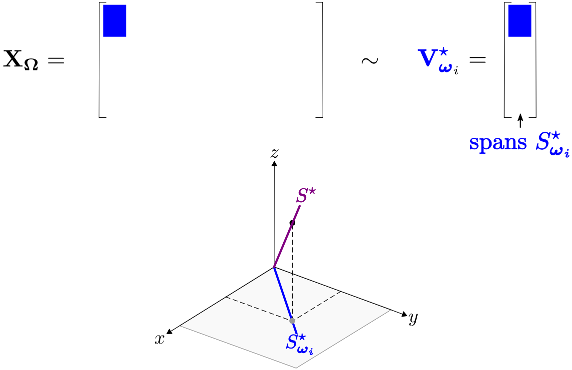
$$m(\mathbf{\Omega}') \;\geq\; n(\mathbf{\Omega}')/r + r.$$

$$\mathbf{X}_{\mathbf{\Omega}} = \begin{bmatrix} 1 & \cdot & 3 & \cdot \\ 1 & 2 & \cdot & \cdot \\ \cdot & 2 & 3 & \cdot \\ \cdot & \cdot & \cdot & 4 \\ \cdot & \cdot & \cdot & 4 \end{bmatrix} \qquad \mathbf{X}_{\mathbf{\Omega}} = \begin{bmatrix} 1 & \cdot & \cdot & \cdot \\ 1 & 2 & \cdot & \cdot \\ \cdot & 2 & 3 & \cdot \\ \cdot & \cdot & 3 & 4 \\ \cdot & \cdot & \cdot & 4 \end{bmatrix}$$

$$\underbrace{m(\mathbf{\Omega}')}_{3} \not\geq \underbrace{n(\mathbf{\Omega}')/r + r}_{4} \qquad \underbrace{m(\mathbf{\Omega}')}_{4} \geq \underbrace{n(\mathbf{\Omega}')/r + r}_{4}$$

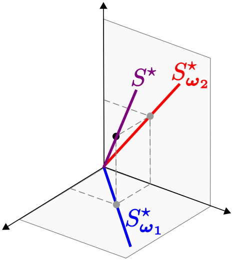
Some Pieces are Better than Others

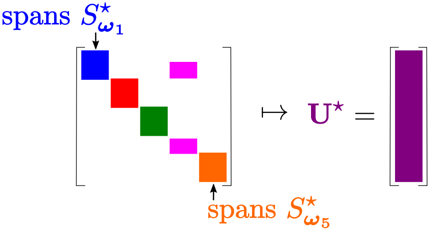
If we observe blocks, then polynomials become linear!



Some Pieces are Better than Others

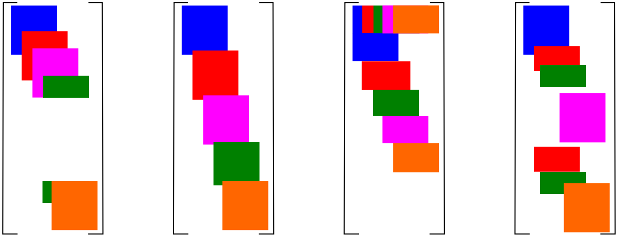
- We are given a bunch of *pieces* of the subspace.
- We want to reconstruct the whole subspace.





Theorem tells us...

- Which pieces to observe.
- How to reconstruct the subspace.

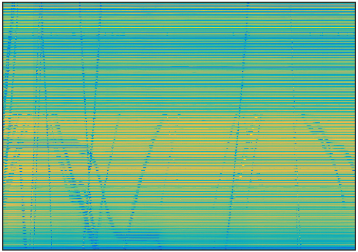


- Now we know which pieces we need.
- And how to reconstruct $S^\star$ from its pieces.
- OK, cool, that's all very nice, but...

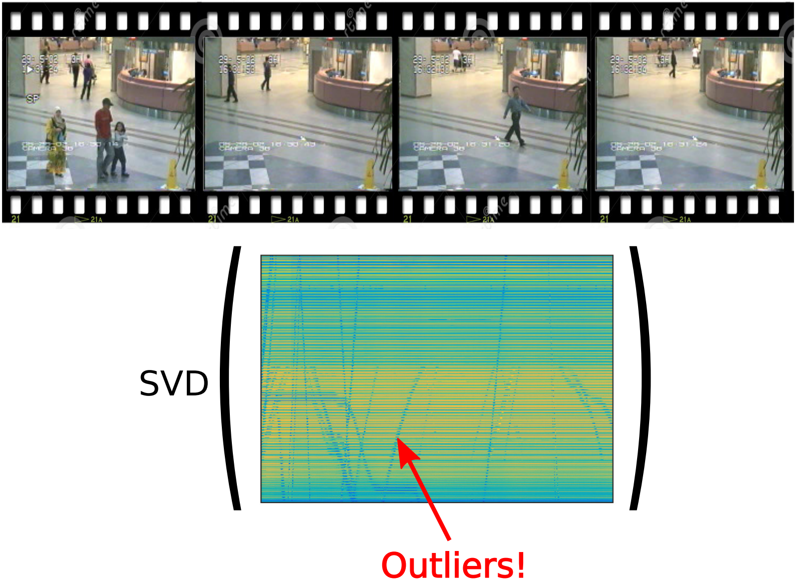
What is this good for?

1. Background Segmentation
- If time allows
  2. Clustering
  3. Missing Data

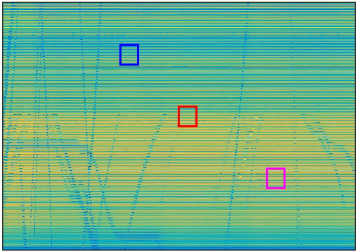
Background Segmentation



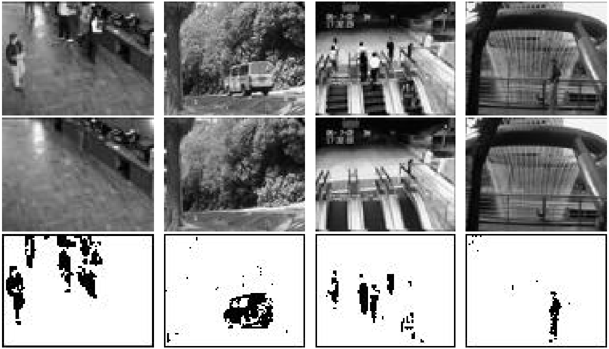
Background Segmentation



Background Segmentation

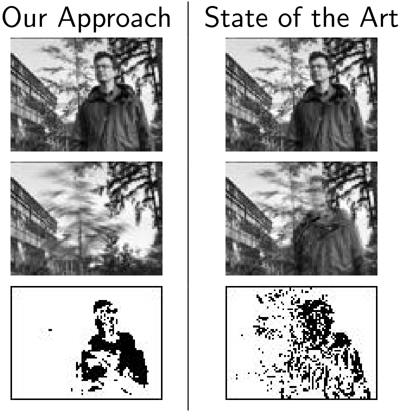


Background Segmentation



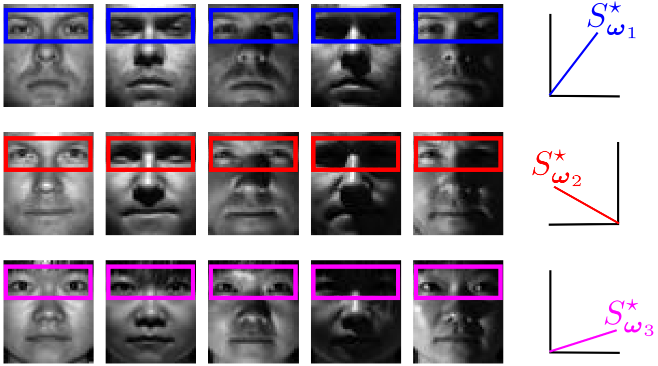
Background Segmentation

Background Segmentation

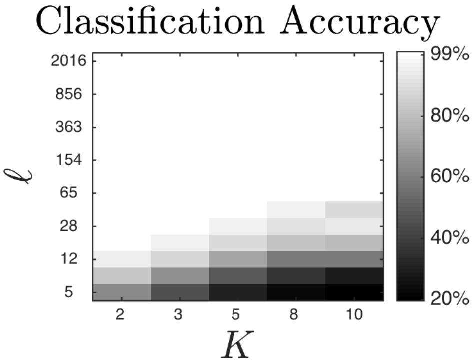


How am I on time?

Clustering

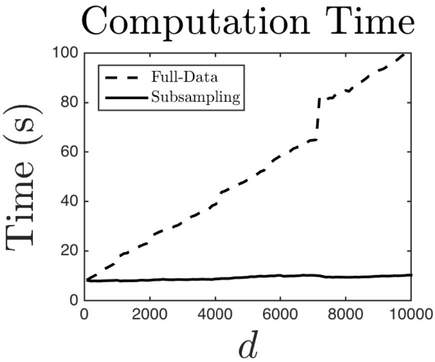


Clustering



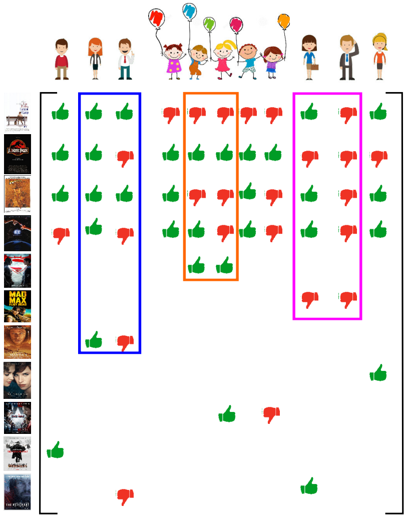
Classification Accuracy
2016
856
363
154
65
28
12
5
ℓ
2
3
5
8
10
K
99%
80%
60%
40%
20%

Clustering



Computation Time
Full-Data
Subsampling
Time (s)
100
80
60
40
20
0
0
2000
4000
6000
8000
10000
d

Missing Data



Thanks.