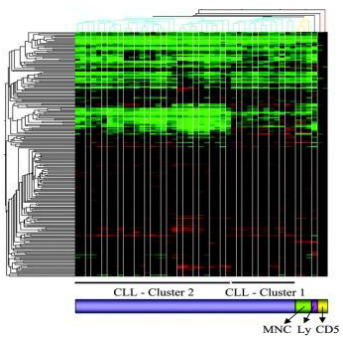
Low Rank Approximation

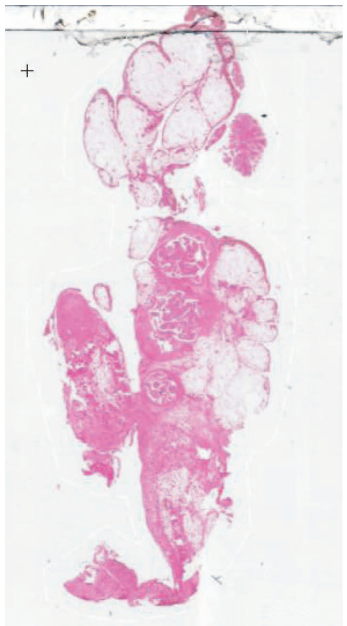
Motivation

- The curse of dimensionality

$$n \text{ ball:} \qquad V_n(R) = \frac{\pi^{n/2}}{\Gamma(\frac{n}{2}+1)} R^n$$

  – inference for limited samples with very high-dimensional data

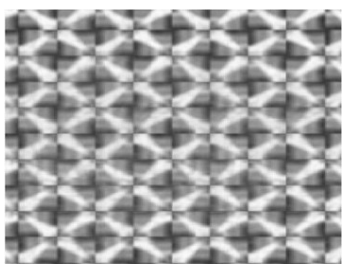


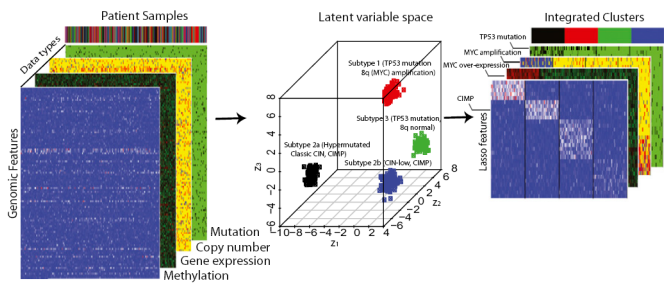


Motivation

- The blessing of dimensionality

  Real data highly concentrate on low-dimensional, sparse, or degenerate **structures** in the high-dimensional space





Goal

Let the data and application tell you the structure ...

A Brief History of l_1 regularization (Sparsity)

- LASSO for linear regression: in statistics (Tibshirani,1995)
- Basis Pursuit: in signal processing (Donoho and Saunders,1996)
- Extension to generalized linear models: e.g., logistic regression.
- Structured sparsity: e.g., fused lasso, group lasso, elastic net, graphical lasso and so on.
- Compressive sensing: near exact recovery of sparse signals in very high dimensions(Donoho 2004, Candes and Tao 2005)
- **Low-rank approximation: from vectors to matrices**

Sparsity of Matrices

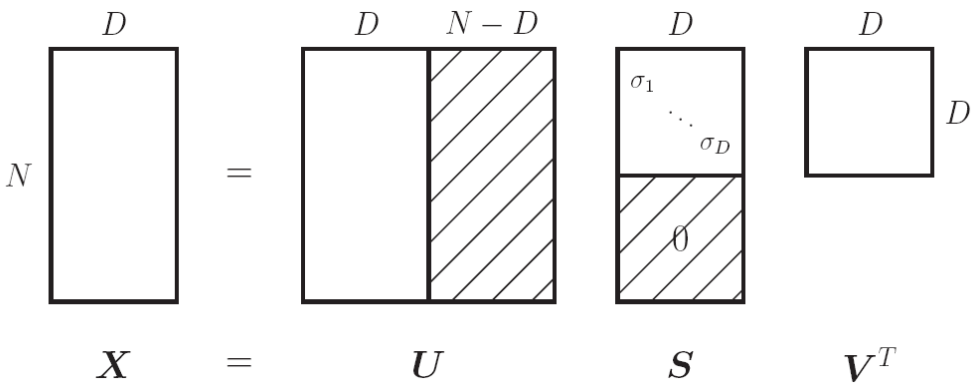
- Elements of matrix are sparse: $||M||_0$
- Singular values of matrix are sparse: $\text{rank}(M)$

Sparsity of Matrices

- Elements of matrix are sparse: $||M||_0$
- Singular values of matrix are sparse: $\text{rank}(M)$

Singular value decomposition

$$M = U\Sigma V^T$$

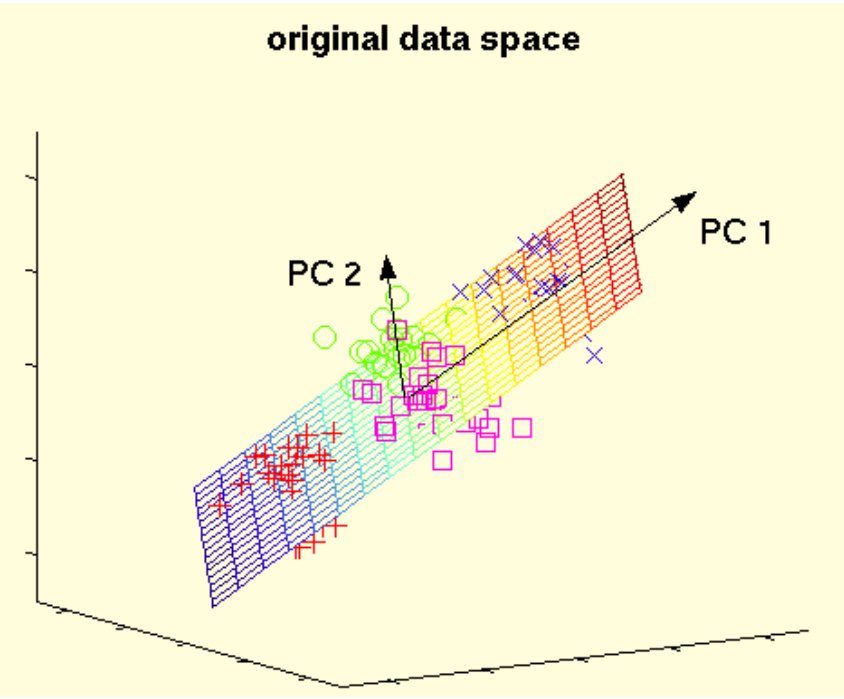


- Singular values: $\sigma_i(M) \geq 0$
- $\text{rank}(M) = \sum_i \mathbb{I}(\sigma_i \neq 0)$

Content

1. Robust PCA
2. Algorithms
3. Sparse Subspace Clustering

PCA as a low rank approximation



PCA: find a low-dimensional representation $\{y_j \in R^d\}$ of a set of sample points $\{x_j \in R^D\}$

- Rank minimization:

  Find a vector μ and rank-d matrix A that best approximate the data matrix X: (suppose $\text{rank}(X) \geq d$)

$$\begin{aligned} \min_{\mu, A} \quad & \sum_{j=1}^{N} ||x_j - \mu - A||^2 \\ s.t. \quad & \text{rank}(A) = d, \qquad A\mathbf{1} = 0 \end{aligned}$$

- Rank minimization:

  Find a vector μ and rank-d matrix A that best approximate the data matrix X: (suppose $\text{rank}(X) \geq d$)

$$\begin{aligned} \min_{\mu, A} \quad & \sum_{j=1}^{N} ||x_j - \mu - A||^2 \\ s.t. \quad & \text{rank}(A) = d, \qquad A\mathbf{1} = 0 \end{aligned}$$

- PCA Solution:

$$\hat{A} = U\hat{\Sigma}_d V^T \qquad \text{where} \qquad \hat{\Sigma}_d = \text{diag}(\sigma_1, \cdots, \sigma_d, 0, \cdots, 0)$$

- Rank minimization:

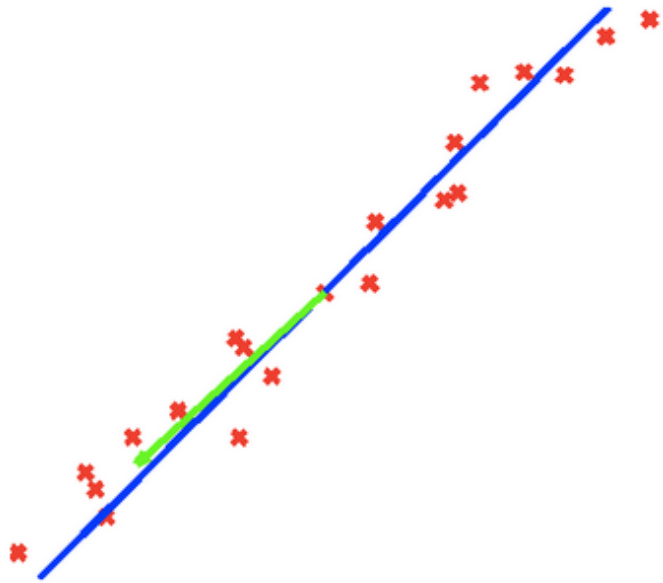
  Find a vector μ and rank-d matrix A that best approximate the data matrix X: (suppose $\text{rank}(X) \geq d$)

$$\begin{aligned} \min_{\mu,A} \quad & \sum_{j=1}^{N} ||x_j - \mu - A||^2 \\ s.t. \quad & \text{rank}(A) = d, \qquad A\mathbf{1} = 0 \end{aligned}$$

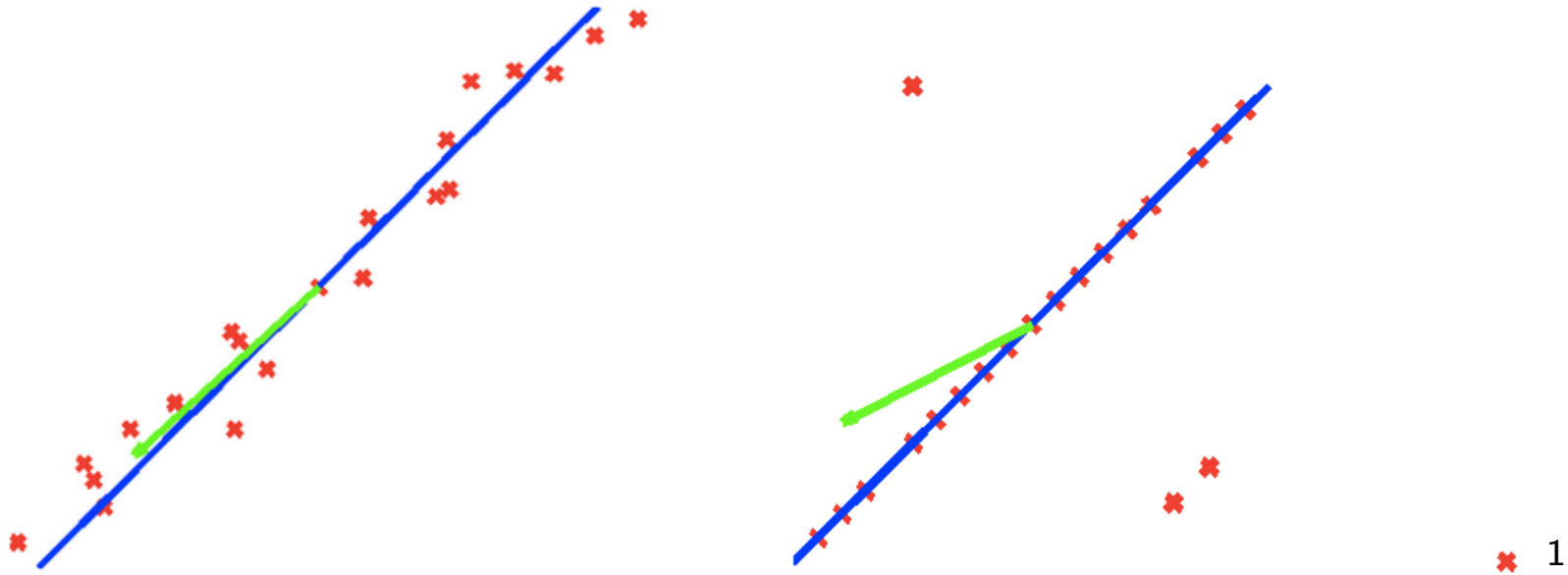
- PCA Solution:

$$\hat{A} = U\hat{\Sigma}_d V^T \qquad \text{where} \qquad \hat{\Sigma}_d = \text{diag}(\sigma_1, \cdots, \sigma_d, 0, \cdots, 0)$$

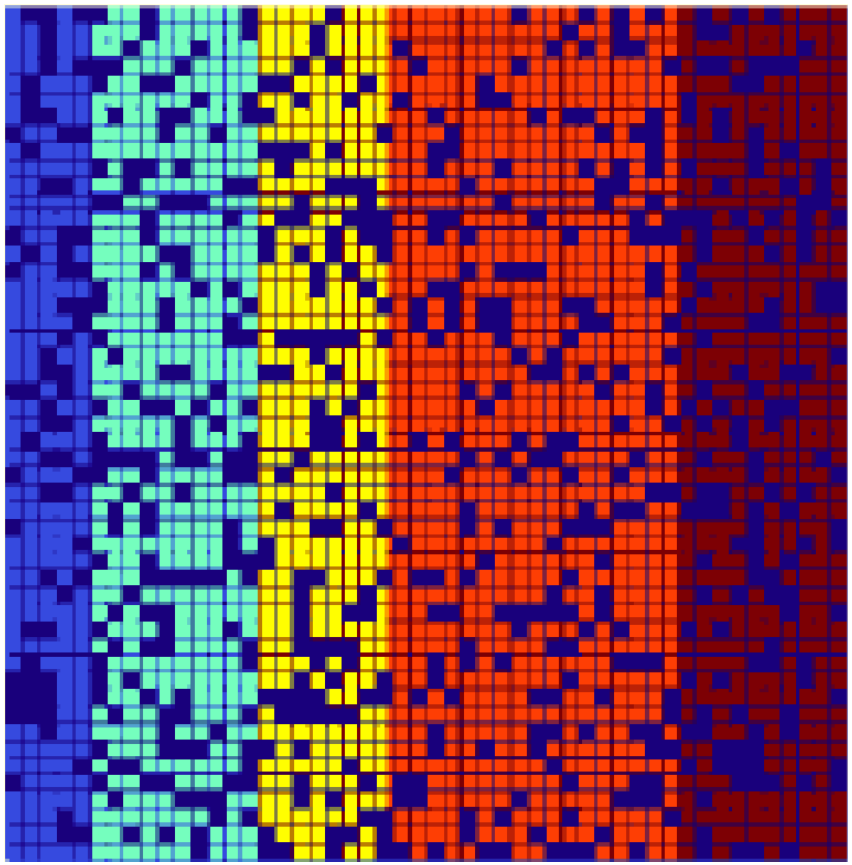
$$X = A + \epsilon$$



[1] Figures from "http://perception.csl.illinois.edu/matrix-rank/introduction.html"



[1]Figures from "http://perception.csl.illinois.edu/matrix-rank/introduction.html"



movies

users

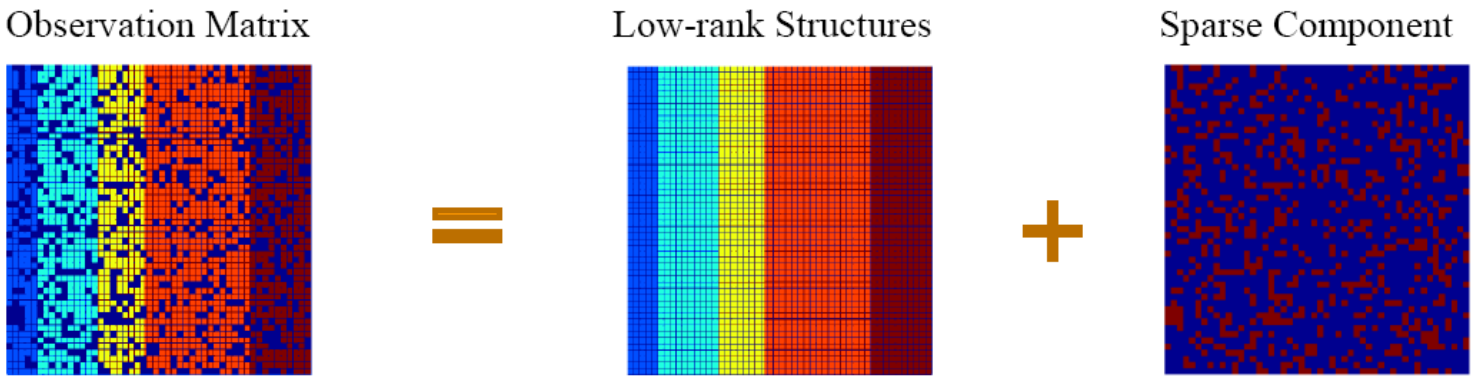
	2		1			4				5	
	5		4				?		1		3
		3		5			2				
4			?			5		3		?	
		4		1	3				5		
			2				1	?			4
	1					5		5		4	
		2		?	5		?		4		
	3		3		1		5		2		1
	3				1			2		3	
	4			5	1			3			
		3				3	?			5	
2	?		1		1						
		5			2	?		4		4	
	1		3		1	5		4		5	
1		2			4				5	?	

Robust PCA

- **Some entries of the data points can be missing or incomplete.**
- **Some entries of the data points can be corrupted.**

  (intra-sample outliers)
- Some data points can be corrupted. (sample outliers)

PCA with Corrupted Data



Some of the entries of the data points have been corrupted by gross errors:

$$X = X^0 + E$$

- Detect and correct the errors:

$$\Omega = \{(i, j) : e_{ij} \neq 0\}$$

Robust PCA by Convex Optimization

Robust PCA by Convex Optimization

The given data matrix X is generated as the sum of two matrices:

$$X = L_0 + E_0$$

- L_0: the ideal low-rank matrix (principle components)
- E_0: intra-sample outliers, which is sparse.

$$\min_{L,E} \ \text{rank}(L) + \lambda ||E||_0 \qquad \text{s.t. } X = L + E$$

1. $D \times N$ equations and $2D \times N$ unknowns

2. There're many trival cases. For example, $x_{11} = 1, x_{ij} = 0$ (X is both low-rank and sparse).

1. $D \times N$ equations and $2D \times N$ unknowns
2. There're many trival cases. For example, $x_{11} = 1, x_{ij} = 0$ (X is both low-rank and sparse).
3. Cost function: non-convex and non-differentiable.

Under certain conditions on L_0 and E_0, the correct solution to decompose $X \to (L_0, E_0)$ can be found by solving the following convex optimization problem:

$$\min_{L,E} \; ||L||_* + \lambda ||E||_1 \qquad \text{s.t. } X = L + E$$

- $||L||_* = \sum_i \sigma_i(L)$
- $||E||_1 = \sum_{i,j} |e_{ij}|$

In some cases, there're also small dense noise in the data.

$$X = L + E + Z$$

where Z is a Gaussian matrix that models small Gaussian noise in the given data.

$$\min_{L,E} \ ||L||_* + \lambda||E||_1 \qquad \text{s.t. } ||X - L - E||_2^2 \leq \epsilon^2$$

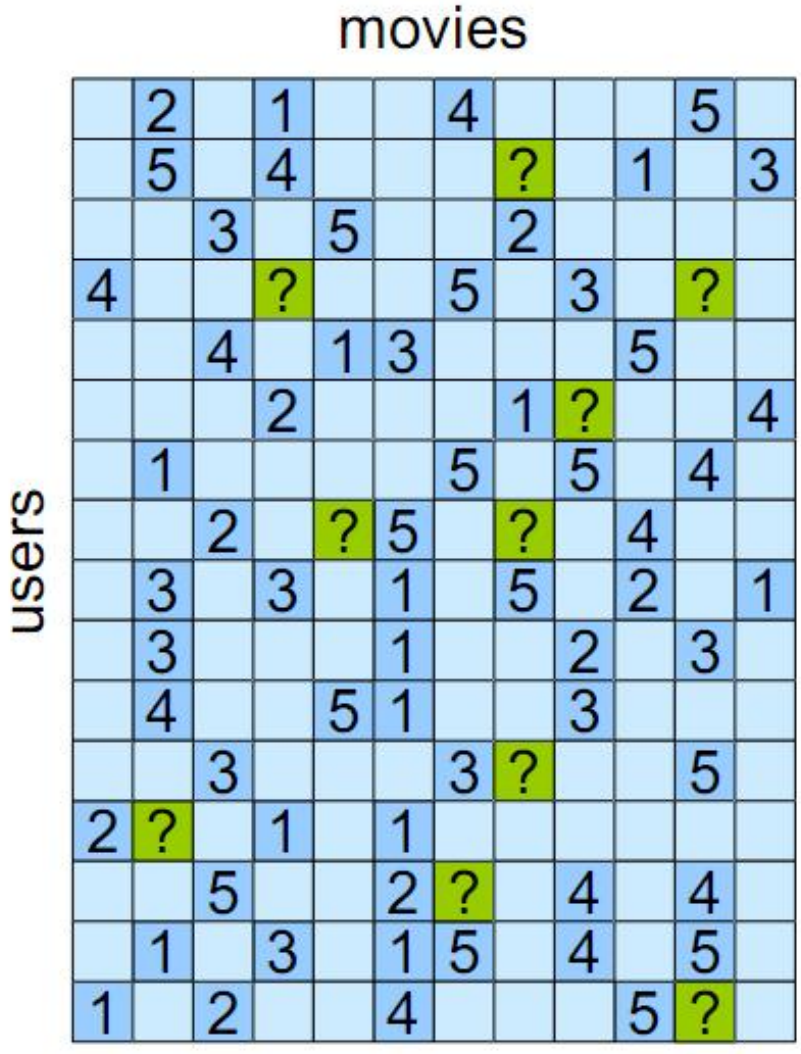
Examples [2]

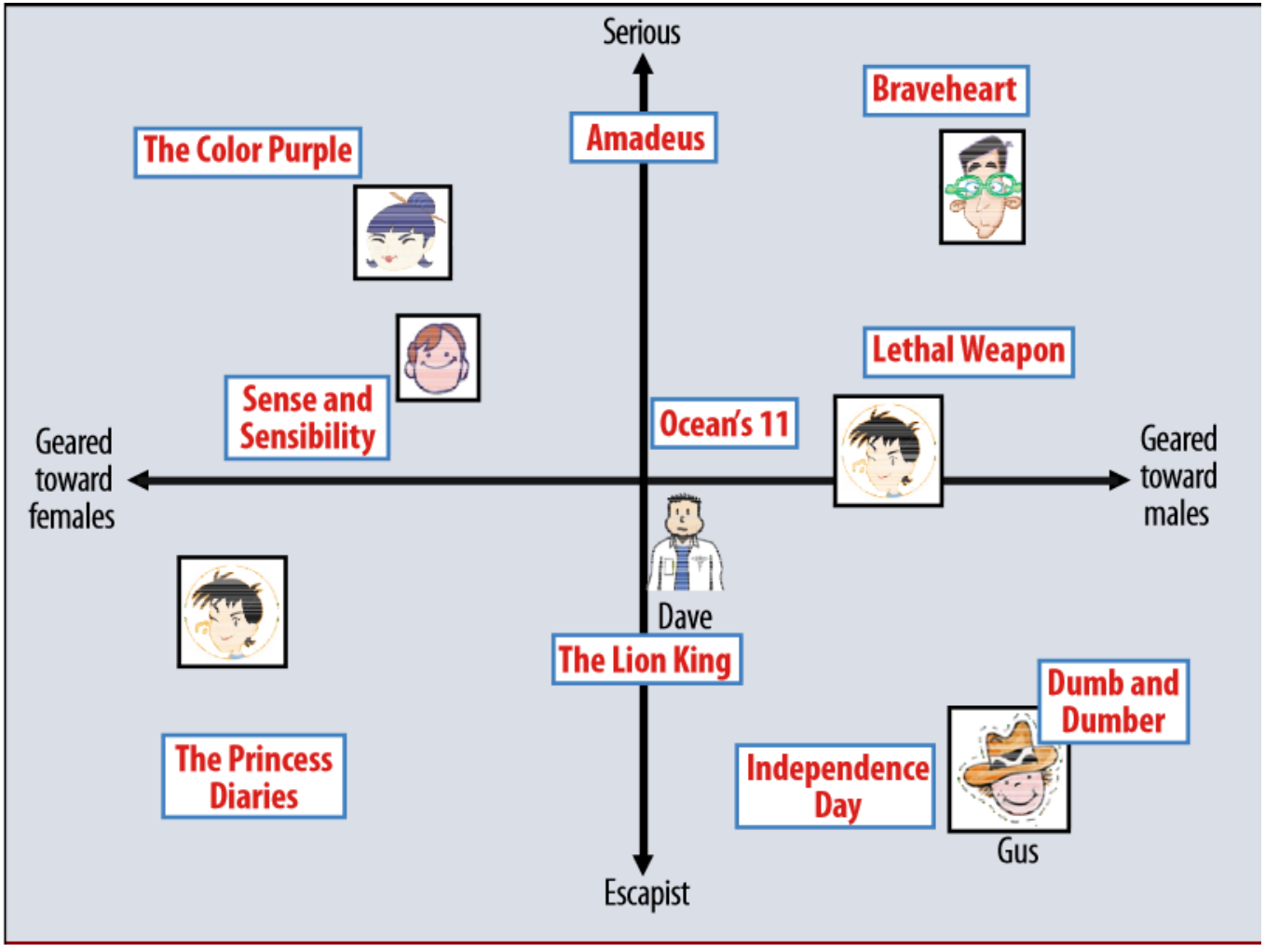
(a) Original frames (b) Low-rank $\hat{L}$ (c) Sparse $\hat{S}$

[2] Please reference "Robust Principal Component Analysis?,Emmanuel Cands,et al,2011. "

(a) Original frames (b) Low-rank $\hat{L}$ (c) Sparse $\hat{S}$

Matrix Completion




Serious
Escapist
Geared toward females
Geared toward males
Braveheart
Amadeus
The Color Purple
Lethal Weapon
Sense and Sensibility
Ocean's 11
Dave
The Lion King
Dumb and Dumber
Independence Day
Gus
The Princess Diaries

PCA with Missing Entries

Let $X \in R^{D \times N}$ be a matrix whose columns are drawn from a low-dimensional subspace of R^D of dimensions $d \ll D$. The observed subset of X is indexed by:

$$\Omega = \{(i,j) : x_{ij} \text{ is observed } \}$$

Define a map $\mathcal{P}_\Omega : R^{D \times N} \to R^{D \times N}$

$$\mathcal{P}_\Omega(X) = \begin{cases} x_{ij} & \text{if } (i,j) \in \Omega \\ 0 & \text{otherwise} \end{cases}$$

PCA with Missing Entries

Let $X \in R^{D \times N}$ be a matrix whose columns are drawn from a low-dimensional subspace of R^D of dimensions $d \ll D$. The observed subset of X is indexed by:

$$\Omega = \{(i,j) : x_{ij} \text{ is observed } \}$$

Define a map $\mathcal{P}_\Omega : R^{D \times N} \to R^{D \times N}$

$$\mathcal{P}_\Omega(X) = \begin{cases} x_{ij} & \text{if } (i,j) \in \Omega \\ 0 & \text{otherwise} \end{cases}$$

It has been shown we may complete the missing entries in X by searching for a complete matrix $A \in R^{D \times N}$ that of low-rank and coincides with X in Ω.

$$\min_A \ \text{rank}(A) \qquad \text{s.t. } \mathcal{P}_\Omega(A) = \mathcal{P}_\Omega(X)$$

Convex relaxation:

$$\min_{A} ||A||_* \qquad \text{s.t. } \mathcal{P}_\Omega(A) = \mathcal{P}_\Omega(X)$$

where $||A||_* = \sum \sigma_i(A)$ is the nuclear norm (the sum of singular values) which is the convex envelop of the rank function.

Convex relaxation:

$$\min_A ||A||_* \qquad \text{s.t. } \mathcal{P}_\Omega(A) = \mathcal{P}_\Omega(X)$$

where $||A||_* = \sum \sigma_i(A)$ is the nuclear norm (the sum of singular values) which is the convex envelop of the rank function.

When the matrix is incoherent, the locations of known entries are sampled uniformly at random, and the number of known entries is sufficiently large, the minimizer to this problem is unique and equal to the matrix X for most matrices X.

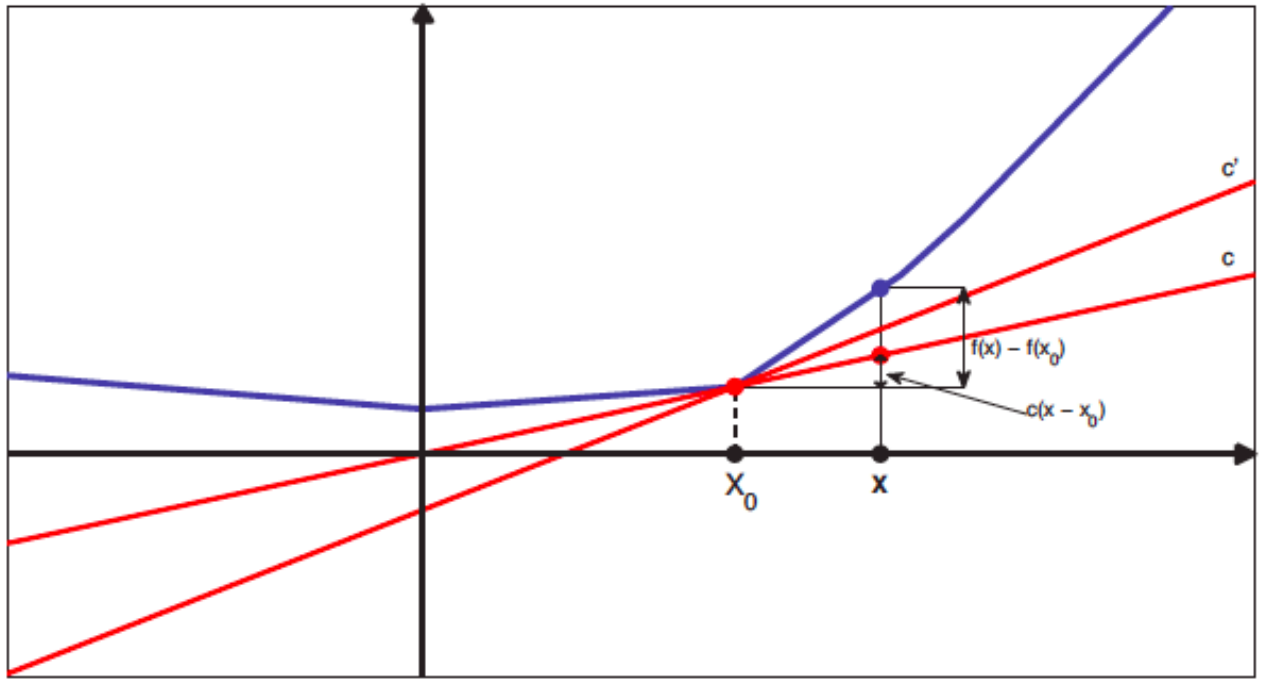
Subgradient

Definition:

Let f be a convex function. A vector g is called a subgradient of function f at point $x_0 \in \text{dom } f$ if:

$$\forall x \in \text{dom } f, f(x) \geq f(x_0) + \langle g, x - x_0 \rangle$$

And the set of all subgradients of f at x_0, denoted as $\partial f(x_0)$ is called the subdifferential set of f at x_0.


c'
c
f(x) – f(x0)
c(x – x0)
X0
x

Consider the function $f(x) = |x|, x \in R^1$.

$$\partial f(x) = \begin{cases} \{-1\} & \text{if } x < 0 \\ [-1, 1] & \text{if } x = 0 \\ \{1\} & \text{if } x > 0 \end{cases}$$

Subgradients and Optimization

We have $f(x^*) = \min_{x \in \text{dom } f} \; f(x)$ if and only if:

$$0 \in \partial f(x^*)$$

Consider a simple problem:

$$\min_{\mathbf{w}} \ ||\mathbf{y} - \mathbf{w}||^2 + \lambda||\mathbf{w}||_1 \quad \mathbf{w} \in R^p$$

- Component separability.

$$||\mathbf{y} - \mathbf{w}||_2^2 = (y_1 - w_1)^2 + \cdots + (y_n - w_n)^2$$

- Consider the n scalar minimization problem.

$$w_i^+ = \text{arg} \ \min_{w_i}(\lambda|w_i| + (y_i - w_i)^2)$$

- The subdifferential of $f(w_i) = \lambda|w_i| + (y_i - w_i)^2$ is:

$$\partial f(w_i) = \begin{cases} \{-\lambda + 2(w_i - y_i)\} & w_i < 0 \\ \{\lambda + 2(w_i - y_i)\} & w_i > 0 \\ [-\lambda + 2(w_i - y_i) \ , \ \lambda + 2(w_i - y_i)] & w_i = 0 \end{cases}$$

- The subdifferential of $f(w_i) = \lambda|w_i| + (y_i - w_i)^2$ is:

$$\partial f(w_i) = \begin{cases} \{-\lambda + 2(w_i - y_i)\} & w_i < 0 \\ \{\lambda + 2(w_i - y_i)\} & w_i > 0 \\ [-\lambda + 2(w_i - y_i)\ ,\ \lambda + 2(w_i - y_i)] & w_i = 0 \end{cases}$$

- So we can get the solution:

$$w_i^+ := S_{\frac{\lambda}{2}}(y_i) = \mathsf{sign}(y_i)\max(|y_i| - \frac{\lambda}{2}, 0)$$

where S is a soft thresholding operator:

$$S_\tau(a) = \begin{cases} a - \tau & a > \tau \\ a + \tau & a < -\tau \\ 0 & |a| < \tau \end{cases}$$

Example: Consider the problem:

$$\min_{A} \ \frac{1}{2}||X - A||_F^2 + \tau||A||_*$$

- The subdifferential w.r.t. A is given by:

$$A - X + \tau\partial||A||_*$$

Nuclear norm:

$$||A||_* = \sum_i \sigma_i(A)$$

- Convexity.
- Subdifferential.

  Suppose rank$(A) = r$, and $A = U\Sigma V^T$ is the compact SVD of A.

$$\partial||A||_* = UV^T + W$$

  where W is a matrix such that $U^TW = 0, WV = 0$ and $||W||_2 \leq 1$

Consider the problem:

$$\min_A \ \frac{1}{2}||X - A||_F^2 + \tau||A||_*$$

The solution of this problem is:

$$A = \mathcal{D}_\tau(X) = US_\tau(\Sigma)V^T$$

Algorithm – ADMM [3]

- **A**lternating **D**irection **M**ethod of **M**ultipliers
- A simple but powerful algorithm that is suited to **large-scale distributed convex optimiztion**
- Follow a decomposition-coordination procedure

[3] Please reference "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers"

ADMM is suited to this type of problems:

$$\begin{aligned} \min \quad & f(x) + g(z) \\ \text{s.t.} \quad & Ax + Bz = c \end{aligned}$$

General convex optimization problem: (f is a convex function and $\mathcal{C}$ is a convex set)

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & x \in \mathbf{C} \end{aligned}$$

Define g as an indicator function of $\mathcal{C}$, so the above problem can be rewrited as:

$$\begin{aligned} \min \quad & f(x) + g(z) \\ \text{s.t.} \quad & x - z = 0 \end{aligned}$$

ADMM Algorithm

$$\begin{aligned} \min \quad & f(x) + g(z) \\ \text{s.t.} \quad & Ax + Bz = c \end{aligned}$$

- Define the **argumented Lagrangian** as:

$$L_\rho(x, z, y) = f(x) + g(z) + y^T(Ax + Bz - c) + \frac{\rho}{2}||Ax + Bz - c||_2^2$$

- Iteration: (simultaneously update prime and dual variables)

$$\begin{aligned} x^{k+1} &:= \arg \min_x L_\rho(x, z^k, y^k) \\ z^{k+1} &:= \arg \min_z L_\rho(x^{k+1}, z, y^k) \\ y^{k+1} &:= y^k + \rho(Ax^{k+1} + Bz^{k+1} - c) \end{aligned}$$

Applications in Robust PCA

1. PCA with missing entries:

$$\min_{A} \; ||A||_* \quad \text{s.t. } \mathcal{P}_\Omega(A) = \mathcal{P}_\Omega(X)$$

where:

- Nuclear norm: $||A||_* = \sum_i \sigma_i(A)$ is the sum of singular values of A.
- Projector $\mathcal{P}_\Omega$

$$\mathcal{P}_\Omega(X) = \begin{cases} x_{ij} & \text{if } (i,j) \in \Omega \\ 0 & \text{otherwise} \end{cases}$$

The argumented Lagrangian is:

$$L_\rho(A, Y) = ||A||_* + \langle Y, \mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(A)\rangle + \frac{\rho}{2}||\mathcal{P}_\Omega(X) - \mathcal{P}_\Omega(A)||_F^2$$

Method of multipliers:

$$\begin{aligned} A^{k+1} &= \text{arg } \min_A \ L(A, Y^k) \\ Y^{k+1} &= Y_k + \rho \frac{\partial L}{\partial Z}(A^{k+1}, Y_k) \end{aligned}$$

$$A^{k+1} = \text{arg } \min_A \ L(A, Z^k) = \mathcal{D}_{\frac{1}{\rho}}(\mathcal{P}_\Omega(X) + \frac{1}{\rho}\mathcal{P}_\Omega(Y^k))$$

2. PCA with corrupted data:

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

- The argumented Lagrangian:

$$\mathcal{L}_\rho(L, E, Y) = ||L||_* + \lambda||E||_1 + \langle Y, X - L - E\rangle + \frac{\rho}{2}||X - L - E||_F^2$$

- Update L:

$$L^{k+1} = \arg \min_L \mathcal{L}_\rho(L, E^k, Y^k) = \mathcal{D}_{\frac{1}{\rho}}(X - E^k + \frac{1}{\rho}Y^k)$$

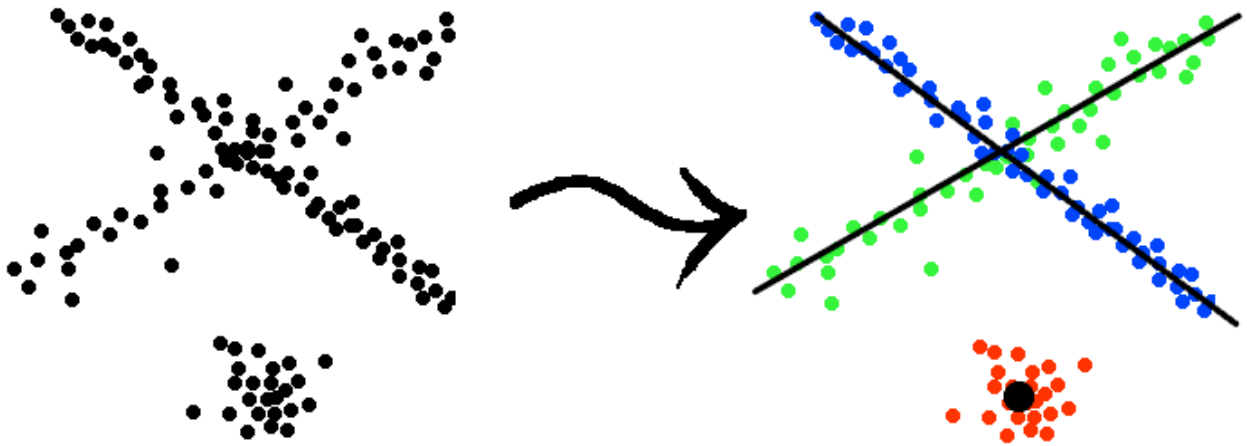
- Update E:

$$E^{k+1} = \arg \min_E \mathcal{L}_\rho(L^{k+1}, E, Y^k) = S_{\frac{\lambda}{\rho}}(X - L^{k+1} + \frac{1}{\rho}Y^k)$$

- Update Y:

$$Y^{k+1} = Y^k + \rho(X - L^{k+1} - E^{k+1})$$

Subspace Clustering



Subspace Clustering Formulation

Given a set of sample points $X = \{x_j \in R^D\}_{j=1}^N$ drawn from $n \geq 1$ distinct linear subspaces $S_i \subset R^D$ of dimensions $d_i < D$, $j = 1, \cdots, n$.

Identify each subspace S_i without knowing which sample point belong to which subspace.

1. Identifying the number of subspaces n and their dimensions $d_i = \dim(S_i)$
2. Identifying the orthonormal basis for each subspace S_i.(or equivalently a basis for its orthogonal complement $S_i^{\perp}$)
3. Clustering the N points into the subspaces to which they belong

How to define a good similarity matrix?

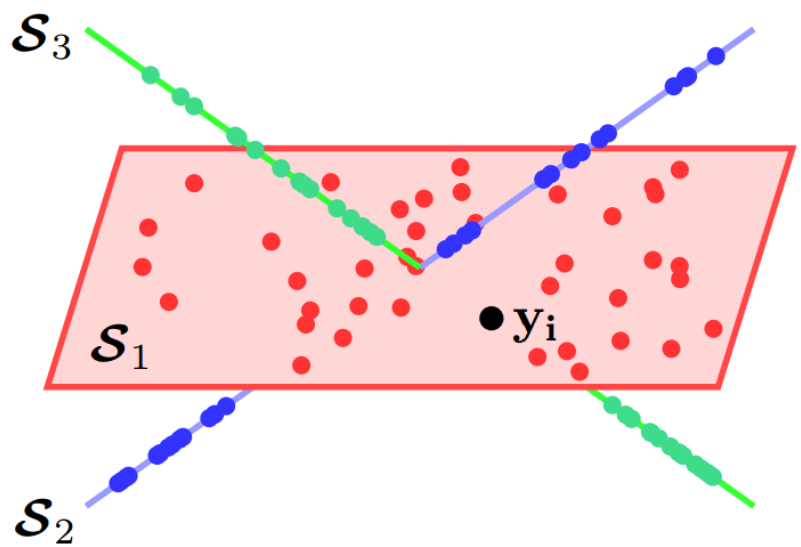
Characterize the local or global subspace structures around the points of interest

- Two points could be very close, but lie in different subspaces(e.g. near the intersection of two subspaces)
- Two points could be far from each other, but belong to the same subspace

Self-expressiveness

Each data point in a union of subspaces can be expressed as a (sparse) linear or affine combination of all other points in the dataset.

$$x_j = \sum_{i \neq j} c_{ij} x_i, \qquad \text{or} \qquad X = XC, \text{diag}(C) = 0$$



Sparse Representation Theory

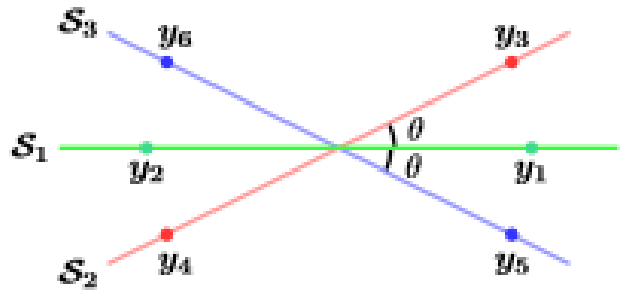
Given the following conditions:

- The subspaces are sufficiently seperated
- The data within the subspaces are well distributed

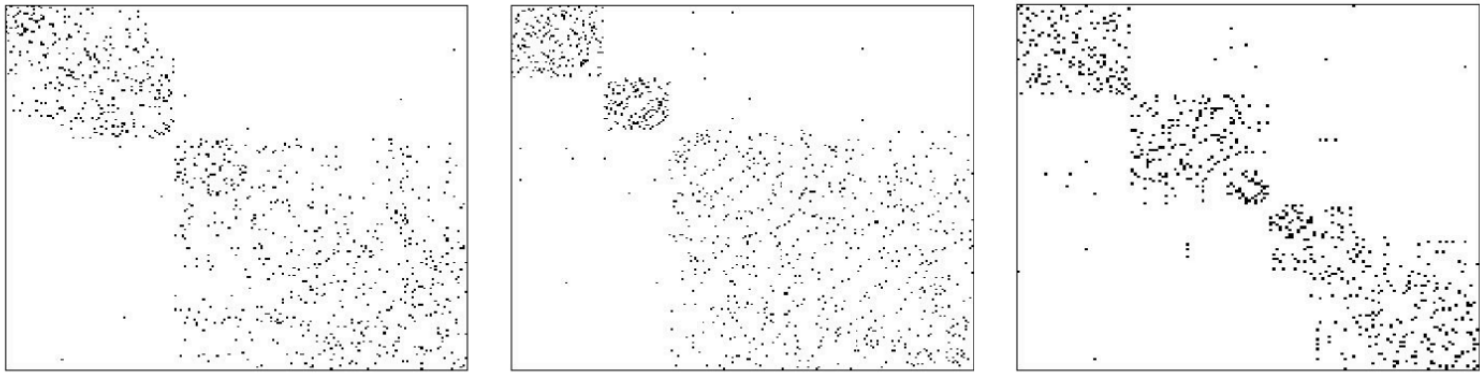
we can rescover the sparse subspace representation by the optimization problem:

$$\min_{C} ||C||_1 \qquad \text{s.t. } X = XC, \text{diag}(C) = 0$$

where $||C||_1 = \sum_{i,j=1}^{N} |c_{ij}|$



$$C_1 = \left(\begin{array}{cc|cc|cc} 0 & -1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 \end{array}\right)$$



Clustering Using Sparse Coefficients

We can use the spectral clustering algorithm.

1. Form a similarity graph $\mathcal{G} = (V, E, W)$ whose nodes are the N data points and whose edges connect points x_i and x_j with a weight $w_{ij} = |c_{ij}| + |c_{ji}|$.

$$W = |C| + |C|^T$$

2. Apply spectral clustering algorithm to the similarity graph with weights W to obtain the segmentation of the data.

Summary

- Efficient algorithms to detect low rank structure from data
- Generalized PCA model
  - Robust PCA
  - Subspace clustering