

Lecture: Dimensionality Reduction

<http://bicmr.pku.edu.cn/~wenzw/bigdata2018.html>

Acknowledgement: Some of these slides are based on Prof. Jure Leskovec's and Prof. Yinyu Ye's lecture notes

Why Reduce Dimensions?

Why reduce dimensions?

- Discover hidden correlations/topics
 - Words that occur commonly together
- Remove redundant and noisy features
 - Not all words are useful
- Interpretation and visualization
- Easier storage and processing of the data

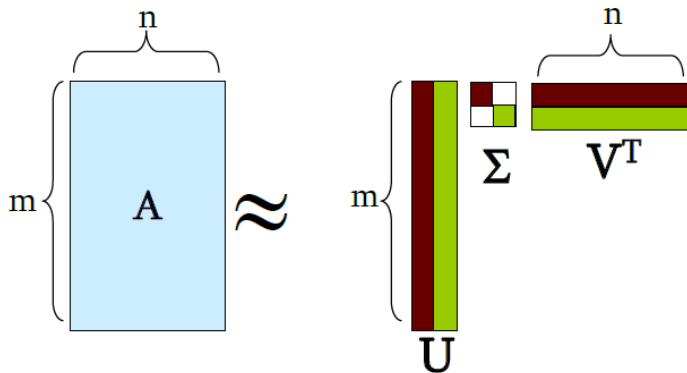
SVD - Definition

$$A_{[m \times n]} = U_{[m \times r]} \Sigma_{[r \times r]} V_{[r \times n]}^T$$

- **A**: Input data matrix
 - $m \times n$ matrix (e.g., m documents, n terms)
- **U**: Left singular vectors: $U^T U = I$
 - $m \times r$ matrix (m documents, r concepts)
- **Σ** : Singular values
 - $r \times r$ diagonal matrix (strength of each 'concept') (r : rank of the matrix A)
- **V**: Right singular vectors: $V^T V = I$
 - $n \times r$ matrix (n terms, r concepts)

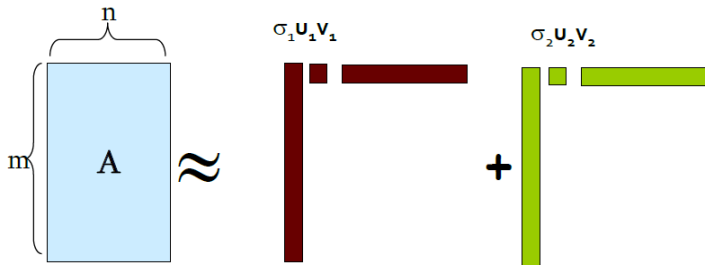
SVD

$$A \approx U\Sigma V^T = \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^T$$



SVD

$$A \approx U\Sigma V^T = \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^T$$



SVD - Properties

Theorem: SVD

If A is a real m -by- n matrix, then there exists

$$U = [u_1, \dots, u_m] \in \mathbb{R}^{m \times m} \text{ and } V = [v_1, \dots, v_n] \in \mathbb{R}^{n \times n}$$

such that $U^T U = I$, $V^T V = I$ and

$$U^T A V = \text{diag}(\sigma_1, \dots, \sigma_p) \in \mathbb{R}^{m \times n}, \quad p = \min(m, n),$$

where $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p \geq 0$.

- **Proof:** Let $V_1 \in \mathbb{R}^{n \times r}$ has orthonormal columns, then exists $V_2 \in \mathbb{R}^{n \times (n-r)}$ such that $V = [V_1, V_2]$ is orthogonal.
- Let $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ be unit 2-norm vectors: $Ax = \sigma y$ with $\sigma = \|A\|_2$. Then exists $V_2 \in \mathbb{R}^{n \times (n-1)}$ and $U_2 \in \mathbb{R}^{m \times (m-1)}$ so $V = [x, V_2] \in \mathbb{R}^{n \times n}$ and $U = [y, U_2] \in \mathbb{R}^{m \times m}$ are orthogonal.

- Then it can be proved that U^TAV has the following structure

$$U^TAV = \begin{pmatrix} \sigma & w^T \\ 0 & B \end{pmatrix} \equiv A_1.$$

Since

$$\left\| A_1 \begin{pmatrix} \sigma \\ w \end{pmatrix} \right\|_2^2 \geq (\sigma^2 + w^T w)^2,$$

we have $\|A_1\|_2^2 \geq (\sigma^2 + w^T w)$. But $\sigma^2 = \|A\|_2^2 = \|A_1\|_2^2$, and so we must have $w = 0$. An induction gives the proof.

Properties:

- $AV = U\Sigma$, $A^TU = V\Sigma^T$: $Av_i = \sigma u_i$, $A^Tu_i = \sigma_i v_i$, $i = 1, \dots, p$.
- $\text{rank}(A) = r$, $\text{null}(A) = \text{span}\{v_{r+1}, \dots, v_n\}$, $\text{ran}(A) = \text{span}\{u_1, \dots, u_r\}$
- $A = \sum_{i=1}^r \sigma_i u_i v_i^T$
- $\|A\|_F^2 = \sigma_1^2 + \dots + \sigma_p^2$, $\|A\|_2 = \sigma_1$

SVD - Best Low Rank Approximation

Theorem

Let the SVD of $A \in \mathbb{R}^{m \times n}$ be given in Theorem: SVD. If $k < r = \text{rank}(A)$ and $A_k = \sum_{i=1}^k \sigma_i u_i v_i^T$, then

$$\min_{\text{rank}(B)=k} \|A - B\|_2 = \|A - A_k\|_2 = \sigma_{k+1}.$$

- Proof: Since $U^T A_k V = \text{diag}(\sigma_1, \dots, \sigma_k, 0, \dots, 0)$ it follows that $\text{rank}(A_k) = k$ and $U^T(A - A_k)V = \text{diag}(0, \dots, 0, \sigma_{k+1}, \dots, \sigma_p)$. Hence $\|A - A_k\|_2 = \sigma_{k+1}$.
- Suppose $\text{rank}(B) = k$ for some $B \in \mathbb{R}^{m \times n}$. We can find orthonormal vectors x_1, \dots, x_{n-k} so $\text{null}(B) = \text{span}\{x_1, \dots, x_{n-k}\}$. A dimension argument shows:

$$\text{span}\{x_1, \dots, x_{n-k}\} \cap \text{span}\{v_1, \dots, v_{k+1}\} \neq \{0\}$$

- Let z be a unit 2-norm vector in this intersection. Since $Bz = 0$ and

$$Az = \sum_{i=1}^{k+1} \sigma_i (v_i^T z) u_i,$$

we have

$$\|A - B\|_2^2 \geq \|(A - B)z\|_2^2 = \|Az\|_2^2 = \sum_{i=1}^{k+1} \sigma_i^2 (v_i^T z)^2 \geq \sigma_{k+1}^2$$

Comments:

- So zeroing small σ_i introduces less error
- How many σ s to keep? Rule of thumb: keep 80-90% of 'energy' ($= \sum \sigma_i^2$)

SVD - Complexity

- To compute SVD: $O(nm^2)$ or $O(n^2m)$
- But:
 - Less work, if we just want singular values
 - or if we want first k singular vectors
 - or if the matrix is sparse
- Implemented in linear algebra packages like
 - Dense matrix: LAPACK
 - Sparse Matrix: ARPACK, PROPACK
 - High Level Software packages: Matlab, SPlus, Mathematica ...

Relation to Eigen-decomposition

- SVD gives us $A = U\Sigma V^T$
- Eigen-decomposition: $A = X\Lambda X^T$
 - A is symmetric
 - U, V, X are orthonormal
 - Λ, Σ are diagonal
- $AA^T = U\Sigma\Sigma^T U^T$
- $A^T A = V\Sigma\Sigma^T V^T$
- $\lambda_i(A^T A) = \sigma_i^2(A)$

- 1 Principal Component Analysis (PCA)
- 2 Maximum variance unfolding
- 3 Graph Realization and Sensor Network Localization
- 4 Matrix Factorization

Dimensionality Reduction

- Assume we have a dataset represented in an $p \times N$ matrix \mathbf{X} consisting of N data vectors \mathbf{x}_i with dimensionality p .
 \mathbf{x}_i is the i th column of \mathbf{X} .
- Assume further that this dataset has intrinsic dimensionality q (often $q \ll p$).
- Dimensionality reduction techniques transform dataset \mathbf{X} with dimensionality p into a new dataset \mathbf{Y} with dimensionality q , while retaining the geometry of the data as much as possible.

- Assume $p < N$. For a collection of data $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N \in \mathbb{R}^p$, we want to represent them by a rank q linear model

$$\mathbf{y}(\mu, \lambda, \mathbf{M}) = \mu + \mathbf{M}\lambda, \mathbf{M} \in \mathbb{R}^{p \times q}, \forall q \leq p,$$

where μ is local vector and $\mathbf{M}^\top \mathbf{M} = I$.

- We write $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N] \in \mathbb{R}^{p \times N}$. The new representation of the data is:

$$\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_N] \in \mathbb{R}^{p \times N}, \quad \mathbf{y}_i = \mu + \mathbf{M}\lambda_i.$$

- Consider the model

$$\min_{\mu, \lambda_i, \mathbf{M}} F(\mu, \lambda_i, \mathbf{M}) = \sum_{i=1}^N \|\mathbf{x}_i - \mu - \mathbf{M}\lambda_i\|_2^2, \quad \text{s.t. } \mathbf{M}^\top \mathbf{M} = I.$$

- Let $\bar{\mathbf{x}} = \frac{1}{N} \sum_i \mathbf{x}_i$ and $\bar{\lambda} = \frac{1}{N} \sum_i \lambda_i$.
- $\frac{\partial F}{\partial \mu} = 0 \implies \mu = \bar{\mathbf{x}} - \mathbf{M}\bar{\lambda}$ (1)
- $\frac{\partial F}{\partial \lambda_i} = 0 \implies \lambda_i = \mathbf{M}^\top (\mathbf{x}_i - \mu) \implies \bar{\lambda} = \mathbf{M}^\top (\bar{\mathbf{x}} - \mu)$ (2)
- Hence, we must have $\mu = \bar{\mathbf{x}}$ and have $\lambda_i = \mathbf{M}^\top (\mathbf{x}_i - \bar{\mathbf{x}})$.
- After partially optimizing λ and μ , the original problem reduce to

$$\min_{\mathbf{M}} F(\mathbf{M}) = \sum_{i=1}^N \|(\mathbf{x}_i - \bar{\mathbf{x}}) - \mathbf{M}\mathbf{M}^\top (\mathbf{x}_i - \bar{\mathbf{x}})\|_2^2, \text{ s.t. } \mathbf{M}^\top \mathbf{M} = I.$$

- Let $\bar{\mathbf{x}}_i = \mathbf{x}_i - \bar{\mathbf{x}}$ and $\bar{\mathbf{X}} = [\bar{\mathbf{x}}_1, \bar{\mathbf{x}}_2, \dots, \bar{\mathbf{x}}_N]$. In fact,

$$\bar{\mathbf{X}} = \mathbf{X} - \frac{1}{N} \mathbf{X} \mathbf{1} \mathbf{1}^\top.$$

Then, we have

$$\begin{aligned} F(\mathbf{M}) &= \sum_{i=1}^N \|\bar{\mathbf{x}}_i - \mathbf{M} \mathbf{M}^\top \bar{\mathbf{x}}_i\|_2^2 \\ &= \|\bar{\mathbf{X}} - \mathbf{M} \mathbf{M}^\top \bar{\mathbf{X}}\|_F^2 \\ &= \text{Tr}(\bar{\mathbf{X}}^\top \bar{\mathbf{X}}) - \text{Tr}(\bar{\mathbf{X}}^\top \mathbf{M} \mathbf{M}^\top \bar{\mathbf{X}}) \\ &= \text{Tr}(\bar{\mathbf{X}} \bar{\mathbf{X}}^\top) - \text{Tr}(\mathbf{M}^\top \bar{\mathbf{X}} \bar{\mathbf{X}}^\top \mathbf{M}). \end{aligned}$$

- Let $\text{cov}(\mathbf{X}) = \bar{\mathbf{X}} \bar{\mathbf{X}}^\top$. Thus the problem is equivalent to

$$\max_{\mathbf{M} \in \mathbb{R}^{p \times q}} \text{Tr}(\mathbf{M}^\top \text{cov}(\mathbf{X}) \mathbf{M}), \text{ s.t. } \mathbf{M}^\top \mathbf{M} = \mathbf{I}.$$

- PCA constructs a low-dimensional representation of the data that describes as much of the variance in the data as possible.
- Let $x \in \mathbb{R}^p$ be a random vector with mean μ and covariance Σ .
- Find $y = \sum_{i=1}^p z_i \mathbf{x}_i = z^T x$ such that the variance of y is maximized.

$$\begin{aligned} \text{Var}(y) &= E[(z^T x - E(z^T x))(z^T x - E(z^T x))] \\ &= E(z^T x - z^T E x)(z^T x - z^T E x)^T \\ &= z^T E[(x - E x)(x - E x)^T] z = z^T \text{cov}(\mathbf{X}) z \end{aligned}$$

- Model problem:

$$\max z^T \text{cov}(\mathbf{X}) z, \quad \text{s.t. } \|z\|_2 = 1.$$

Find the maximal eigenpair (λ_1, v_1) of $\text{cov}(\mathbf{X})$.

Principal Component Analysis (PCA)

- find a linear mapping \mathbf{M} that maximizes data variance

$$\begin{aligned} \max_{\mathbf{M}} \quad & \text{Tr}(\mathbf{M}^\top \text{cov}(\mathbf{X})\mathbf{M}) \\ \text{s.t.} \quad & \mathbf{M}^\top \mathbf{M} = \mathbf{I} \end{aligned}$$

- Lagrangian function:

$$L(\mathbf{M}, \Lambda) = \text{Tr}(\mathbf{M}^\top \text{cov}(\mathbf{X})\mathbf{M}) - \langle \Lambda, \mathbf{M}^\top \mathbf{M} - \mathbf{I} \rangle$$

- Stationary point (KKT condition) at

$$\begin{aligned} \text{cov}(\mathbf{X})\mathbf{M} &= \mathbf{M}\Lambda \\ \mathbf{M}^\top \mathbf{M} &= \mathbf{I} \end{aligned}$$

- Thus PCA essentially requires eigenvalue decomposition

- Let $\bar{\mathbf{X}} = \mathbf{X} - \frac{1}{N}\mathbf{X}\mathbf{1}\mathbf{1}^\top$, where $\mathbf{1}$ is a column vector of all ones
- SVD: $\bar{\mathbf{X}} = U\Sigma V^\top$, $\Sigma = [\text{diag}(\sigma_1, \dots, \sigma_p), 0]$, $U^\top U = I$, $V^\top V = I$.
Eigen: $\bar{\mathbf{X}}\bar{\mathbf{X}}^\top = U\Lambda U^\top$, $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_p)$, and $\lambda_i = \sigma_i^2$
- PCA computes

$$\max_{\mathbf{M}} \text{Tr}(\mathbf{M}^\top \text{cov}(\mathbf{X})\mathbf{M}), \text{ s.t. } \mathbf{M}^\top \mathbf{M} = I.$$

Then the optimal solution $\mathbf{M} = U_q$.

- Therefore, we have: $\mathbf{y}_i = \mu + \mathbf{M}\mathbf{M}^\top(\mathbf{x}_i - \bar{\mathbf{x}})$, i.e.,

$$\mathbf{Y} = \mu\mathbf{1}^\top + \mathbf{M}\mathbf{M}^\top\bar{\mathbf{X}} = \mu\mathbf{1}^\top + \mathbf{M}\Sigma_q V_q^\top.$$

- The matrix $\mathbf{M}^\top\bar{\mathbf{X}} = \Sigma_q V_q^\top \in R^{q \times N}$ is called the first q principal components of $\bar{\mathbf{X}}$.

Classical MDS

- Known data points $\mathbf{x}_i \in \mathbb{R}^p$, $i = 1, \dots, N$.
compute pairwise distance matrix $D(\mathbf{X})$ with

$$d_{ij}(\mathbf{X}) = \|\mathbf{x}_i - \mathbf{x}_j\|_2$$

- Let $D_2(\mathbf{X}) = (d_{ij}^2(\mathbf{X}))$. MDS: Find $\mathbf{y}_i \in \mathbb{R}^q$ such that

$$\min_{\mathbf{Y}} \|HD_2(\mathbf{X})H - HD_2(\mathbf{Y})H\|_F^2$$

Lemma

Let $D_2(\mathbf{X}) = (d_{ij}^2(\mathbf{X}))$ of $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N)$, $H = I_N - \frac{1}{N}\mathbf{1}\mathbf{1}^T$. Let $B = -\frac{1}{2}HD_2(\mathbf{X})H$ and $\bar{\mathbf{X}} = \mathbf{X} - \frac{1}{N}\mathbf{X}\mathbf{1}\mathbf{1}^T$. Then we have $B = \bar{\mathbf{X}}^T \bar{\mathbf{X}}$.

Classical MDS

- Proof: Define $K = \mathbf{X}^\top \mathbf{X}$, $K_{ij} = \mathbf{x}_i^\top \mathbf{x}_j$. Then

$$D_2(\mathbf{X})_{ij} = (\mathbf{x}_i - \mathbf{x}_j)^\top (\mathbf{x}_i - \mathbf{x}_j) = \mathbf{x}_i^\top \mathbf{x}_i + \mathbf{x}_j^\top \mathbf{x}_j - 2\mathbf{x}_i^\top \mathbf{x}_j.$$

Define $w = \text{diag}(K)$. Then

$$D_2(\mathbf{X}) = w\mathbf{1}^\top + \mathbf{1}w^\top - 2K.$$

Thus

$$B = -\frac{1}{2}H(w\mathbf{1}^\top + \mathbf{1}w^\top - 2K)H = H\mathbf{X}^\top \mathbf{X}H = \bar{\mathbf{X}}^\top \bar{\mathbf{X}},$$

since $H\mathbf{1}w^\top = \mathbf{1}w^\top - \frac{1^\top \mathbf{1}}{N}\mathbf{1}w^\top = 0$ and $\mathbf{1}w^\top H = 0$.

Classical MDS

- Let $B = \bar{\mathbf{X}}^\top \bar{\mathbf{X}}$ and SVD: $\bar{\mathbf{X}} = U\Sigma V^\top$, $\Sigma = [\text{diag}(\sigma_1, \dots, \sigma_p), 0]$.
The eigenvalue decomposition: $B = V\Lambda V^\top$, $\lambda_i = \sigma_i^2$.
- Assume that \mathbf{Y} is centered, i.e., $-\frac{1}{2}HD_2(Y)H = Y^\top Y$, then
Classical MDS is equivalent to

$$\min_{\mathbf{Y}} \|\mathbf{B} - \mathbf{Y}^\top \mathbf{Y}\|_F^2$$

- The optimal solution $\mathbf{Y} = \Lambda^{\frac{1}{2}} V^\top$
- Classical MDS: take $\tilde{\mathbf{X}} = \Lambda_q^{\frac{1}{2}} V_q^\top = \Sigma_q V_q^\top$
- PCA and classical MDS are equivalent!

Extension of MDS

- Known data points $\mathbf{x}_i \in \mathbb{R}^p$, $i = 1, \dots, N$.
Given norms $\|\cdot\|_x$, $\|\cdot\|_y$, compute pairwise distance matrix

$$d_{ij}(\mathbf{X}) = \|\mathbf{x}_i - \mathbf{x}_j\|_x, \quad d_{ij}(\mathbf{Y}) = \|\mathbf{y}_i - \mathbf{y}_j\|_y$$

- Find $\mathbf{y}_i \in \mathbb{R}^q$ such that

$$\min_{\mathbf{Y}} \|D_x(\mathbf{X}) - D_y(\mathbf{Y})\|_F^2$$

or with centering matrix $H = I_N - \frac{1}{N}11^T$ and $D_{2x}(\mathbf{X}) = (d_{ij}^2(\mathbf{X}))$:

$$\min_{\mathbf{Y}} \|HD_{2x}(\mathbf{X})H - HD_{2y}(\mathbf{Y})H\|_F^2$$

- Kernel PCA: $K_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$

$$\begin{aligned}\|\mathbf{x}_i - \mathbf{x}_j\|_x^2 &= \|\mathbf{x}_i\|_x^2 + \|\mathbf{x}_j\|_x^2 - 2\langle \mathbf{x}_i, \mathbf{x}_j \rangle_x \\ (D_x(\mathbf{X}))_{ij}^2 &= K_{ii} + K_{jj} - 2K_{ij}\end{aligned}$$

- Isomap**: geodesic distance for \mathbf{X} and F-norm for \mathbf{Y}

Outline

- 1 Principal Component Analysis (PCA)
- 2 Maximum variance unfolding**
- 3 Graph Realization and Sensor Network Localization
- 4 Matrix Factorization

Maximum variance unfolding

- PCA and MDS are linear dimensionality reduction methods that compute mappings to preserve Euclidean distances between all pairs of data points
- Based on the notion of *isometry*, maximum variance unfolding (MVU) considers the much larger class of nonlinear transformations that only preserve the geometric properties of local neighborhoods
- We say that the data sets are ***k*-locally isometric** if for every point x_i , there exists a rotation and translation that maps \mathbf{x}_i and its k nearest neighbors $\{\mathbf{x}_{j1}, \mathbf{x}_{j2}, \dots, \mathbf{x}_{jk}\}$ precisely onto the points \mathbf{y}_i and $\{\mathbf{y}_{j1}, \mathbf{y}_{j2}, \dots, \mathbf{y}_{jk}\}$
- The notion of isometry can be translated into various sets of equality constraints of inputs $\{\mathbf{x}_i\}_{i=1}^n$ and outputs $\{\mathbf{y}_i\}_{i=1}^n$

MVU - constraints

- Inputs $\{\mathbf{x}_i\}_{i=1}^N$ and outputs $\{\mathbf{y}_i\}_{i=1}^N$ are locally isometric if whenever \mathbf{x}_i and \mathbf{x}_j are themselves neighbors or common neighbors of another point in the data set, we have:

$$\|\mathbf{y}_i - \mathbf{y}_j\|^2 = \|\mathbf{x}_i - \mathbf{x}_j\|^2$$

- We also constrain the outputs \mathbf{y}_i to be centered on the origin:

$$\sum_i \mathbf{y}_i = 0$$

this simply removes a translational degree of freedom from the final solution

MVU - objective function

- Any "fold" between two points on a manifold serves to decrease the Euclidean distance between the points
- This suggests an optimization that we can perform to compute the outputs \mathbf{y}_i that unfold a manifold sampled by inputs \mathbf{x}_i .
- Maximize the sum of pairwise squared distances between outputs:

$$\Phi = \frac{1}{2N} \|\mathbf{y}_i - \mathbf{y}_j\|^2$$

- We pull the outputs as far apart as possible, subject to the constraints in the previous slide
- It can be verified that the objective function is indeed bounded
- Intuitively, the constraints to preserve local distances prevent a divergence to pull the outputs infinitely far apart

MVU - optimization

- Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denotes the graph formed by pairwise connecting each input to all of its k -nearest neighbors
- Then, in terms of the squared distance matrix $D_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|^2$, the optimization can be written as:

$$\begin{aligned} \max \quad & \sum_{ij} \|\mathbf{y}_i - \mathbf{y}_j\|^2 \\ \text{s.t.} \quad & \sum_i \mathbf{y}_i = 0, \\ & \|\mathbf{y}_i - \mathbf{y}_j\|^2 = D_{ij}, \forall (i, j) \in \mathcal{E} \end{aligned}$$

- This problem is not convex, as it involves maximizing a quadratic form subject to quadratic equality constraints

MVU - reformulation

- The inner product matrix $K_{ij} = \mathbf{y}_i \cdot \mathbf{y}_j$ determines the outputs up to rotation

- Expanding the square in the distance constraint we obtain

$$K_{ii} - 2K_{ij} + K_{jj} = D_{ij}$$

- Likewise, the centering constraint can be expressed as

$$0 = \left\| \sum_i \mathbf{y}_i \right\|^2 = \sum_{ij} \mathbf{y}_i \cdot \mathbf{y}_j = \sum_{ij} K_{ij}$$

- Both are now linear equality constraints on the elements of K
- We may view our original problem as an optimization over inner product matrices K_{ij} rather than vectors \mathbf{y}_i
- Only symmetric matrices with nonnegative eigenvalues can be interpreted as inner product matrices, therefore

$$K = K^T \succeq 0$$

MVU - reformulation

- For the objective function:

$$\begin{aligned}\Phi &= \frac{1}{2N} \sum_{ij} (\|\mathbf{y}_i\|^2 + \|\mathbf{y}_j\|^2 + 2\mathbf{y}_i \cdot \mathbf{y}_j) \\ &= \sum_i \|\mathbf{y}_i\|^2 \\ &= \sum_i K_{ii} \\ &= \text{Tr}(K).\end{aligned}$$

- We will obtain a low dimensional embedding by maximizing the trace of the inner product matrix

MVU - reformulation

- We rewrite the MVU problem as:

$$\begin{aligned} \max \quad & \text{Tr}(K) \\ \text{s.t.} \quad & K = K^\top \succeq 0, \\ & \mathbf{1}^\top K \mathbf{1} = 0, \\ & K_{ii} - 2K_{ij} + K_{jj} = D_{ij}, \forall (i,j) \in \mathcal{E} \end{aligned}$$

- This is a semidefinite program
 - The domain is the cone of psd matrices intersected with hyperplanes (equality constraints)
 - The objective function is bounded above and linear
 - The problem is guaranteed to be feasible because the constraints are trivially satisfied by the Gram matrix $G_{ij} = \mathbf{x}_i \cdot \mathbf{x}_j$
- The reformulation into SDP not only allows global and efficient solution of the MVU problem, but also gives the extra capability of estimating the intrinsic dimension

MVU - reformulation

- From the inner product matrix K^* solved by this SDP, the outputs \mathbf{y}_i can be recovered by applying MDS on K^*
- Let $V_{\alpha i}$ denote the i th element of the α th eigenvector, with eigenvalue λ_α
- Then the inner product matrix can be written as

$$K_{ij}^* = \sum_{\alpha=1}^N \lambda_\alpha V_{\alpha i} V_{\alpha j}$$

- The embedding is obtained by identifying the α th element of the output \mathbf{y}_i as

$$Y_{\alpha i} = \sqrt{\lambda_\alpha} V_{\alpha i}$$

- A low dimensional embedding that is approximately locally isometric is given by truncating the elements of \mathbf{y}_i

The dual MVU problem

- Examining the dual of an optimization problem often gives further insight and offers theoretical and computational advantages
- The MVU problem is no exception
- For notational convenience, write the last set of equality constraints as

$$\text{Tr}(KE^{\{i,j\}}) = D_{ij}, \quad (i,j) \in \mathcal{E}$$

where the $N \times N$ matrix $E^{\{i,j\}}$ has only four nonzero elements:

$$E_{ii}^{\{i,j\}} = E_{jj}^{\{i,j\}} = 1, E_{ij}^{\{i,j\}} = E_{ji}^{\{i,j\}} = -1$$

- In forming the Lagrangian, we associate the dual variables $Z = Z^T \succeq 0$, $\nu \in \mathbb{R}$ and W_{ij} for $\forall (i,j) \in \mathcal{E}$

The dual MVU problem

- The Lagrangian:

$$\begin{aligned}L(K, Z, \nu, W) &= \text{Tr}(K) + \text{Tr}(KZ) - \nu \mathbf{1}^\top K \mathbf{1} \\ &\quad - \sum_{(i,j) \in \mathcal{E}} W_{ij} (\text{Tr}(KE^{\{i,j\}}) - D_{ij}) \\ &= \text{Tr}[K(I + Z - \nu \mathbf{1} \mathbf{1}^\top - \sum_{(i,j) \in \mathcal{E}} W_{ij} E^{\{i,j\}})] \\ &\quad + \sum_{(i,j) \in \mathcal{E}} D_{ij} W_{ij}\end{aligned}$$

- The dual function is obtained as

$$\begin{aligned}g(Z, \nu, W) &= \sup_{K=K^\top} L(K, Z, \nu, W) \\ &= \begin{cases} \sum_{(i,j) \in \mathcal{E}} D_{ij} W_{ij} & \text{if } I + Z - \nu \mathbf{1} \mathbf{1}^\top - \sum_{(i,j) \in \mathcal{E}} W_{ij} E^{\{i,j\}} = 0 \\ +\infty & \text{otherwise} \end{cases}\end{aligned}$$

The dual MVU problem

- Eliminating Z from the equality, the feasibility condition in the above equation becomes

$$I - \nu \mathbf{1}\mathbf{1}^\top - L \preceq 0, \quad L = \sum_{(i,j) \in \mathcal{E}} W_{ij} E^{\{i,j\}}$$

- Note that L is a weighted Laplacian of the graph \mathcal{G} . The above linear matrix inequality is equivalent to

$$\nu \geq \frac{1}{N}, \quad \lambda_{N-1}(L) \geq 1$$

where λ_{N-1} denotes the second smallest eigenvalue of a symmetric matrix (Here $\lambda_N(L) = 0$ with associated eigenvector $\mathbf{1}$)

The dual MVU problem

- The dual MVU problem is:

$$\begin{aligned} \min \quad & \sum_{(i,j) \in \mathcal{E}} D_{ij} W_{ij} \\ \text{s.t.} \quad & \lambda_{N-1}(L) \geq 1, \\ & L = \sum_{(i,j) \in \mathcal{E}} W_{ij} E^{\{i,j\}} \end{aligned}$$

- This is a convex optimization problem because the function $\lambda_{N-1}(L)$ is concave under the implicit constraint $\lambda_N(L) = 0$
- Note that the dual variable ν does not appear in the problem

Duality

- The following duality results hold for the primal MVU problem and the dual MVU problem
 - **Weak duality:** For any primal feasible K and any dual feasible W , we have

$$\text{Tr}(K) \leq \sum_{i,j} D_{ij} W_{ij}$$

(Note that $W_{ij} = 0$ if $(i,j) \notin \mathcal{E}$) This can be seen by checking the duality gap

- **Strong duality:** There exist a primal-dual feasible pair (K^*, W^*) with zero duality gap, i.e.

$$\text{Tr}(K^*) = \sum_{i,j} D_{ij} W_{ij}^*$$

Strong duality follows from Slater's condition for constraint qualification

Optimality conditions

- A pair (K^*, W^*) is primal-dual optimal iff they satisfy the following KKT conditions
 - primal feasibility

$$K^* \succeq 0, \quad \mathbf{1}^\top K^* \mathbf{1} = 0$$
$$K_{ii}^* - 2K_{ij}^* + K_{jj}^* = D_{ij}, \quad \forall (i, j) \in \mathcal{E}$$

- dual feasibility

$$L^* = \sum_{(i,j) \in \mathcal{E}} W_{ij}^* E^{\{i,j\}}, \quad \lambda_{N-1}(L^*) \geq 1$$

- complementary slackness

$$L^* K^* = K^* \quad (\text{why?})$$

Optimality conditions

- Note that we always have $\lambda_{N-1}(L^*) \geq 1$, thus the complementary slackness condition $L^*K^* = K^*$ means that the range of K^* lies in the eigenspace (e.s.) of L^* associated with λ_{N-1}
- Since K^* is a dense Gram matrix while L^* is a sparse weighted Laplacian, $L^*K^* = K^*$ means precisely

top e.s. of dense $K^* \subseteq$ bottom e.s. of sparse L^*

where "bottom e.s." means the eigenspace associated with λ_{n-1} (discard the eigenvector $\mathbf{1}$ and $\lambda_N = 0$)

- Another direct consequence is

$$r \leq \text{rank}(K^*) \leq \text{multiplicity of } \lambda_{N-1}(L^*)$$

where r is the dimension of the low dimensional representations obtained by performing MDS on K^*

Outline

- 1 Principal Component Analysis (PCA)
- 2 Maximum variance unfolding
- 3 Graph Realization and Sensor Network Localization**
- 4 Matrix Factorization

Graph Realization and Sensor Network Localization Problems

- **Input:** m known points (anchors) $a_k \in \mathbb{R}^2, k = 1, \dots, m$, and n unknown points (sensors or targets) $x_j \in \mathbb{R}^2, j = 1, \dots, n$. For some pair of two points, we have a Euclidean distance measure d_{kj} between a_k and x_j , or distance measure d_{ij} between x_i and x_j
- **Output:** Position estimation for all unknown points, and confidence measures on reliability of each position estimation
- **Objective:** Robust, fast and accurate

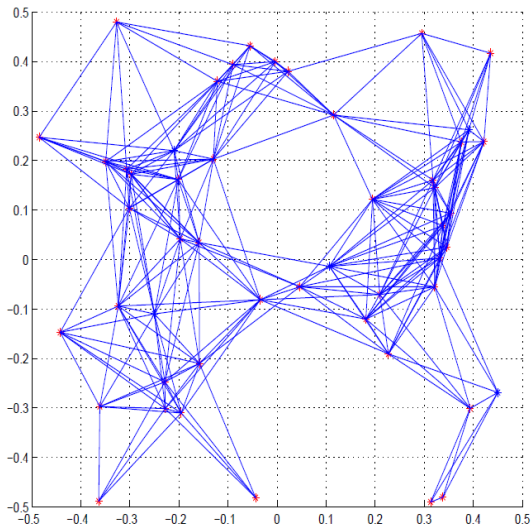


Figure: 50-Sensor Network with Radio Range .3

Euclidean Distance Geometry Model

$$\|x_i - x_j\|^2 = d_{ij}^2, \forall (i, j) \in \mathcal{N}_x, i < j,$$

$$\|a_k - x_j\|^2 = d_{kj}^2, \forall (k, j) \in \mathcal{N}_a$$

$d_{ij}^2(d_{kj}^2)$ connects x_i to x_j (a_k to x_j) with an edge whose length is $d_{ij}(d_{kj})$

- Does the system has a localization or realization of all x_j s?
- Is the localization unique?
- Is the localization reliable or trustworthy?
- Is the system partially localizable?

Global and Nonlinear Least Squares

$$\min \sum_{i,j \in \mathcal{N}_x} (\|x_i - x_j\|^2 - d_{ij}^2)^2 + \sum_{k,j \in \mathcal{N}_a} (\|a_k - x_j\|^2 - d_{kj}^2)^2$$
$$\min \sum_{i,j \in \mathcal{N}_x} (\|x_i - x_j\| - d_{ij})^2 + \sum_{k,j \in \mathcal{N}_a} (\|a_k - x_j\| - d_{kj})^2$$

For example, Moré and Wu (1997)

Matrix representation

Let $X = [x_1, x_2, \dots, x_n]$ be the $2 \times n$ matrix that needs to be determined. Then

$$\begin{aligned}\|x_i - x_j\|^2 &= (e_i - e_j)^\top X^\top X (e_i - e_j), \\ \|a_k - x_j\|^2 &= (a_k; -e_j)^\top [I \quad X]^\top [I \quad X] (a_k; -e_j)\end{aligned}$$

where e_j is the vector of all zero except 1 at the j th position

$$\begin{aligned}(e_i - e_j)^\top Y (e_i - e_j) &= d_{ij}^2, \forall i, j \in \mathcal{N}_x, i < j, \\ (a_k; -e_j)^\top \begin{pmatrix} I & X \\ X^\top & Y \end{pmatrix} (a_k; -e_j) &= d_{kj}^2, \forall k, j \in \mathcal{N}_a, \\ Y &= X^\top X\end{aligned}$$

SDP relaxation and analysis

- Change $Y = X^\top X$ to $Y \succeq X^\top X$, which is equivalent to (e.g., Boyd and Vandenberghe 2005)

$$Z = \begin{pmatrix} I & X \\ X^\top & Y \end{pmatrix} \succeq 0$$

Find a symmetric matrix $Z \in \mathbb{R}^{(2+n) \times (2+n)}$ such that

$$Z_{1:2,1:2} = I,$$

$$(\mathbf{0}; (e_i - e_j))(\mathbf{0}; (e_i - e_j))^\top \bullet Z = d_{ij}^2, \forall (i, j) \in \mathcal{N}_x, i < j,$$

$$(a_k; -e_j)(a_k; -e_j)^\top \bullet Z = d_{kj}^2, \forall (k, j) \in \mathcal{N}_a,$$

$$Z \succeq 0.$$

- Any matrix solution for the SDP relaxation has rank at least 2
- If every sensor point is connected, directly or indirectly, to an anchor point, then the solution set must be bounded

Outline

- 1 Principal Component Analysis (PCA)
- 2 Maximum variance unfolding
- 3 Graph Realization and Sensor Network Localization
- 4 Matrix Factorization**

K-均值聚类

聚类(clustering)不同于分类(classification), 在聚类问题中我们仅仅知道数据点本身, 而不知道每个数据点具体的标签。聚类分析的任务就是将一些无标签的数据点按照某种相似度来进行归类, 进而从数据点本身来学习其内蕴的类别特征。

给定 p 维空间中 n 个数据点 a_1, a_2, \dots, a_n , 聚类问题就是要寻找 k 个不相交的非空集合 S_1, S_2, \dots, S_k , 使得

$$\{a_1, a_2, \dots, a_n\} = S_1 \cup S_2 \cup \dots \cup S_k,$$

并且使得组内距离平方和最小, 即

$$\begin{aligned} \min_{S_1, S_2, \dots, S_k} \quad & \sum_{i=1}^k \sum_{a \in S_i} \|a - c_i\|^2, \\ \text{s.t.} \quad & S_1 \cup S_2 \cup \dots \cup S_k = \{a_1, a_2, \dots, a_n\}, \\ & S_i \cap S_j = \emptyset, \quad \forall i \neq j, \end{aligned} \tag{1}$$

K-均值聚类的等价表述一

为了表示聚类方式 S_1, S_2, \dots, S_k , 一个很自然的想法是使用一个向量 $\phi_i \in \mathbb{R}^k$ 来表示点 a_i 所处的类别. 具体地, 定义 ϕ_i 为

$$(\phi_i)_j = \begin{cases} 1, & a_i \in S_j, \\ 0, & a_i \notin S_j, \end{cases}$$

则聚类问题可以等价描述为

$$\begin{aligned} \min_{\Phi, H} \quad & \|A - \Phi H\|_F^2, \\ \text{s.t.} \quad & \Phi \in \mathbb{R}^{n \times k}, \text{ 每一行只有一个元素为1, 其余为0,} \\ & H \in \mathbb{R}^{k \times p}. \end{aligned} \quad (2)$$

在这里 Φ 的第 i 行的向量就是 ϕ_i^T .

K-均值聚类的等价表述二

首先定义 $\mathbf{1}_{S_t}$, $1 \leq t \leq k$ 为 n 维空间中每个分量取值0或1的向量, 且

$$\mathbf{1}_{S_t}(i) = \begin{cases} 1, & a_i \in S_t, \\ 0, & a_i \notin S_t. \end{cases}$$

可以证明, 第 t 类 S_t 中每个点到其中心点的距离平方和可以写成 $\frac{1}{2n_t} \text{tr}(D\mathbf{1}_{S_t}\mathbf{1}_{S_t}^T)$, 其中 $D \in \mathbb{R}^{n \times n}$ 的元素为 $D_{ij} = \|a_i - a_j\|^2$.

因此, 我们将问题(1)转化为

$$\begin{aligned} & \min_{S_1, S_2, \dots, S_k} \quad \frac{1}{2} \text{tr}(DX), \\ & \text{s.t.} \quad X = \sum_{t=1}^k \frac{1}{n_t} \mathbf{1}_{S_t} \mathbf{1}_{S_t}^T, \\ & \quad S_1 \cup S_2 \cup \dots \cup S_k = \{a_1, a_2, \dots, a_n\}, \\ & \quad S_i \cap S_j = \emptyset, \quad \forall i \neq j. \end{aligned} \tag{3}$$

K-均值聚类的等价表述二

对半定矩阵 X 进行分解 $X = YY^T$, $Y \in \mathbb{R}^{n \times k}$, 我们可以进一步得到如下矩阵优化问题 (这里 $\mathbf{1}$ 是 n 维向量且分量全为1) :

$$\begin{aligned} \min_{Y \in \mathbb{R}^{n \times k}} \quad & \text{tr}(Y^T D Y), \\ \text{s.t.} \quad & Y Y^T \mathbf{1} = \mathbf{1}, \\ & Y^T Y = I_k, Y \geq 0. \end{aligned} \tag{4}$$

- $S_+^{n,k} := \{X \in \mathbb{R}^{n \times k} : X^\top X = I_k, X \geq 0\}$
- Orthogonal NMF (ONMF), Ding, Li, Peng & Park (2006)
 - Data matrix $A \in \mathbb{R}_+^{n \times r}$, n data samples, each with r features, k clusters
 - $A \approx XY^\top$, $X \in S_+^{n,k}$, $Y \in \mathbb{R}_+^{r \times k}$
 - $X_{ij} \begin{cases} > 0, & \text{if sample } i \in \text{cluster } j \\ = 0, & \text{otherwise} \end{cases}$
 - ONMF model

$$\min_{X \in S_+^{n,k}, Y \in \mathbb{R}_+^{r \times k}} \|A - XY^\top\|_F^2 \quad (5)$$

- Orthonormal projective NMF (OPNMF) model, Yang & Oja (2010)

$$\min_{X \in S_+^{n,k}} \|A - XX^\top A\|_F^2 \quad (6)$$

- K-indicators model, Chen, Yang, Xu, Zhang & Zhang (2019)

$$\min_{X \in S_+^{n,k}, Y^\top Y = I} \|UY - X\|_F^2 \quad \text{s.t.} \quad \|X_{i,:}\|_0 = 1, i \in [n], \quad (7)$$

where U is the features matrix extracted from the data matrix A .

Spectral clustering

We have n data points (x_1, \dots, x_n) and wish to partition them into k disjoint clusters C_1, \dots, C_k

- 1 Form affinity matrix $A \in \mathbb{R}^{n \times n}$ defined by

$$A_{ij} = \begin{cases} \exp(-\frac{\|x_i - x_j\|^2}{2\sigma^2}) & \text{if } i \neq j \\ 0 & \text{if } i = j \end{cases}$$

- 2 Define D to be the diagonal matrix with $D_{ii} = \sum_j A_{ij}$, i.e. the sum of A 's i th row, and construct the matrix

$$L = D^{-1/2} A D^{-1/2}$$

- 3 Find the eigenvectors v_j of L corresponding to the k largest eigenvalues, and form the matrix $V = [v_1 v_2 \dots v_k] \in \mathbb{R}^{n \times k}$
- 4 Form the matrix Y from V by renormalizing each of V 's rows to have unit length
- 5 Treating each row of Y as a point in \mathbb{R}^k , cluster them into k clusters via k -means (or any other clustering algorithm)

Spectral clustering

- Use methods from spectral graph partitioning to do clustering
- Needed: pairwise distances between data points
- Can be thought of as weights of links in a graph: clustering problem becomes a graph partitioning problem
- Unlike k-means, clusters need not be convex
- Matrix factorization form: $A = DX$ with unknown D and X , solve for sparse X and $X_i = 0$ or 1

Subspace clustering

- Subspace clustering: self-representation of data

$$\begin{aligned} \min_{\mathbf{X}, \mathbf{E}} \quad & \text{rank}(\mathbf{X}) + \lambda \|\mathbf{E}\|_0 \\ \text{s.t.} \quad & \mathbf{A} = \mathbf{A}\mathbf{X} + \mathbf{E} \end{aligned}$$

- Convex relaxation

$$\begin{aligned} \min_{\mathbf{X}, \mathbf{E}} \quad & \|\mathbf{X}\|_* + \lambda \|\mathbf{E}\|_1 \\ \text{s.t.} \quad & \mathbf{A} = \mathbf{A}\mathbf{X} + \mathbf{E} \end{aligned}$$

- Essentially: $A = AX$ with unknown X , solve for sparse/other conditions on X

Graph matching

- $A = XBXT^T$ with unknown X , B solve for X as a permutation

$$\min_X \|A - XBXT^T\|_F^2, \text{ s.t. } X \text{ is a permutation}$$

Non-Negative matrix factorization (NMF)

- Non-negative matrix factorization (NMF): $A = DX$ with unknown D and X , solve for elements of $D, X > 0$

$$\begin{aligned} \min_{\mathbf{D}, \mathbf{X}} \quad & \|\mathbf{A} - \mathbf{DX}\|^2 \\ \text{s.t.} \quad & d_{ij} \geq 0 \\ & x_{ij} \geq 0 \end{aligned}$$

- Nonconvex problem; alternating minimization

Matrix completion

- Matrix completion (MC) as matrix factorization: $A = M \circ X$ with a known mask M ; solve for X with lowest rank possible

$$\begin{aligned} \min_{\mathbf{X}} \quad & \text{rank}(\mathbf{X}) \\ \text{s.t.} \quad & \mathbf{A} = \mathbf{M} \circ \mathbf{X} \end{aligned}$$

- Convex relaxation

$$\begin{aligned} \min_{\mathbf{X}} \quad & \|\mathbf{X}\|_* \\ \text{s.t.} \quad & \mathbf{A} = \mathbf{M} \circ \mathbf{X} \end{aligned}$$

Robust PCA

- Robust principal component analysis (RPCA)

$$\begin{aligned} \min_{\mathbf{X}, \mathbf{E}} \quad & \text{rank}(\mathbf{X}) + \lambda \|\mathbf{E}\|_0 \\ \text{s.t.} \quad & \mathbf{A} = \mathbf{X} + \mathbf{E} \end{aligned}$$

- Matrix factorization view: $A = X + E$ with a low rank component X and a sparse component E
- Convex relaxation

$$\begin{aligned} \min_{\mathbf{X}, \mathbf{E}} \quad & \|\mathbf{X}\|_* + \lambda \|\mathbf{E}\|_1 \\ \text{s.t.} \quad & \mathbf{A} = \mathbf{X} + \mathbf{E} \end{aligned}$$

Stable Principle Component Pursuit (SPCP)/ Noisy Robust PCA

- RPCA is limited to the low-rank component being exactly low-rank and the sparse component being exactly sparse
- In real world applications the observations are often corrupted by noise
- Introducing entry-wise noise N : $A = X + E + N$ with X, E, N unknown, solve for X low rank, E sparse
- Optimization problem

$$\begin{aligned} \min_{\mathbf{X}, \mathbf{E}} \quad & \|\mathbf{X}\|_* + \lambda \|\mathbf{E}\|_1 \\ \text{s.t.} \quad & \|\mathbf{A} - \mathbf{X} + \mathbf{E}\|_F \leq \delta \end{aligned}$$

Sparse PCA

- $A = DX$ with unknown D and X , solve for sparse D
- Sparsity in number of basis vectors
- Optimization form

$$\begin{aligned} \min_{\mathbf{D}, \mathbf{X}} \quad & \|\mathbf{A} - \mathbf{DX}\|_F^2 + \lambda \|\mathbf{D}\|_1 \\ \text{s.t.} \quad & \|\mathbf{x}_i\|_2 \leq 1 \end{aligned}$$

Dictionary learning

- $A = DX$ with unknown D and X , solve for sparse X
- Sparsity in representation coefficients
- Optimization form

$$\begin{aligned} \min_{\mathbf{D}, \mathbf{X}} \quad & \|\mathbf{A} - \mathbf{DX}\|_F^2 + \lambda \|\mathbf{X}\|_1 \\ \text{s.t.} \quad & \|\mathbf{d}_i\|_2 \leq 1 \end{aligned}$$

Matrix Compressive Sensing (MCS)

- Find a rank- r matrix L such that $A(L) = b$ or $A(L + S) = b$
- Optimization form

$$\begin{aligned} \min_{\mathbf{X}} \quad & \|\mathbf{X}\|_* \\ \text{s.t.} \quad & \mathcal{A}(\mathbf{X}) = \mathbf{b} \end{aligned}$$

Other applications

- $Y = A X$ with unknown X and rows of X are sparse
- Compressive sensing is a special case of MMV called Single Measurement Vector (SMV)
- $Y = A X$ with unknown X and rows of X are sparse, X is one column
- $Y = A X$ with unknown A and X and statistical independence between columns of X or subspaces of columns of X

Summary

- Matrix factorization: $\mathbf{A} = \mathbf{D}\mathbf{X}$
- Utilizing different structural properties on \mathbf{D} and/or \mathbf{X}
 - Low-rank factorization: \mathbf{D} and \mathbf{X} have few columns/rows
 - Dictionary Learning / Sparse PCA: either \mathbf{D} or \mathbf{X} has few non-zeros
 - Clustering: sparse binary \mathbf{X}
 - nonnegative matrix factorization: element-wise positivity
 - ...
- Algorithms
 - Convex relaxation: (A)PG, ALM, ADMM
 - Jointly nonconvex problems: alternating minimization, BCD
 - ...