Experiments on Spectral Clustering

Yu-Hsuan Huang
asd00012334@hotmail.com

November 2019

Abstract

In this writeup, we first briefly introduce the spectral clustering, then discuss the details we take to implements the clustering. Then, based on experiment results, we give some comment on the parameter settings.

1 Problem Description

In this assignment, one is given a distance matrix $D \in \mathbb{R}^{n\times n}$ from some unknown $X \in \mathbb{R}^{n\times 2}$ that,

$$D_{ij} := \left(\left\|X_i - X_j\right\|\right)_{1\le i,j\le n},$$

and asked to find one feasible X with columns centered at 0, i.e.

$$\sum_{1\le i\le n} X_i = 0.$$

Then, one is asked to do the 2/3/4 spectral clustering to X with weights specified as,

$$W_{i,j} = 1 - \frac{D_{ij} - \min_{i'\neq j'}(D_{i'j'})}{\max_{i'\neq j'} D_{i'j'} - \min_{i'j'} D_{i'j'}}, \text{ for } i \neq j.$$

2 Finding position matrix

We note that the defining equation of D is shift invariant, that being said, we could assume existence of some X with $X_1 = 0$. And solve for

$$G_{ij} := \left(X^T X\right)_{ij} = \frac{D_{i1}^2 + D_{j1}^2 - D_{ij}^2}{2}.$$

Then we unitarily diagonalize G, for $Q \in \mathsf{SO}(n), \Lambda \in \mathsf{diag}(n)$,

$$G = Q\Lambda Q^T = (\sqrt{\Lambda}Q^T)^T(\sqrt{\Lambda}Q^T).$$

Let $\tilde{X} = \sqrt{\Lambda}Q^T$ and we have $G = \tilde{X}^T\tilde{X}$. Since $\tilde{X}\tilde{X}^T = X^TX$, they admit the same diagonalization. Thus $\tilde{X}, X$ admits the same right singular vector, for $U_1, U_2, V \in \mathsf{SO}(n)$ and $\Sigma \in \mathsf{diag}(n)$,

$$\tilde{X} = U_1\Sigma V,$$
$$X = U_2\Sigma V.$$

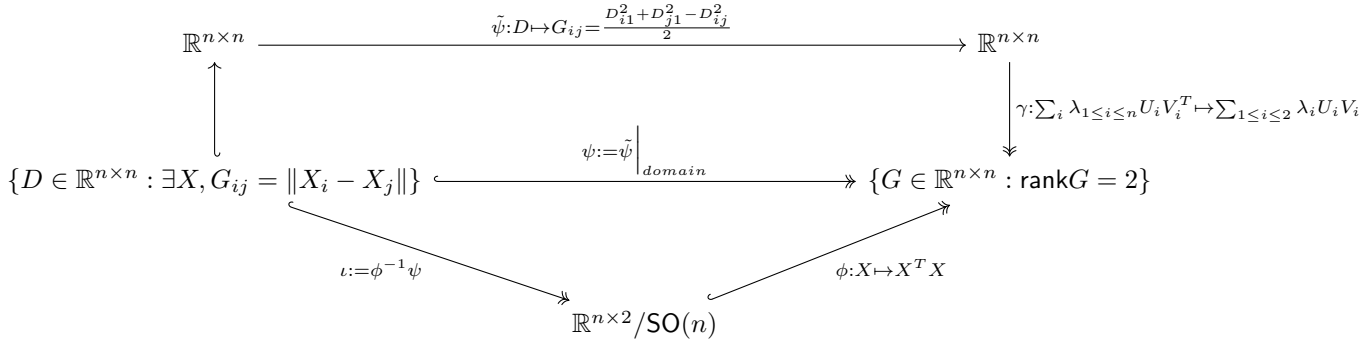
Thus $\tilde{X} = U_1 U_2^\dagger X$ differ by only a unitary and thus share the same distance matrix. This would yield a natural isomorphism,

$$\mathbb{R}^{n\times n}/\mathsf{SO(n)} \xrightarrow{\sim} \left\{A \in \mathbb{R}^{n\times n} : A \succcurlyeq 0\right\},$$

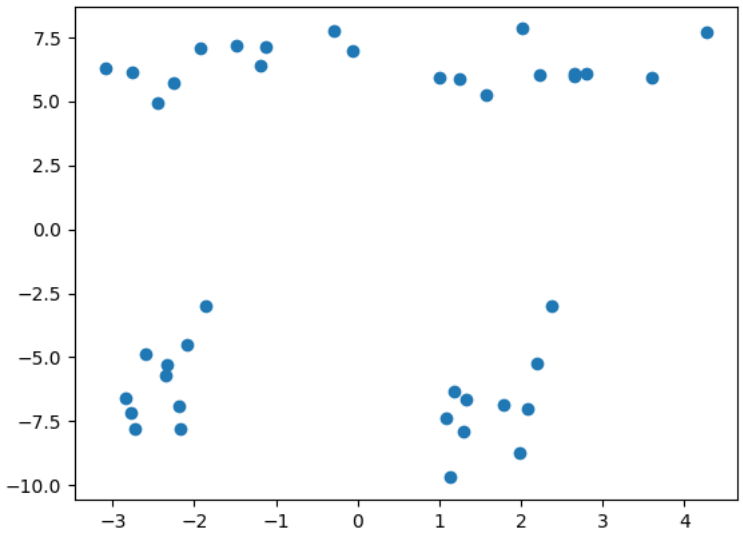
by $X \mapsto X^T X$. The left hand side admits the usual multiplication and the right hand side performs product by,

$$A_1 * A_2 := \sqrt{A_2}^T A_1 \sqrt{A_2},$$

with $\sqrt{A_2}^T \sqrt{A_2} = A_2$. Finally, by shift invariance, we could recenter columns of $\tilde{X}$ back to 0. Ideally G should be rank 2. However, due to floating error, we might obtain the perturbated $\tilde{G} = G + \epsilon$. Luckily, due to Eckar-Young's theorem, picking the eigenvectors corresponding to the largest eigenvalues would be the optimal choice. This would yield a natural embedding defined by $\phi^{-1}\gamma\tilde{\psi}$ according to the following commutative diagram.



As result, we obtain the following scattering of $\{X_i\}_{1\le i\le n} \subseteq \mathbb{R}^2$.



3 Clustering

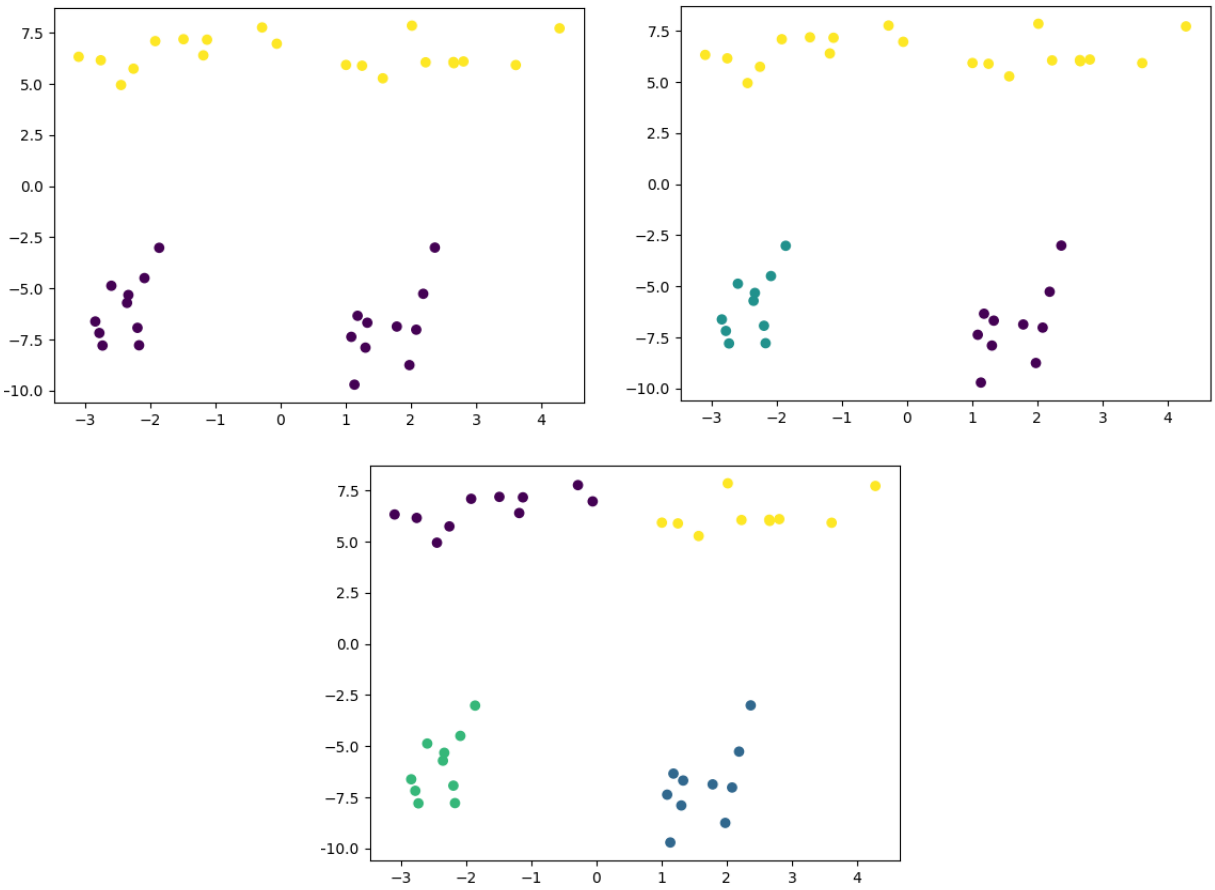
Since the weight matrix, $W \geq 0$ would induce a complete graph, we simply consider the normalized Laplacian matrix $L \in \mathbb{R}^{n \times n}$,

$$L_{ij} := 1 - \left(\sum_{\ell \neq i} W_{i\ell} \right)^{-1/2} W_{ij} \left(\sum_{\ell \neq j} W_{\ell j} \right)^{-1/2}.$$

When doing the k clustering, we do the orthgonal diagonalization,

$$L = \sum_{1 \leq i \leq n} \lambda_i Q_i Q_i^T$$

and take k eigenvectors $\{Q_1, \ldots, Q_k\}$ with largest eigenvalues. Doing k means on the rows of $(Q_1 \ldots Q_k)$ would be the spectral clustering. For k= 2, 3, or 4, we have the following scattering.



4 Disscussion and Extra Experiments

From the case where $k = 3$, one could see that, unlike naive k-means, the spectral clustering depend not only on the nearest cluster centroid, but also the "shape" of the data points. In this case, since the data points were embedded into $\mathbb{R}^2$, we have the complete topology, but different weights assigned to edges would capture geometric information of the embedding. Intuitively speaking, an edge with positve weight attracts

its adjacent vertices and an edge with negative weight repells them. Connected vertices with larger edge weights are more likely to be clustered into same class. That is the reason that when we set up the weight matrix, we revert its sign then renormalize it.

What happen if we change the number r of chosen eigenvectors? In some sense it appears to be an underfitting/overfitting issue. Here we fix the cluster number to 3 and change r=1, 2, 10, or 20, from left to right, top to down. As we can see, when $r < 3$, it could still perform clustering but lose track of shapes. In another word, it fall back to something like naive k means. When $r >> 3$, the boundary of different cluster becomes sharpy and the largest cluster eventually took over the all data points.