Automatic Selection of the Scale Parmeter of the Gaussian Kernel in Spectral Clustering

Jisu Kim : Master degree candidate, Department of Statistics, Pusan National University (first5454@pusan.ac.kr)
Choongrak Kim : Professor, Department of Statistics, Pusan National University

Introduction

- Spectral clustering solves the clustering problem as the way of graph partitioning problem based on the spectral graph theory using the graph Laplacian.
- Gassian kernel is widely used to construct the weighted adjacency matrix for the graph Laplacian especially when a given dataset lives in Euclidean space.
- The global scale parameter fails to cluster the points from data which is complex and has different scales. Because if σ is fixed, the adjacency only depends on the distance.
- To handle with this problem, Zelnik-Manor and Perona (2005) proposed self-tuning spectral clustering which set the σ_i is the distance from $\boldsymbol{x_i}$ to 7^{th}nearest point from $\boldsymbol{x_i}$.
- We propose an automatic selection algorithm to estimate the locally weighted scale parameter from the neighbors.
- From the compared results with the previous suggested methods; self-tuning spectral clustering (Zelnik-Manor et al., 2005) and fast density-aware spectral clustering (Christopher et al., 2020), the proposed method outperforms the others with respect to the CER(clustering error rate).

Spectral clustering

- The most common spectral clustering algorithms are the unnormalized spectral clustering using the graph Laplacian $\boldsymbol{L}$ suggested by Hagen and Kahng (1992) and two versions of the normalized spectral clustering suggested by Shi and Malik (2000) using the graph Laplacian $\boldsymbol{L_{rw}}$ and Ng, Jordan and Weiss (2002) using the graph Laplacian $\boldsymbol{L_{sym}}$.
- The normalized spectral clustering using $\boldsymbol{L_{rw}}$.
  1. A given data $\boldsymbol{x_1}, \dots, \boldsymbol{x_n}, \boldsymbol{x_i} \in \mathbb{R}^p$.Number of clusters k.
  2. Using the Gaussian kernel, we could get the weighted adjacency matrix $\boldsymbol{A} \in \mathbb{R}^{nxn}$ as

$$a_{ij} = exp\left\{-\frac{d^2(\boldsymbol{x}_i, \boldsymbol{x}_j)}{\sigma^2}\right\}, i, j = 1, ..., n$$

     where $d(\boldsymbol{x_i}, \boldsymbol{x_j})$ is the Euclidean distance of $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$ and σ is the scale parameter.
  3. The degree matrix is the diagonal matrix $\boldsymbol{D} \in \mathbb{R}^{nxn}$ with the element $d_i = \sum_{j=1}^{n} a_{ij}$ and let $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$.
  4. Compute the graph Laplacians $\boldsymbol{L_{rw}} = \boldsymbol{D}^{-1}\boldsymbol{L}$
  5. Compute the first k generalized eigenvectors $\boldsymbol{u_1}, \dots, \boldsymbol{u_k}$ of the generalized eigenproblem $\boldsymbol{L_{rw}u} = \lambda\boldsymbol{Du}$.
  6. Let $\boldsymbol{y_i}$ be the i^{th} row of matrix $\boldsymbol{U} = (\boldsymbol{u_1}, \dots, \boldsymbol{u_k})$ and cluster the points $\boldsymbol{y_i}, i = 1, \dots, n$ into k clusters with k-means.

The proposed method

- The proposed algorithm finds c_i which maxmizes the mean difference of distance from $\boldsymbol{x_i}$ to the others points.

Algorithm 1: The proposed algorithm

Input: A given data $\boldsymbol{x_1}, \dots, \boldsymbol{x_n}, \boldsymbol{x_i} \in \mathbb{R}^p$, $n^* = n/10$, the Euclidean distance $d_{i,j} = d(\boldsymbol{x_i}, \boldsymbol{x_j})$

step 1. Let $d_{i,(1)} \leq \cdots \leq d_{i,(n-1)}$ the order statistics

step 2. Compute

$$m_k = \frac{1}{n^* - k}\sum_{j=k+1}^{n^*} d_{i,(j)} - \frac{1}{k}\sum_{j=1}^{k} d_{i,(j)} \text{ for } k = 3, \dots, n^*$$

step 3. $c_i = argmax_k m_k$ and $\sigma_i = d_{i,(c_i)}$

step 4. find $c_1, \dots, c_n$ repeating step1 - step3

Output: c_i and σ_i for all data points $\boldsymbol{x_1}, \dots, \boldsymbol{x_n}$

- We assume that the points until c_i^{th} nearest are in the same group with $\boldsymbol{x_i}$.
  - ⋆ The local σ_i is estimated as the distance from $\boldsymbol{x_i}$ to c_i^{th} nearest data point $\boldsymbol{x_i}$.
  - ⋆ $CNN(\boldsymbol{x_i}, \boldsymbol{x_j})$ is also determined with the number of common points of two neighbor groups $\boldsymbol{x_i}$ and $\boldsymbol{x_i}$.
- With σ_i and $CNN(\boldsymbol{x_i}, \boldsymbol{x_j})$, the estimated adjacency matrix with the proposed algorithm is defined by

$$\tilde{a}_{ij} = exp\left\{-\frac{d^2(\boldsymbol{x}_i, \boldsymbol{x}_j)}{\sigma_i\sigma_j(CNN(\boldsymbol{x_i}, \boldsymbol{x_j}) + 1)}\right\}, i, j = 1, ..., n$$

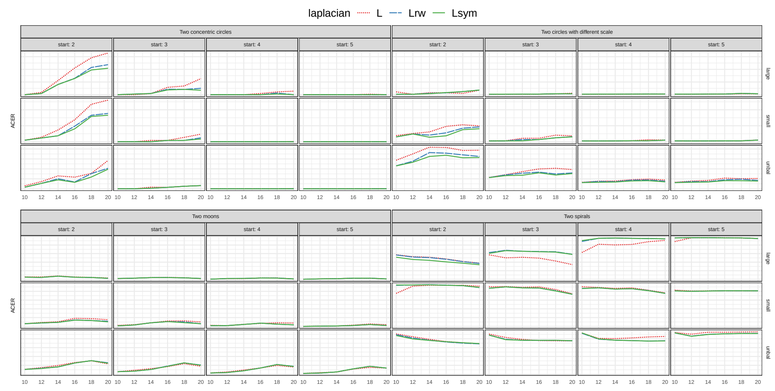

Figure 1. The simulation results to find the optimal parameter

- Figure 1 shows the simulation results in various setting of the starting point from 2 to 5 and the number of data points from 10% to 20% percents of total numbers.
- From the results, we decided to set the searching range of c_i from $k = 3$ to $k = n^*$ where $n^* = n/10$.
  - ⋆ The adjacency of a paricular data point with 1^{st} nearest or 2^{nd} nearest point are very high so the distance value would likely be a outlier.
  - ⋆ About n^*, the purpose of our algorithm is to define neighbor so there is no reason to explore the distance to whole data points.

Numerical studies

Visualization of the proposed method

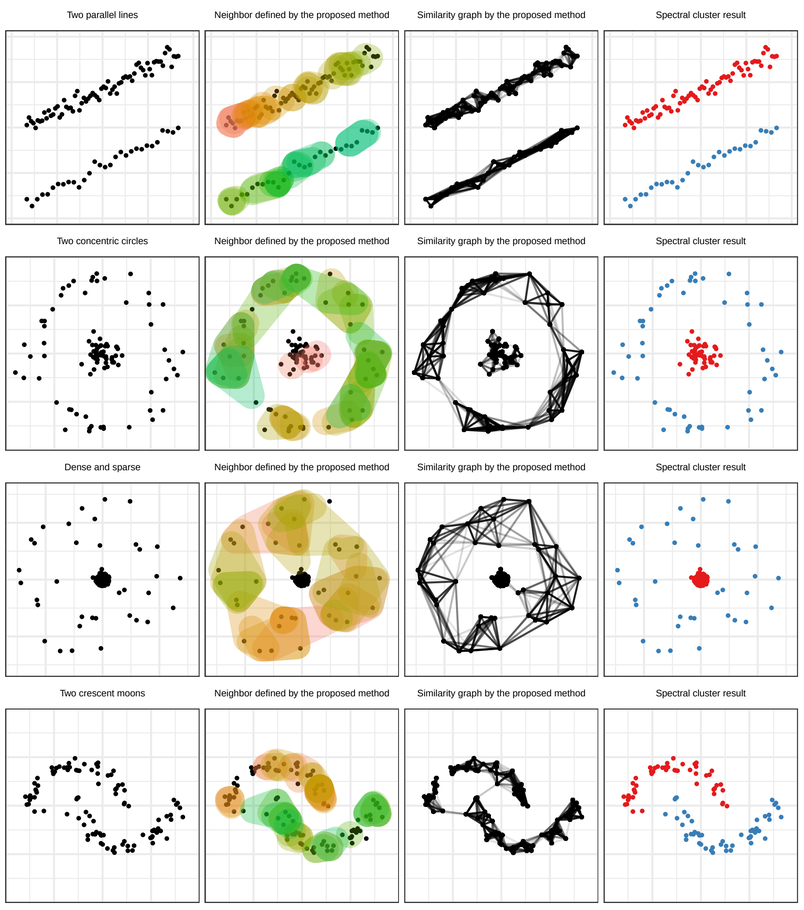

Figure 2. Neighbor from each data point and similarity graphs

- Figure 2 is the visualization results of the proposed algorithm. We could see how the algorithms find the neighbor areas of each data point.
- Data points within cluster share the neighbors. It makes the two points which is far from each other but still in the same cluster be connected by paths through the other points in the cluster. Based on the neighbor part, the similarity graph has large weighted edges within cluster.
- As the results in the last column, we could identify the proposed algorithm successes to cluster the data.

Comparison of clustering performance

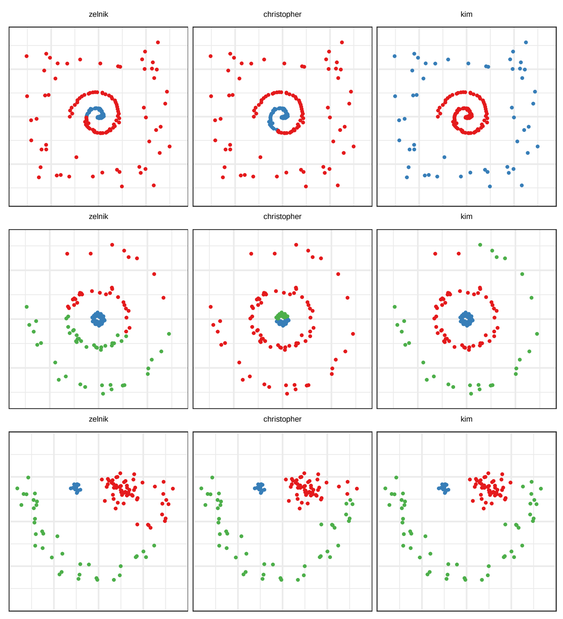

Figure 3. Spectral clustering results for three algorithms

- Figure 3 are the simulation results of the normalized spectral clustering with $\boldsymbol{L_{sym}}$.The first two columns are from self-tuning spectral clsutering(SP-ST) and fast density-aware spectral clustering(SP-FA). The last one is from **the proposed method**(SP-WN*).

Table 1. Average clustering error rate of 100 iterations for artificial datasets

Dataset	Methods	ACER $\boldsymbol{L}$	$\boldsymbol{L_{rw}}$	$\boldsymbol{L_{sym}}$
Spiral and square	SP-ST	0.38	0.38	0.38
	SP-FA	0.31	0.31	0.31
	SP-WN*	**0.11**	**0.15**	**0.15**
Three concentric circles	SP-ST	0.32	0.33	0.33
	SP-FA	0.24	0.28	0.28
	SP-WN*	**0.14**	**0.16**	**0.16**
Smile face	SP-ST	0.06	0.06	0.06
	SP-FA	0.04	0.04	0.04
	SP-WN*	**0.02**	**0.02**	**0.02**

- Clustering results for real datasets.

Table 2. Clustering error rate for UCI datasets

Dataset	Methods	ACER $\boldsymbol{L}$	$\boldsymbol{L_{rw}}$	$\boldsymbol{L_{sym}}$
Diabetes	SP-ST	0.354	0.354	0.354
	SP-FA	0.353	0.353	0.354
	SP-WN*	**0.352**	0.353	0.354
Glass	SP-ST	0.603	0.603	0.598
	SP-FA	0.589	0.603	0.603
	SP-WN*	**0.561**	**0.556**	**0.556**
Ionosphere	SP-ST	0.484	**0.265**	**0.265**
	SP-FA	0.476	0.296	0.296
	SP-WN	**0.285**	0.311	0.311
Iris	SP-ST	0.107	0.100	0.100
	SP-FA	0.100	0.067	0.093
	SP-WN*	**0.060**	**0.040**	**0.040**
Leukemia	SP-ST	0.486	0.236	0.236
	SP-FA	**0.25**	**0.125**	**0.125**
	SP-WN*	0.458	0.153	0.153
Seeds	SP-ST	0.133	0.133	0.133
	SP-FA	0.157	0.157	0.157
	SP-WN*	**0.100**	**0.105**	**0.105**
Sonar	SP-ST	**0.442**	0.462	0.462
	SP-FA	0.481	0.481	0.481
	SP-WN*	0.447	**0.385**	**0.385**
Thyroid	SP-ST	**0.047**	**0.042**	**0.037**
	SP-FA	0.051	0.061	0.066
	SP-WN*	0.056	0.061	0.047
Wine	SP-ST	0.034	0.034	0.034
	SP-FA	**0.028**	0.034	0.034
	SP-WN*	**0.028**	0.034	0.034

- We tested the performace of the three versions of spectral clustering algorithms with different estimation methods for the adjacency matrix on real datasets from the UCI data repository.
- In most cases, the proposed algorithm (SP-WN*) would be considered the better choice than self-tuning spectral clsutering(SP-ST) or fast density-aware spectral clustering (SP-FA).

Conclusion

- In this study, we proposed the automatic selection algorithm for the scale parameter of Gaussian kernel in spectral clustering.
- The motivation for the proposed algorithm is searching neighbors of each data which represents the dispersion of the data points and shows the relationship between others.
- With the well-defined neighbors, we could estimate the locally weighted scale parameter and construct the good adjacency matrix.
- The results of simulation with artificial and real datasets demonstrate the proposed algorithm outperfoms the previous suggested methods in three types of spectral clustering using different graph Laplacians.