Large-Scale Spectral Clustering on Graphs

Jialu Liu, Chi Wang, Marina Danilevsky, Jiawei Han
University of Illinois at Urbana-Champaign, Urbana, IL, United States
{jliu64, chiwang1, danilev1, hanj}@illinois.edu

Introduction

- We propose an efficient and effective spectral clustering algorithm on *graphs*.
  1. Many existing works which were claimed to be efficient in the published papers do not suit for graph data.
  2. Some "efficient" algorithms may not be efficient for all kinds of graph data.
  3. It is challenging to mitigate the computational bottleneck while still providing a high-quality clustering.
- Compared to standard spectral clustering:
  1. Efficiency: one order of acceleration on average.
  2. Effectiveness: Comparable with the traditional Spectral Clustering.
  3. Easy to implement.

Brief review of Spectral Clustering

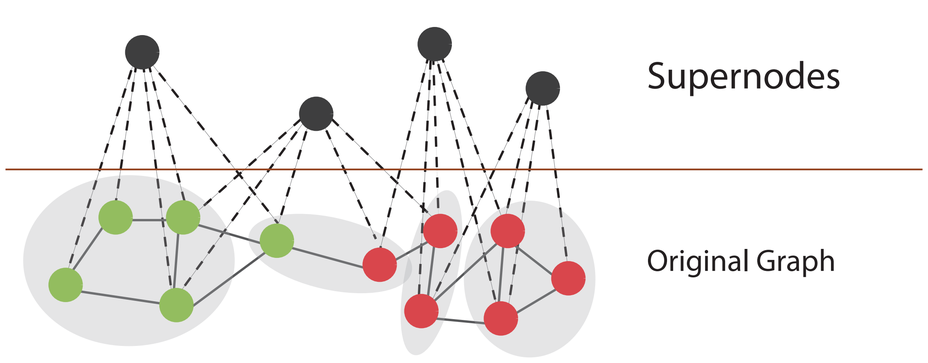
Spectral clustering aims at finding k orthonormal vectors $X_1, X_2, \ldots, X_k$ with the following objective function:

$$\max_X \ Tr(X^T D^{-1/2} W D^{-1/2} X) \quad \text{s.t. } X^T X = I$$

where $X \in \mathbb{R}^{n \times k}$ is a matrix of k column vectors and k is number of clusters. W is the adjacency matrix denoting an undirected and weighted graph.
It is worth noting that the complexity of EVD on an $n \times n$ graph is $O(n^3)$ without considering the sparsity or calculating limited pairs of eigenvalues/eigenvectors.

Intuition

Supernodes are behaving like aggregations of local clusters in the graph. We expect the partition on both sides (regular nodes and supernodes) can mutually enhance each other.



Method Overview (ESCG)

1. Obtain initial partition on the graph (very fast but poor performance).
2. Generate "supernodes" based on this coarse summarization.
3. Transform original graph to bipartite structure by linking supernodes and regular nodes.
4. Compute embeddings on this bipartite representation.

Time complexity: $O(md + nd \log n + nd^2)$

Step 1: Initial Clustering

- Randomly pick d seeds in the graph.
- Compute shortest paths from these seeds to the rest.

$$M_{ij} = -\log \frac{W_{ij}}{\max W} + \varepsilon$$

- Partition all the nodes into d disjoint subsets represented by the seeds.

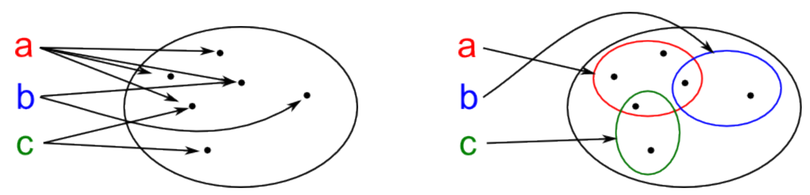
Step 2: Generation of Supernodes

A binary matrix $R \in \mathbb{R}_{d \times n}$ is used to describe the current linkage between supernodes and regular nodes.

Step 3: From Homogeneous to Bipartite Graph

The one-to-many linkage between supernodes and original nodes can not help propagating information. We should extend this to many-to-many relationship and incorporate the overlap via

$$\hat{W} = RW$$



Step 4: Clustering on Bipartite Graph $\hat{W}$

Similar to BSGP, we solve a generalized eigenvalue decomposition (GEVD) of L' and D':

$$\begin{bmatrix} D_1 & -\hat{W}^T \\ -\hat{W} & D_2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \lambda \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}.$$

Compute $D_2^{-1/2} \hat{W} D_1^{-1/2}$ and apply SVD on it to obtain right singular vectors $\hat{X}$.

$$U = D_1^{-1/2} \hat{X}$$

Iterative Method (ESCG-R)

We can improve the previously obtained initial partitions by making use of embeddings obtained from Step 4 above.
After finishing ESCG, ESCG-R repeats Steps 2-4 for some iterations.

Data sets

Data sets	Nodes	Edges	Clusters	Sparsity
Syn-1K	1,000	10,000	3	0.01
Syn-100K	100,000	8.2M	10	0.0016
DBLP	28,702	62.4M	4	0.1515
IMDB	30,731	257K	4	5×10^{-4}

Compared Methods

- Shortest Paths (**SP**) algorithm (Step 1).
- Efficient Spectral Clustering on Graphs (**ESCG**), the method proposed.
- Efficient Spectral Clustering on Graphs with Regeneration (**ESCG-R**), the second method in this paper with regeneration of supernodes.
- Standard Spectral clustering (**SC**) algorithm.
- Resistance Embedding Spectral Clustering (**RESC**).
- **Nyström**, an method to find numerical approximation to eigen-decomposition.

Table 1: Clustering accuracy on the four data sets (%)

Data sets	Syn-1K	Syn-100K	DBLP	IMDB
SP	73.6	37.6	58.0	46.3
ESCG	**100**	97.0	70.6	49.5
ESCG-R	**100**	**100**	**82.1**	51.8
SC	**100**	**100**	78.2	55.2
RESC	**100**	10.6	27.9	**58.2**
Nyström	40.0	17.1	78.4	40.1

Table 2: Running time on the four data sets (s)

Data sets	Syn-1K	Syn-100K	DBLP	IMDB
SP	0.001	0.33	1.72	0.02
ESCG	**0.023**	**1.36**	**3.81**	**0.17**
ESCG-R	0.041	7.38	9.23	0.49
SC	0.228	5.06	27.7	74.4
RESC	0.166	201	2394	24.9
Nyström	0.033	19.1	3.89	4.41

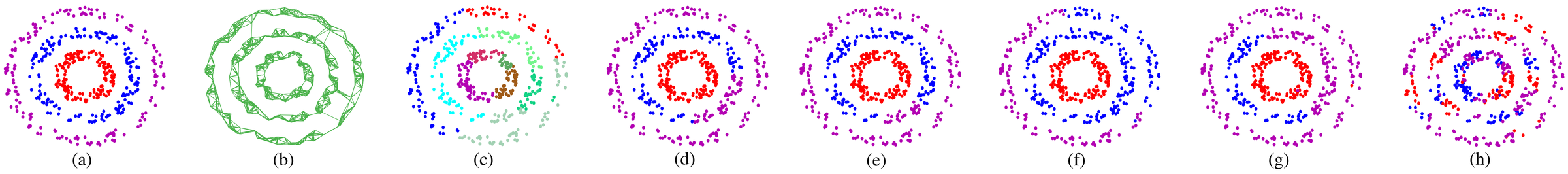


Figure 1: (a) Ground-truth partition; (b) k-NN graph; (c) Initial clustering (Step 1); (d) ESCG; (e) ESCG-R after 5 iterations; (f) Spectral clustering; (g) RESC; (h) Nyström.