Fast Approximate Spectral Clustering

Donghui Yan Ling Huang Michael I. Jordan

Department of Statistics, U.C. Berkeley

Intel Research Berkeley

Department of Statistics and EECS, U.C. Berkeley

Challenges of clustering on modern datasets

Modern datasets scale along several dimensions

- ◇ Large number of features (dimensionality)
  - e.g., web access log (~ 20), image (> 100), microarray and genomics data (~ 4000)
- ◇ Huge number of observations (scalability)
  - e.g., US Census Income (285,779), Poker hand (1,000,000)
- ◇ Increasingly complex in structure
  - e.g., nonlinearity of interesting patterns, “heterogeneity” (“locality”) of data in the space.

This work focuses on the scalability issue for spectral clustering

- ▶ To leverage the remarkable ability of spectral clustering in handling complex patterns with scalability in mind.

Spectral clustering

Spectral clustering aims to partition a set of given points $V = \{X_1, ..., X_N\}$ into K disjoint classes by spectral decomposition over an affinity graph $\mathcal{G} = (V, \mathcal{E}, A)$ with the edge weights $(A_{ij})_{i,j=1}^{N}$ encoding the pairwise similarity of points in V.

Popular spectral clustering algorithms include

- Normalized cuts (Shi & Malik, 2000)
- Ng, Jordan and Weiss (2002)
- Kannan, Vempala and Vetta (2004).

Normalized cuts is adopted in this work.

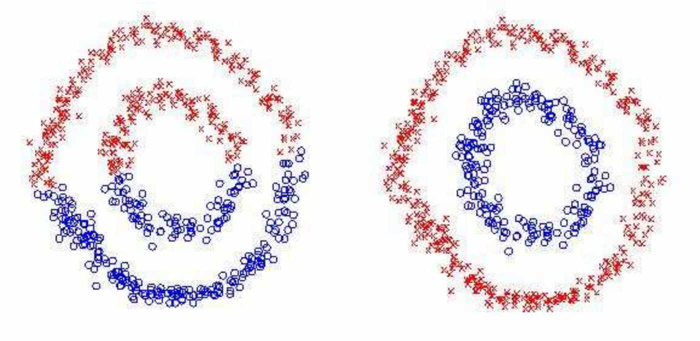
Why spectral clustering?

- ◇ Extensive studies in computer vision, machine learning, parallel computing during the last two decades.
- ◇ Wide range of applications in image segmentation, circuit design, search (clusty), spam detection, social network mining.
- ◇ Theoretical support (von Luxburg et al 2008; Kannan et al 2004; Ng et al 2002).
- ◇ Compared to competitors (e.g., K-means, hierarch. clustering)
  - ▸ More flexible and capture a wider range of geometries (e.g., nonlinearity and nonconvexity)
  - ▸ Typically superior empirical performance.

BUT not widely viewed as a player for large-scale data mining due to a complexity of up to $O(N^3)$.

Why spectral clustering?

- An example of K-means (left) and spectral clustering (right).



Methods to speed up spectral clustering

- Lanzcos/Arnoldi methods
  - Computation depends highly on problem difficulty
- Rank reduction methods (the Nyström methods)
  - To sparsify Gram matrix with a low-rank approximation
  - Sample columns of Gram matrix and approximate the full matrix

$$G = \begin{bmatrix} C & B \\ B^T & D \end{bmatrix} \approx \begin{bmatrix} C & B \\ B^T & B^T C^{-1} B \end{bmatrix}$$

  - Williams and Seeger (2001), Drineas and Mahoney (2005)
  - General issues
    - The working memory can be very high ($\sim O(N^2)$)
    - For unbalanced data sets, small clusters may be missed and potential problems with numerical stability.

A framework for fast approximate spectral clustering

A **class** of algorithms which consists of three steps

- Replace the original data with a small "representative" set via a "distortion" minimizing local transformation.
- Spectral clustering on the representative set.
- Recover cluster membership for the original data according to their correspondence to the representative set.

The key is to look for a distortion-minimizing transformation (min. quant. error is sufficient by our perturb. analysis)

- K-means clustering
- Random projection trees (Dasgupta and Freund, 2008).

A framework for fast approximate spectral clustering

$$\begin{array}{rl} \text{Original set } \mathcal{S}_0 = \{X_1, ..., X_N\} & \dashleftarrow O(N^3) \\ \Downarrow & \text{Distortion minimizing} \\ \text{Bridge set } \mathcal{S} = & \{Y_1, ..., Y_1, ..., Y_n, ..., Y_n\} \\ \Updownarrow & \text{Equiv. (Embed. Lemma)} \\ \text{Representative set } \mathcal{S}_1 = \{Y_1, ..., Y_n\} & \dashleftarrow O(n^3) \end{array}$$

- Distortion min. $\Longleftrightarrow$ small loss in accuracy (perturb. analysis)
- $|\mathcal{S}_1| = n \ll |\mathcal{S}_0| = N$ $\Longleftrightarrow$ significant reduction in computation
- Overall computational complexity $O(n^3) + O(ndN)$.

A framework for fast approximate spectral clustering

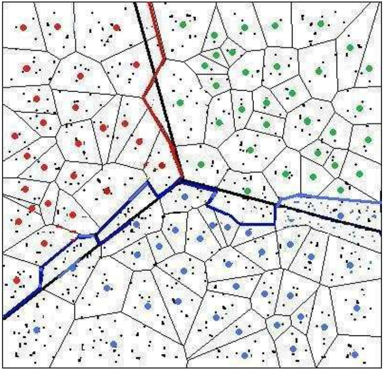


Figure: *Small loss in clustering accuracy via distortion minimizing local transformation. Straight and zigzag solid lines indicate cluster boundaries on original and transformed data, respectively.*

Datasets

- Datasets used in the experiments (smaller ones omitted)

Data set	# Features	# instances	# classes
Connect-4	42	67,557	3
USCI	37	285,779	2
Poker Hand	10	1,000,000	3

- Competing algorithms
  - Various K-means algorithms
    - Hartigan and Wong (1979)
    - K-means in Matlab with the “cluster” option
    - Bradley and Fayyad (1998).
  - The Fowlkes et al implementation of Nyström (2004).

Experimental results

	RF	K-means	Nyström	KASP	RASP
Connect-4	75.00	65.33	65.82	65.69	63.95
		3	181	51	67
		0.19	4.0	0.20	< 0.4
USCI	95.27	63.47	93.88	94.03	92.09
		11	1603	282	418
		0.65	12.0	0.78	< 0.8
Poker Hand	60.63	35.56	50.24	49.84	49.70
		35	1047	310	215
		0.42	17.0	0.45	< 0.5

Table: *Comparison on accuracy, running time and memory footprint. Numbers for Nyström produced by Matlab while the rest in R. Further increasing running time for K-means does not improve its accuracy.*

Statistical perturbation analysis

- Assume the cluster is generated by mixture

$$G = \sum_{i=1}^{K} \pi_i G_i. \tag{1}$$

- Limit to additive perturbation $\tilde{X} = X + \epsilon$ and assume $\epsilon \in \mathbb{R}^d$ is symmetric about 0.
- What is the impact of perturbation on spectral clustering?
  - Measured by *mis-clustering rate* defined as

$$\rho = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}\{I_i \neq \tilde{I}_i\},$$

I_i and $\tilde{I}_i$ indicates cluster ID before and after perturbation.

Mis-clustering rate of KASP

Theorem. Let the data be generated from (1) with density $f : \mathbb{R}^d \mapsto \mathbb{R}^+$. Then, under suitable assumptions, the mis-clustering rate is bounded by

$$Cb_{2,d}||f||_{d/(d+2)}n^{-2/d} + O\left(n^{-4/d}\right)$$

where C is a constant depending on the number of clusters, the variance of the original data, the similarity metric and the eigengap of $\mathcal{L}$ (or that of all Laplacian matrices used in Ncut).

$\Longrightarrow$ The mis-clustering rate ρ vanishes when $n \to \infty$.

The embedding lemma

Let $\mathcal{S} = \{Y_1, Y_1, \ldots, Y_1, Y_2, \ldots, Y_2, \ldots, Y_n, \ldots, Y_n\}$ be the bridge set with repetition counts r_i s.t. $\sum_{i=1}^{n} r_i = N$.

Lemma. 1). The 2^{nd} eigenvector, $\boldsymbol{v}_2$, for $\mathcal{L}_{\mathcal{S}}$ can be written as

$$\boldsymbol{v}_2 = [x_1, \ldots, x_1, x_2, \ldots, x_2, \ldots, x_n, \ldots, x_n]^T,$$

where the number of repetitions for x_i is exactly r_i.

2). Let matrix $B = [r_1\boldsymbol{a}_1, r_2\boldsymbol{a}_2, ..., r_n\boldsymbol{a}_n]$ with $[\boldsymbol{a}_1, \boldsymbol{a}_2, ..., \boldsymbol{a}_n]$ the affinity matrix for $\mathcal{S}_1$. Let $\boldsymbol{v}_B = [y_1, y_2, \ldots, y_n]^T$ be the second eigenvector of $\mathcal{L}_B$. Then, up to scaling,

$$x_1 = y_1, x_2 = y_2, \ldots, x_n = y_n.$$

$\Longrightarrow$ $\boldsymbol{v}_2$ can be computed through $\boldsymbol{v}_B$.

Summary

- ◇ A general framework for fast approximate spectral clustering
- ◇ Distortion-minimizing local transformations implemented by K-means and RP tree partitions
- ◇ Statistical perturbation analysis of spectral clustering serves as the theoretical motivation of the general framework
- ◇ Empirically our algorithms are competitive in terms of accuracy, running time, and working memory.

http://www.cs.berkeley.edu/∼jordan/fasp.html

The end

Thank you!