Lecture 19: Spectral Clustering

Akshay Krishnamurthy
akshay@cs.umass.edu

November 14, 2017

Today we will talk about unsupervised learning, and in particular an algorithm called spectral clustering. This algorithm makes deep connections between machine learning, graph theory, and linear algebra.

1 The setting and algorithm

We are given n data points $x_1, \ldots, x_n$ and some way to compute similarities between them, call $s_{i,j}$ the similarity between the n points. Assume that the similarity function is symmetric, so $s_{i,j} = s_{j,i}$. For the purposes of this lecture let us just consider the simplified clustering problem where we would like to split the dataset into two clusters. We would like to find a subset $S \subset [n]$ of size roughly $n/2$ such that the similarity between points in S (and S^C) is high, while the similarity between the clusters is low.

We do not assume that we have any object features, just the similarities $s_{i,j}$. Using these, we construct a similarity graph $G = (V, E)$ where the vertices are the n objects, so $|V| = n$ and the edges are weighted with $s_{i,j}$. In graph-theoretic terminology, we want to partition the graph so that the edge between the partitions have low weight while the edges within each side of the partition have high weight.

Let us form the adjacency matrix of this graph, which we will always call $A \in \mathbb{R}^{n \times n}$. This matrix has $A_{i,j} = s_{i,j}$ and is unspecified on the diagonal (don't worry we won't really use it at all). Define the degree matrix $D \in \mathbb{R}^{n \times n}$ which is a diagonal matrix with $D_{i,i} = \sum_{j=1}^{n} s_{i,j}$, the (weighted) degree of vertex i in the graph.

1.1 Eigenvectors

Spectral clustering deals largely with the eigenvectors and eigenvalues of matrices defined in terms of the graph. Let us first review some basic facts about these objects. Throughout let $M \in \mathbb{R}^{n \times n}$ be a symmetric matrix.

Definition 1. *A vector $v \in \mathbb{R}^n$ and a scalar λ are an eigenvector/value pair if $Mv = \lambda v$.*

Theorem 2. *Let $M \in \mathbb{R}^{n \times n}$ be a real symmetric matrix. Then there exists a set of orthonormal eigenvectors $v_1, \ldots, v_n$ and corresponding eigenvalues $\lambda_1, \ldots, \lambda_n$ such that $Mv_i = \lambda_i v_i$ for all i. Hence we can write $M = \sum_i \lambda_i v_i v_i^T = V \Lambda V^T$ where V is a matrix with orthonormal columns and Λ is a diagonal matrix.*

Multiplicity. Note that there can be multiple eigenvectors with the same eigenvalue. For instance, we could have $Mv_1 = \lambda v_1$ and $Mv_2 = \lambda v_2$ where $v_1 \perp v_2$. In this case, any linear combination of v_1, v_2 is also an eigenvector with eigenvalue λ. In fact the entire subspace $\text{span}(v_1, v_2)$ is an *eigenspace*. In this instance we say that λ is an eigenvalue with multiplicity two. Where the multiplicity is the dimension of the eigenspace.

Rayleigh Quotient. The above definition of an eigenvector is a fixed-point characterization, specifically v is a fixed point of the map $M : \mathbb{R}^n \to \mathbb{R}^n$ when restricted to the unit sphere. Another characterization is an optimization based characterization, via the Rayleigh Quotient

$$\lambda_{\max} = \max_{v \neq 0} \frac{v^T M v}{v^T v}$$

As such we can find the eigenvectors by solving this (nonconvex!) optimization problem and then deflating by writing $M' = M - \lambda v v^T$ and repeating.

Theorem 3. *The eigenvalues of M are the roots of the* characteristic polynomial $f(t) = \det(tI - M)$.

Definition 4. *A symmetric matrix M is* positive semi-definite *if all eigenvalues are non-negative. It is positive definite if all eigenvalues are strictly positive.*

As a corollary, if a matrix is positive-semidefinite then $x^T Mx \geq 0$ for all x.

1.2 Graph Laplacian

The main graph theoretic object in spectral clustering is the *Graph Laplacian*. While there are multiple different types, we will mostly focus on the *unnormalized graph laplacian*, defined as $L = D - W$.

Lemma 5 (Properties of Laplacian)**.** *1. For every $x \in \mathbb{R}^n$ we have $x^T Lx = \frac{1}{2}\sum_{i,j} s_{i,j}(x_i - x_j)^2$*

2. L is symmetric, p.s.d.

3. $\lambda_{\min}(L) = 0$ and the corresponding eigenvector is $\mathbf{1}/\sqrt{n}$.

Proof. For the first property

$$x^T Lx = x^T Dx - x^T Ax = \sum_i d_i x_i^2 - \sum_{i,j} x_i x_j s_{i,j} = \frac{1}{2}\left(\sum_i x_i^2 d_i - 2\sum_{i,j} x_i x_j s_{i,j} + \sum_j x_j^2 d_j\right) = \frac{1}{2}\sum_{i,j} s_{i,j}(x_i - x_j)^2$$

The p.s.d. property follows since the matrix quadratic $x^T Lx \geq 0$ for all x. Symmetry is obvious. The minimum eigenvalue follows from the expansion of the matrix quadratic. □

The key intuition behind spectral clustering is the following fact, relating the spectrum of the laplacian to the number of connected components.

Proposition 6. *The multiplicity k of the eigenvalue 0 of L equals the number of connected components in the graph G. If $A_1, \ldots, A_k$ are the connected components, then the eigenspace corresponding to eigenvalue 0 is spanned by the vectors $\{\mathbf{1}_{A_i}\}_{i=1}^k$.*

The proof is somewhat straightforward, using the expansion of the quadratic form. The other important fact is that when G has k connected components, then the Laplacian is block diagonal, but moreover each block corresponds to a proper Laplacian

$$L_G = \begin{pmatrix} L_{A_1} & 0 & 0 \\ \vdots & \ddots & \cdots \\ 0 & 0 & L_{A_k} \end{pmatrix}$$

where L_{A_i} is the graph laplacian for the subgraph corresponding to the vertices in A_i.

1.3 The algorithm

The above proposition motivates the following algorithm. Let us focus on the case where $k = 2$.

1. Compute v_2, the eigenvector corresponding to the second smallest eigenvalue of L.
2. Let $C_1 = \{i : v_2(i) > 0\}$ and $C_2 = \{i : v_2(i) \leq 0\}$. Return $\{C_1, C_2\}$ as the clustering.

Consider the case where $k = 2$ and the graph has two connected components A_1, A_2. While v_2 is not uniquely defined, once we commit to $v_1 = \mathbf{1}$, we are guaranteed that

$$v_2 = \left(\underbrace{\frac{|A_2|}{\sqrt{n|A_1||A_2|}}, \ldots, \frac{|A_2|}{\sqrt{n|A_1||A_2|}}}_{A_1}, \underbrace{\frac{-|A_1|}{\sqrt{n|A_1||A_2|}}, \ldots, \frac{-|A_1|}{\sqrt{n|A_1||A_2|}}}_{A_2} \right)$$

$$= \left(\underbrace{\sqrt{\frac{|A_2|}{n|A_1|}}, \ldots, \sqrt{\frac{|A_2|}{n|A_1|}}}_{A_1}, \underbrace{-\sqrt{\frac{|A_1|}{n|A_2|}}, \ldots, -\sqrt{\frac{|A_1|}{n|A_2|}}}_{A_2} \right)$$

Thus if we cut this vector at 0, we will identify the right clusters.

When we are looking for k clusters, we modify the algorithm slightly

1. Compute $v_1, \ldots, v_k$, the k eigenvectors corresponding to the smallest k eigenvalues of L.
2. Embed the ith point as $u_i = (v_1(i), \ldots, v_k(i)) \in \mathbb{R}^k$.
3. Run any clustering algorithm (like k-means) on the embeddings $u_1, \ldots, u_n$.

Typical analyses of k-way spectral clustering either design specialized clustering algorithms for the last step, or do not say much about this part. The binary case is in some sense the easiest since you embed into one dimension (you don't need to use v_1), so clustering algorithms can be trivial there.

2 Optimization perspective

So far we have seen why spectral clustering makes some sense only for a graph with disconnected components. Of course in this case clustering is trivial, so it is important to understand what happens when the graph is connected. There are two main ways to think about this, one is predominantly from the theoretical computer science community and is based on viewing the spectral clustering algorithm as an approximation to various NP-hard optimization problems. Let us first dive into this perspective.

Focusing on binary clustering problems define

$$\text{Cut}(S) = \sum_{i \in S, j \notin S} A_{i,j}.$$

We might imagine that finding the set S that minimizes $\text{Cut}(S)$ reveals a good clustering. This is the well-studied min-cut problem, that hopefully many of you have seen. It can be solved in polynomial time via the min-cut max-flow theorem. Unfortunately Min-cut fails to capture what we want in clustering, since it often finds a very small cluster, e.g., one where we separate just a single vertex from the rest of the graph. This is clearly not what we want for clustering, and hence we need a new objective function

$$\text{RatioCut}(S) = \text{Cut}(S) \times \frac{1}{2} \left(\frac{1}{|S|} + \frac{1}{|S^C|} \right)$$

This RatioCut objective now penalizes choosing a small cluster, and hence leads to more balanced partitions.

Let us rewrite this optimization problem in a vectorized notation. For a set S, define

$$x_S = \begin{cases} \sqrt{|S^C|/|S|} \text{ if } i \in S \\ -\sqrt{|S|/|S^C|} \text{ if } i \notin S \end{cases}$$

Now we show that the RatioCut objective is nothing but the Laplacian quadratic form for this vector

Lemma 7.

$$x_S^T L x_S = n \times \mathit{RatioCut}(S)$$

Proof.

$$\begin{aligned} x_S^T L x_S &= \frac{1}{2}\sum_{i,j} s_{i,j}(x_S(i) - x_S(j))^2 \\ &= \frac{1}{2}\sum_{i\in S, j\notin S} s_{i,j}\left(\sqrt{\frac{|S^C|}{|S|}} + \sqrt{\frac{|S|}{|S_C|}}\right)^2 + \frac{1}{2}\sum_{i\notin S, j\in S} s_{i,j}\left(-\sqrt{\frac{|S^C|}{|S|}} - \sqrt{\frac{|S|}{|S_C|}}\right)^2 \\ &= \mathrm{Cut}(S)\left(\frac{|S^C|}{|S|} + \frac{|S|}{|S^C|} + 2\right) = n \times \mathrm{RatioCut}(S) \end{aligned}$$

□

Additionally it is not hard to see that $\mathbf{1}^T x_S = \sum_i x_S(i) = 0$ with this definition, hence x_S for any S, is orthogonal to the all-ones vector. Moreover it is easy to check that $\|x_S\| = \sqrt{n}$. Thus we can re-write the ratio cut optimization

$$\min_{S\subset V} n \times \mathrm{RatioCut}(S) = \min_{S\subset V} x_S^T L x_S \text{ s.t. } x_S \perp \mathbf{1}, \|x_S\| = \sqrt{n}$$

This is still a discrete optimization problem, which is NP-hard in general. The most obvious relaxation is to remove the integrality requirement on the structure of the vector x. This gives

$$\min_x x^T L x \text{ s.t. } x \perp \mathbf{1}, \|x\| = \sqrt{n}$$

Since we know that $\mathbf{1}$ is the smallest eigenvector, this is Rayleigh-Quotient definition of the second smallest eigenvector!

What can we say about the approximation properties of this relaxation? I am not aware of anything here but there is a slightly different algorithm that does produce a (weak) approximation guarantee to a related problem. Instead of the unnormalized Laplacian $L = D - A$, we use the normalized Laplacian $\bar{L} = D^{-1/2}(D-A)D^{-1/2} = I - D^{-1/2}AD^{-1/2}$. This normalized Laplacian has similar properties to L (since it is equivalent up to a row/column scaling). Let us further assume that G is d-regular, so that $D = dI$ or all vertices have degree d. In this case, define the *edge expansion*

$$\phi(G) = \min_{S\subset V} \frac{\mathrm{Cut}(S, S^C)}{d\min\{|S|, |S^C|\}}.$$

This is quite close to our definition of RatioCut and morally is doing the same thing.

Theorem 8 (Cheeger Inequalities)**.** *Let G be a undirected regular graph and let λ_2 be the second smallest eigenvalue of the normalized Laplacian $\bar{L}$. Then*

$$\frac{\lambda_2}{2} \le \phi(G) \le \sqrt{2\lambda_2}$$

Moreover there is a simple algorithm based on sorting the coordinates of the second eigenvector v_2 and taking the best partitioning of the form $S = \{v_2(1), \ldots, v_2(k)\}$ that has $\phi(S) \le \sqrt{2\lambda_2}$

This gives an approximation of $2\sqrt{\phi(G)}$ for edge expansion, which at least by visual inspection is closely related to RatioCut.

3 Statistics perspective

The second perspective to understand spectral clustering, largely from the statistics community, is based on matrix perturbation theory. The idea is to view the graph G as a random object generated according to a particular probabilistic model. Two common models that are quite similar

1. Gaussian perturbation – G has true clusters S, S^C say both of size $n/2$ and we define

$$A_{ij} = \mu\left(\mathbf{1}\{i,j \in S \vee i,j \in S^C\} - \mathbf{1}\{i \in S, j \notin S \vee i \notin S, j \in S\}\right) + r_{ij}$$

   where $r \sim \mathcal{N}(0,\sigma^2)$. Or defining the cluster indicator $\mathbf{1}_S(i) = \mathbf{1}\{i \in S\} - \mathbf{1}\{i \notin S\}$, we have $A = \mu \mathbf{1}_S \mathbf{1}_S^T + R$ where R is a gaussian random matrix.

2. Stochastic Block Model – This is similar but with bernoulli perturbation. If (i,j) belong to the same cluster then we see an edge with probability p, otherwise we see an edge with probability q.

In both cases we can write the adjacency matrix $A = M + R$ where R is some mean-zero perturbation. For this discussion the model doesn't really matter too much. Since the normalized Laplacian is also additive, we can write $\hat{L} = L_M + L_R$ where L_R is the Laplacian of the random matrix. The main two tools used in the analysis are Weyl's theorem and the Davis-Kahan theorem

Theorem 9 (Weyl)**.** *Suppose that A, M, R are matrices with $A = M + R$. Let A have eigenvalues $\lambda_1 \geq, \ldots, \lambda_n$ and let M have eigenvalues $\mu_1, \ldots, \mu_n$. Then*

$$|\lambda_i - \mu_i| \leq \lambda_{\max}(R)$$

Theorem 10 (Davis-Kahan)**.** *Let $A = M + R$ have eigenvalues and eigenvectors $\lambda_1 \geq \ldots \geq \lambda_n$, $v_1, \ldots, v_n$. Let M have eigenvalues $\mu_1 \geq \ldots \geq \mu_n$ and eigenvectors $u_1, \ldots, u_n$. Define $\delta_i = \min_{j \neq i} |\lambda_i - \lambda_j|$. Then*

$$\sin \angle(u_i, v_i) \leq \frac{\lambda_{\max}(R)}{\delta_i}$$

A rough analysis for spectral clustering goes as follows. Assume one of the models above and characterize the eigenvalues and eigenvectors of the mean Laplacian matrix. This is easy in those models since the mean adjacency matrix is rank one. Then characterize the spectral norm of the noise laplacian L_R. We haven't seen this in class but you can use some of the ideas we saw in the concentration of measure section. Specifically, the Rayleigh characterization of spectral norm is

$$\|R\|_{op} = \sup_{x:\|x\|=1} x^T R x$$

and hence we can use ideas from uniform convergence for this step. Next, since for unit vectors the sine of the angle is related to the Euclidean norm, the Davis-Kahan theorem gives us a perturbation bound on the eigenvector of the observed (noisy) Laplacian. Using this, we can understand what signal to noise ratio is sufficient for the algorithm to succeed with high probability. These steps can all be complicated and to get

I will just give you one of the results I know in this area

Theorem 11 (Balakrishnan et al., McSherry)**.** *With balanced clusters of size $n/2$, in the gaussian block model there is a spectral clustering algorithm that succeeds with high probability when $\frac{\mu}{\sigma} = \omega(\sqrt{\log(n)/n})$.*

The algorithm is slightly different from spectral clustering, but is based on using the eigenvectors. Spectral clustering can also be shown to achieve a slightly worse bound, but Balakrishnan et al., analyze the algorithm under more general assumptions.

For the stochastic block model a similar consistency guarantee can be proved.

Theorem 12 (McSherry)**.** *In the stochastic block model, there is a constant c such that for sufficiently large n, if*

$$\frac{p-q}{p} > c\sqrt{\frac{\log(n/\delta)}{pn}}$$

then we can recover the planted partition with probability at least $1 - \delta$.

The SBM has been extensively studied in the last 5 years. This result is now quite old and much more is known.