CISC 271 Class 35

Spectral Clustering Of Data

Text Correspondence: Von Luxburg, 2007 [15]

Main Concepts:

- *Laplacian matrix of a graph*
- *Spectral clustering for graphs*
- *Gram matrix as a weight matrix*

Sample Problem, Machine Inference: How can we find more than two clusters in a graph?

We have used principal components analysis, or PCA, extensively for linear data analysis. We have also extended the method to kernel PCA, which used a Gram matrix that was computed from a data matrix and a kernel function.

All of the matrices we have analyzed so far have been symmetric and positive semidefinite. We could reasonably inquire whether PCA works for *any* symmetric and positive semidefinite matrix.

The terse answer is no, but a more nuanced answer is "it depends". We began our explorations with graphs and we can now revisit this topic with our deeper understanding of methods.

35.1 Multiple Clusters In Graphs

The Fiedler vector of the Laplacian matrix of a graph can be used to cut, or cluster, the graph into two partitions. Suppose that a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ has n vertices. The adjacency matrix will be $A \in \mathbb{R}^{n \times n}$. The Laplacian matrix K for an adjacency matrix A is the difference between the degree matrix and the adjacency matrix, which we can write as

$$K = \text{diag}(A\,\vec{1}) - A \tag{35.1}$$

How can we find more than two clusters in a graph? A widely used method is called *spectral clustering*, which is clustering that is based on a spectral decomposition. The Laplacian matrix of a graph, which is K in Equation 35.1, is symmetric and positive semidefinite. This implies that K is diagonalizable and that the orthogonal matrix on the decomposition holds the eigenvectors of the Laplacian matrix.

To perform spectral clustering on a graph, we will simplify our problem by supposing that there is a zero eigenvalue that has the vector $\vec{1}$ as its eigenvector. For n observations, we will write this eigenvalue as $\lambda_n = 0$. Our spectral clustering method, to find k clusters in a graph, is:

- Find the spectral decomposition $K = Q\Lambda Q^T$ so that the eigenvalues λ_j in Λ are sorted as
$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{n-1} \geq \lambda_n = 0$$
- Gather the last k eigenvectors $\vec{q}_j$ into a matrix $E \in \mathbb{R}^{m \times k}$
- Treat E as a set of m observations and perform *k*-means clustering on the matrix E
- Assign the indexes from the *k*-means clustering of the eigenvectors to the original observations in the data

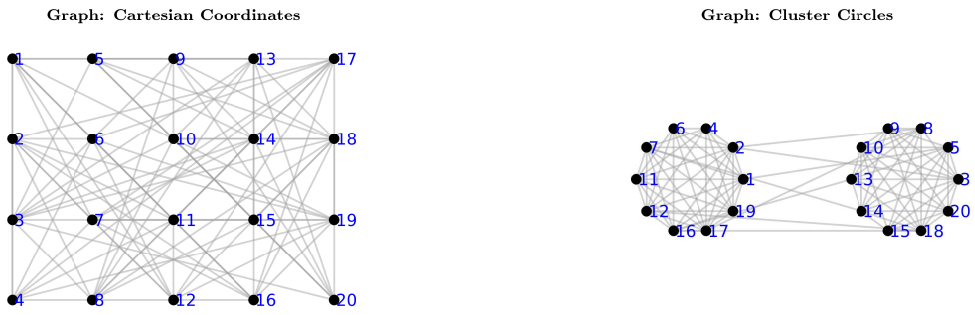
We should note that this is the reverse of PCA: instead of using the k eigenvectors that have the largest eigenvalues, we are using the k eigenvectors that have the *smallest* eigenvalues.

The difference from PCA arises from the adjacency matrix. Because the adjacency matrix has zero for each diagonal entry, the sum of its eigenvalues is zero and, generally, there are negative eigenvalues that confound PCA. As we show in the extra notes for this class, the Laplacian matrix that is created from the adjacency matrix is symmetric and positive semidefinite.

Our algorithm now shows why the Fielder vector is effective at finding two clusters in a graph. It is the eigenvector associated with λ_{n-1} and, when combined with the $\vec{q}_n = \frac{1}{\sqrt{n}}\vec{1}$ of $\lambda_n = 0$, works well for 2-means clustering. The clusters are distinguished by their signs, which must happen because $\vec{q}_{n-1}$ is orthogonal to $\vec{e}_n$ and the positive entries must be "balanced" by the negative entries to have a sum of zero.

35.1.1 Example: 30 Vertices In 3 Clusters

Let us explore a sample graph that was randomly generated. The graph had 30 vertices, 140 edges, and there were three clusters in the graph. We can apply our simple spectral clustering algorithm to this graph and plot the results. As we see in Figure 35.1(A), a Cartesian plot of the vertices has no distinctive visual pattern. When we cluster the vertices and plot the clusters, we can see in Figure 35.1(B) that the clusters are visually apparent.



(A) **(B)**

Figure 35.1: Randomly generated graph that has 30 vertices, 140 edges, and the vertices constitute 3 clusters. (A) The graph shown in Cartesian coordinates. (B) The graph shown with vertices of each cluster on a distinct circle, which reveals the clustered structure of the graph.

35.2 Graph Representation Of Simple Data

Next, we will explore how a graph representation of data can be used to find an alternative clustering of the data. For example, suppose that we are provided data that are six observations

$$X_1 = \begin{bmatrix} 2 & 1 \\ 2 & 2 \\ 3 & 2 \\ 3 & 3 \\ 4 & 4 \\ 4 & 5 \end{bmatrix} \tag{35.2}$$

If we attempt to use *k*-means clustering of the data in Equation 35.2, we have the original data and the clusters that are shown in Figure 35.2. The clustering is visually poor and we can proceed to explore a graph-based method.

From Equation 35.2, we can find the distances between observation i and observation j using the ℓ^2 vector norm. This distance matrix is

$$C_1 = \begin{bmatrix} 0.00 & 1.00 & 1.41 & 2.24 & 3.61 & 4.47 \\ 1.00 & 0.00 & 1.00 & 1.41 & 2.83 & 3.61 \\ 1.41 & 1.00 & 0.00 & 1.00 & 2.24 & 3.16 \\ 2.24 & 1.41 & 1.00 & 0.00 & 1.41 & 2.24 \\ 3.61 & 2.83 & 2.24 & 1.41 & 0.00 & 1.00 \\ 4.47 & 3.61 & 3.16 & 2.24 & 1.00 & 0.00 \end{bmatrix} \tag{35.3}$$

(A) **(B)**

Figure 35.2: Six 2D observations and *k*-means clustering. (A) The data are shown as black circles. (B) The two clusters are shown in red and blue. One observation, shown in blue, is visually not appropriately clustered.

For a distance matrix C, the ε-neighborhood adjacency matrix can be defined as having entries that are

$$a_{ij} = \begin{cases} 1 & \text{if } (c_{ij} > 0) \wedge (c_{ij} \leq \varepsilon) \\ 0 & \text{otherwise} \end{cases} \tag{35.4}$$

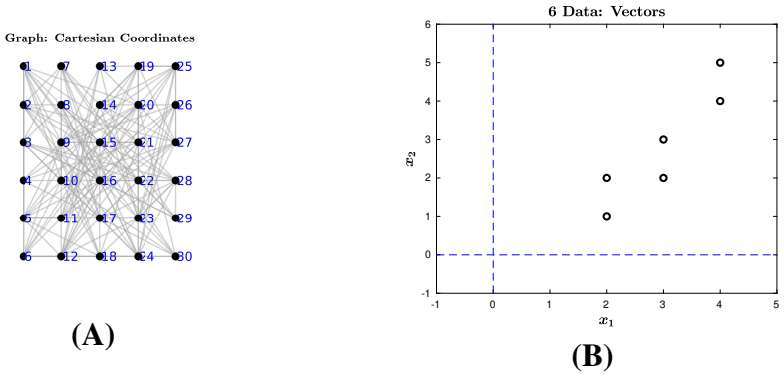
We can use Equation 35.4 create an ε-neighborhood adjacency matrix from the distance matrix C_1 in Equation 35.3, which for a threshold of $\varepsilon = 2$ would be

$$A_1 = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{35.5}$$

The Laplacian matrix K_1 for A_1 in Equation 35.5, computed using Equation 35.1, and the last two eigenvectors of K_1 as a matrix E_1, would be

$$K_1 = \begin{bmatrix} 2 & -1 & -1 & 0 & 0 & 0 \\ -1 & 3 & -1 & -1 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 \\ 0 & -1 & -1 & 3 & -1 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix} \qquad E_1 = \begin{bmatrix} -0.39 & 0.41 \\ -0.31 & 0.41 \\ -0.31 & 0.41 \\ -0.09 & 0.41 \\ 0.39 & 0.41 \\ 0.70 & 0.41 \end{bmatrix} \tag{35.6}$$

If we use the Fiedler vector to cluster the observations in the matrix E_1 in Equation 35.6, we have the result shown in Figure 35.3. For these data, graph clustering is superior to *k*-means clustering of the observations.



(A) **(B)**

Figure 35.3: Six 2D observations and *k*-means clustering of the Laplacian matrix. (A) The data are shown as black circles. (B) The two clusters are shown in red and blue; these data are, visually, in appropriate clusters.

When we are asked to find more than two clusters in data, we can use our method of spectral clustering by transforming the data to a graph and clustering the "last" eigenvectors of the Laplacian matrix of the graph.

Suppose that we are given two observation in addition to those in Equation 35.2. The new data matrix is

$$X_2 = \begin{bmatrix} 2 & 1 \\ 2 & 2 \\ 3 & 2 \\ 3 & 3 \\ 4 & 4 \\ 4 & 5 \\ 2 & 4 \\ 2 & 5 \end{bmatrix} \tag{35.7}$$

We can use *k*-means clustering to try to find 3 clusters, which would give the results shown in Figure 35.4. These results are visually disappointing, so we might next try to use spectral clustering for these data.

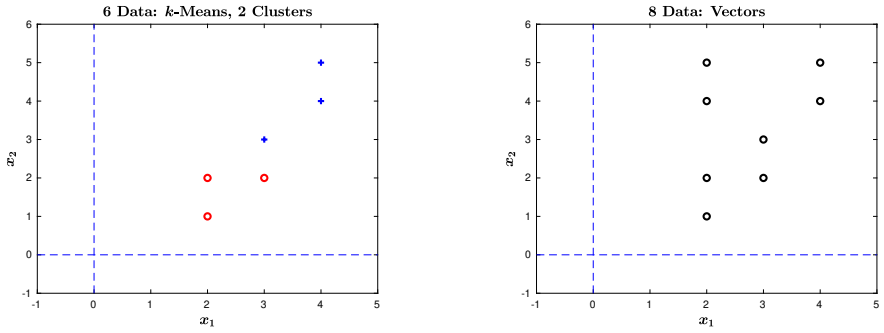


Figure 35.4: Eight 2D observations and k-means clustering. (A) The data are shown as black circles. (B) The three clusters are shown in red, blue, and magenta. One observation, shown in blue, is visually not appropriately clustered.

We can use create a distance matrix B_2 from Equation 35.7, use a threshold of $\varepsilon = 2$ to create an adjacency matrix A_2, and use Equation 35.1 to create the Laplacian matrix K_2 and eigenvector matrix E_2 that are

$$K_2 = \begin{bmatrix} 2 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 3 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix} \quad E_2 = \begin{bmatrix} -0.49 & -0.00 & -0.35 \\ -0.37 & -0.00 & -0.35 \\ -0.37 & -0.00 & -0.35 \\ -0.08 & -0.00 & -0.35 \\ 0.23 & 0.37 & -0.35 \\ 0.43 & 0.60 & -0.35 \\ 0.23 & -0.37 & -0.35 \\ 0.43 & -0.60 & -0.35 \end{bmatrix} \tag{35.8}$$

The Laplacian matrix K_2 of Equation 35.8 can be graphed, an example of which is shown in Figure 35.5.

We can see, from Figure 35.5, that the data are better partitioned when one cluster is the four observations with the lowest x_2 values and the other four observations are in two clusters that are separated by their x_1 values.

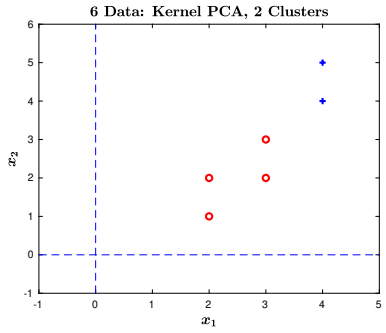


Figure 35.5: Eight 2D observations and a graph that is derived from the ε-neighborhood adjacency matrix using a threshold distance of $\varepsilon = 2$.

35.3 Weighted Graphs

Our spectral clustering method can be extended to a *weighted* adjacency matrix. Our original concept of an adjacency matrix was that its entries were 0 if vertices were not adjacent and 1 if the vertices were adjacent. We can extend this concept by permitting an adjacency to have a non-negative weight; for vertices i and j, the adjacency weight must have the property

$$w_{ij} \geq 0 \tag{35.9}$$

Weighted graphs, and many other related topics, are discussed in the tutorial by Von Luxburg [15].

Extra Notes

35.4 Extra Notes on The Laplacian Matrix

Consider a weight matrix $W \in \mathbb{R}^{m \times m}$, which could be an adjacency matrix, with these properties:

- W has non-negative entries, so $w_{ij} \geq 0$
- W is symmetric, so $w_{ij} = w_{ji}$

From W we can create a diagonal degree matrix D, which we define as having entries

$$d_{ij} \stackrel{\text{def}}{=} \begin{cases} \sum_{i=1}^{m} w_{ij} & \text{if} \quad i = j \\ 0 & \text{if} \quad i \neq j \end{cases} \tag{35.10}$$

For any $\vec{u} \in \mathbb{R}^m$, we can compute the quadratic form $\vec{u}^T D \vec{u}$ by a double summation, for which the summation indexes can be commuted because W is symmetric, as

$$\vec{u}^T D \vec{u} = \sum_{i=1}^{m} d_{ii} u_i^2 = \sum_{i=1}^{m} \sum_{j=1}^{m} w_{ij} u_i^2 = \sum_{i=1}^{m} \sum_{j=1}^{m} w_{ij} u_j^2 \tag{35.11}$$

The Laplacian matrix of W is defined as the difference between the degree matrix and the weight matrix, which is

$$K \stackrel{\text{def}}{=} D - W \tag{35.12}$$

Theorem: The Laplacian matrix is positive semidefinite

For any symmetric positive $W \in \mathbb{R}^{m \times m}$ that has a degree matrix D as defined in Equation 35.10,

$$[D - W] \succeq 0 \tag{35.13}$$

Proof:

It suffices to show that, for any $\vec{u} \in \mathbb{R}^m$ and K as defined in Equation 35.12, the quadratic form is non-negative. We can write the quadratic form, using the property in Equation 35.11, as

$$\begin{aligned}
\vec{u}^T K \vec{u} &= \vec{u}^T D \vec{u} - \vec{u}^T W \vec{u} \\
&= \sum_{i=1}^{m} d_{ii} u_i^2 - \sum_{i=1}^{m} \sum_{j=1}^{m} w_{ij} u_i u_j \\
&= \frac{1}{2} \left(2 \sum_{i=1}^{m} d_{ii} u_i^2 - 2 \sum_{i=1}^{m} \sum_{j=1}^{m} w_{ij} u_i u_j \right) \\
&= \frac{1}{2} \left(\sum_{i=1}^{m} \sum_{j=1}^{m} w_{ij} u_i^2 + \sum_{i=1}^{m} \sum_{j=1}^{m} w_{ij} u_j^2 - 2 \sum_{i=1}^{m} \sum_{j=1}^{m} w_{ij} u_i u_j \right) \\
&= \frac{1}{2} \left(\sum_{i=1}^{m} \sum_{j=1}^{m} \left(w_{ij} u_i^2 + w_{ij} u_j^2 - 2 w_{ij} u_i u_j \right) \right) \\
&= \frac{1}{2} \left(\sum_{i=1}^{m} \sum_{j=1}^{m} w_{ij} \left(u_i^2 + u_j^2 - 2 u_i u_j \right) \right) \\
&= \frac{1}{2} \left(\sum_{i=1}^{m} \sum_{j=1}^{m} w_{ij} \left(u_i - u_j \right)^2 \right)
\end{aligned} \tag{35.14}$$

The right-hand side of Equation 35.14 is the sum of products of $w_{ij} \geq 0$ and $(u_i - u_j)^2 \geq 0$, so

$$\vec{u}^T K \vec{u} \geq 0 \tag{35.15}$$

Equation 35.15 is equivalent to Equation 35.13 so our proof is complete.