Spectral Clustering on Handwritten Digits Database

Danielle Middlebrooks
dmiddle1@math.umd.edu

Advisor: Kasso Okoudjou
kasso@umd.edu
Department of Mathematics

University of Maryland- College Park
Advance Scientific Computing I

October 6, 2015

Outline

Background Information

- Spectral Clustering is clustering technique that makes use of the spectrum of the similarity matrix derived from the data set.
- Implements a clustering algorithm on a reduced dimension.
- Advantages: Simple algorithm to implement and uses standard linear algebra methods to solve the problem efficiently.
- Motivation: Implement an algorithm that groups objects in a data set to other objects with ones that have a similar behavior.

Definitions

- A graph $G = (V, E)$ where $V = \{v_1, ..., v_n\}$
- W- Adjacency matrix.

$$w_{ij} = \begin{cases} 1, & \text{if } \; v_i, \; v_j \text{ are connected by an edge} \\ 0, & \text{otherwise} \end{cases}$$

- The degree of a vertex $d_i = \sum_{j=1}^{n} w_{ij}$. The Degree matrix denoted D,where each $d_1, ..., d_n$ are on the diagonal.
- Denote a subset of vertices $A \subset V$ and the compliment $\bar{A} = V \setminus A$
- $|A|$ = number of vertices in A
- $vol(A) = \sum_{i \in A} d_i$
- $W(A, B) = \sum_{i \in A, j \in B} w_{ij}$

Example

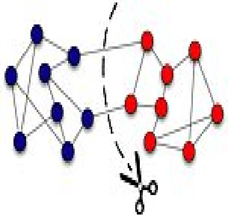
$G = (V, E)$ where $V = \{1, ..., 7\}$.

$$W = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} D = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 \end{bmatrix}$$

Definitions

- Similarity graph: Given a data set $x_1, \ldots, x_n$ and a notion of "similar", a similarity graph is a graph where x_i and x_j have an edge between them if they are considered "similar". Some ways to determine if data points are similar are:
  - e-neighborhood graph
  - k-nearest neighborhood graph
  - Use Similarity Function
- Unnormalized Laplacian Matrix: $L = D - W$
- Normalized Laplacian Matrix: $L_{sym} = D^{-1/2} L D^{-1/2} = I - D^{-1/2} W D^{-1/2}$

Why this works?



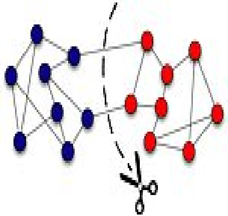
Spectral Clustering is motivated by approximating the RatioCut or NCut on a given graph.

- Given a similarity graph, to construct a partition is the solve the min cut problem. That is

$$min\ cut(A_1, ...A_k) := \frac{1}{2}\sum_{1}^{k} W(A_i, \bar{A}_i)$$

- In order to insist each partition is reasonably large, use RatioCut or NCut. Thus the size of each partition is measured by the number of vertices or weights of the edges, respectively.

Why this works?



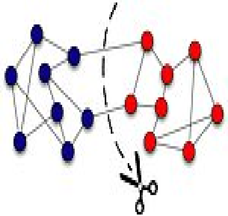
- Thus

$$RatioCut(A_1, ..., A_k) := \frac{1}{2} \sum_{i=1}^{k} \frac{W(A_i, \bar{A}_i)}{|A_i|}$$

$$NCut(A_1, ..., A_k) := \frac{1}{2} \sum_{i=1}^{k} \frac{W(A_i, \bar{A}_i)}{vol(A_i)}$$

- Solving these versions makes the problem NP hard.
- Spectral Clustering solves the relaxed versions of these problems. [2.]

Why this works?



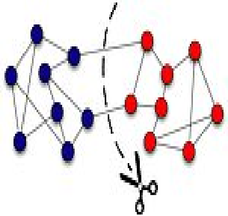
Case $k = 2$. Given a subset A. Then $NCut(A, \bar{A}) = \frac{W(A,\bar{A})}{vol(A)} + \frac{W(A,\bar{A})}{vol(\bar{A})}$. Define the cluster indicator vector f by

$$f(v_i) = f_i = \begin{cases} \frac{1}{vol(A)}, & \text{if} \quad v_i \in A \\ -\frac{1}{vol(\bar{A})}, & \text{if} \quad v_i \in \bar{A} \end{cases}$$

Then

$$f^T L f = \sum w_{ij}(f_i - f_j)^2 = W(A, \bar{A})(\frac{1}{vol(A)} + \frac{1}{vol(\bar{A})})^2$$

$$f^T D f = \sum d_i f_i^2 = \frac{1}{vol(A)} + \frac{1}{vol(\bar{A})}$$



Why this works?

Thus minimizing the NCut problem is equivalent to

$$\min \ NCut(A, B) = \frac{f^T L f}{f^T D f}$$

The relaxation problem is given by

$$\begin{aligned} \underset{f \in \mathbb{R}^n}{\text{minimize}} \quad & \frac{f^T L f}{f^T D f} \\ \text{subject to} \quad & f^T D \mathbf{1} = 0 \end{aligned}$$

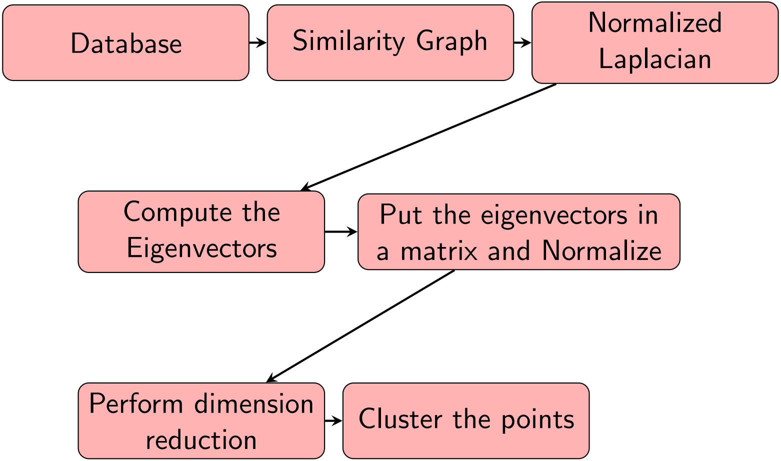
Why this works?

It can be should the relaxation problem is a form of the Rayleigh-Ritz quotient. The Rayleigh Ritz theorem states: Given A a Hermitian matrix, then

$$\lambda_{min} = min_{x\neq 0}\frac{x^T Ax}{x^T x}$$

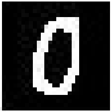
Thus in the relaxation problem, the solution f is the second eigenvector of the generalized problem.

Procedure


Database
Similarity Graph
Normalized Laplacian
Compute the Eigenvectors
Put the eigenvectors in a matrix and Normalize
Perform dimension reduction
Cluster the points

Databases

The database I will be using is the MNIST Handwritten digits database. Has 1000 of each digit 0-9. Each image is of size 28x28 pixels.



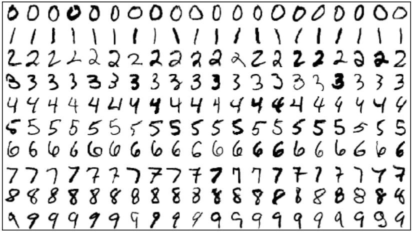
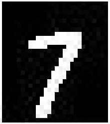
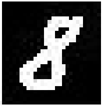


Figure: Test images. Simon A. J. Winder





Similarity Graph

Guassian Similarity Function: $s(x^i, x^j) = e^{\frac{-||x^i - x^j||^2}{2\sigma^2}}$ where σ is a parameter. If $s(x^i, x^j) < \epsilon$ connect an edge between x^i and x^j. Each $x^i \in \mathbb{R}^{28x28}$ and corresponds to an image. Thus

$$||x^i - x^j||_2^2 = \sum_{k=1}^{28}\sum_{l=1}^{28}(x^i_{kl} - x^j_{kl})^2$$

Laplacian Matrix

- W- Adjacency matrix $w_{ij} = \begin{cases} 1, & \text{if } \; s(x^i, x^j) < \epsilon \\ 0, & \text{otherwise} \end{cases}$
- D- Degree matrix
- $L = D - W$
- $L_{sym} = D^{-1/2} L D^{-1/2} = I - D^{-1/2} W D^{-1/2}$

Computing Eigenvectors

Use an iterative method called the Power Method to find the first k eigenvectors of $L_{sym} = D^{-1/2}LD^{-1/2} = I - D^{-1/2}WD^{-1/2}$.

- Start with an initial nonzero vector, v_0, for the eigenvector
- Let $B = D^{-1/2}WD^{-1/2}$. Form the sequence given by:

$$\text{for } i = 1, ..., l$$

$$x_i = Bv_{i-1}$$

$$v_i = \frac{x_i}{||x_i||}$$

$$\text{end}$$

Computing Eigenvectors (Con't)

- For large values of l we will obtain a good approximation of the dominant eigenvector of B.

This will give us the eigenvector corresponding the the largest eigenvalue of B which corresponds to the smallest eigenvalue of L_{sym}.
To find the next eigenvector, after selecting the random initial vector v_0, subtract the component of v_0 that is parallel to the eigenvector of the largest eigenvalue.

Computing Eigenvectors (Con't)

We put the first k eigenvectors into a matrix and normalize it.
Let $T \in \mathbb{R}^{nxk}$ be the eigenvector matrix with norm 1.
Set

$$t_{i,j} = \frac{v_{i,j}}{(\sum_k v_{i,k}^2)^{1/2}}$$

$$\begin{bmatrix} v_{11} & v_{12} & v_{13} & \dots & v_{1k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ v_{i1} & v_{i2} & v_{i3} & \dots & v_{ik} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ v_{n1} & v_{n2} & v_{n3} & \dots & v_{nk} \end{bmatrix} \Rightarrow \begin{bmatrix} t_{11} & t_{12} & t_{13} & \dots & t_{1k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{i1} & t_{i2} & t_{i3} & \dots & t_{ik} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{n1} & t_{n2} & t_{n3} & \dots & t_{nk} \end{bmatrix}$$

Dimension Reduction

Project the eigenvectors onto new space.
Let $y_i \in \mathbb{R}^k$ be a vector from the i^{th} row of T

$$\begin{bmatrix} t_{11} & t_{12} & t_{13} & \dots & t_{1k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{i1} & t_{i2} & t_{i3} & \dots & t_{ik} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{n1} & t_{n2} & t_{n3} & \dots & t_{nk} \end{bmatrix} \Rightarrow y_i = \begin{bmatrix} t_{i1} \\ t_{i2} \\ \vdots \\ t_{ik} \end{bmatrix}$$

Clustering

Perform a k-means algorithm on the set of vectors.

- Randomly select k cluster centroids, z_j.
- Calculate the distance between each y_i and z_j.
- Assign the data point to the closest centroid.
- Recalculate centroids and distances from data points to new centroids.
- If no data point was reassigned then stop, else reassign data points and repeat.

Clustering

Perform a k-means algorithm on the set of vectors.

- Randomly select k cluster centroids, z_j.
- Calculate the distance between each y_i and z_j.
- Assign the data point to the closest centroid.
- Recalculate centroids and distances from data points to new centroids.
- If no data point was reassigned then stop, else reassign data points and repeat.

Finally, assign the original point x^i to cluster j if and only if row i of the matrix Y was assigned to cluster j.

Validation

- Validate the k-means clustering on a well known clustered set. See if they match.
- Validate the computation of the eigenvectors by using Matlab toolbox.
- Validate the solution of clustering found by seeing if similar images are grouped together.

Implementation

- Personal Laptop: Macbook Pro.
- I will be using Matlab R2014b for the coding.

Project Schedule

- End of October/ Early November: Construct Similarity Graph and Normalized Laplacian matrix.
- End of November/ Early December: Compute first k eigenvectors validate this.
- February: Normalize the rows of matrix of eigenvectors and perform dimension reduction.
- March/April: Cluster the points using k-means and validate this step.
- End of Spring semester: Implement entire algorithm, optimize and obtain final results.

Results

By the end of the project, I will deliver

- Code that delivers database
- Codes that implement the algorithm
- Final report of algorithm outline, testing on database and results
- Final presentation

References

[1.] Von Cybernetics, U. A Tutorial on Spectral Clustering. Statistics and Computing, 7 (2007) 4.

[2.] Shi, J. and Malik J. Normalized cuts and image segmentation. IEEE Transations on Pattern Analysis and Machine Intelligence, 22 (2000) 8.

[3.] Chung, Fan. Spectral Graph Theory. N.p.: American Mathematical Society. Regional Conference Series in Mathematics. 1997. Ser. 92.

[4.] Vishnoi, Nisheeth K. $Lx = b$ Laplacian Solvers and their Algorithmic Applications. N.p.: Foundations and Trends in Theoretical Computer Science, 2012.

Thank you