Spectral Clustering on Handwritten Digits Database
Mid-Year Presentation

Danielle Middlebrooks
dmiddle1@math.umd.edu

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

University of Maryland- College Park
Advance Scientific Computing I
December 10, 2015

Outline

Background Information

- Spectral Clustering is clustering technique that makes use of the spectrum of the similarity matrix derived from the data set.
- Motivation: Implement an algorithm that groups objects in a data set to other objects with ones that have a similar behavior.

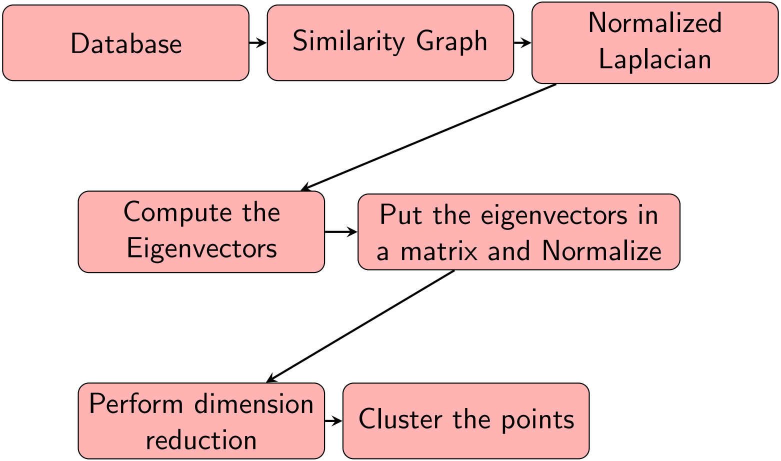
Definitions

- A graph $G = (V, E)$ where $V = \{v_1, ..., v_n\}$
- W- Adjacency matrix.
  $$W(i,j) = \begin{cases} 1, & \text{if } v_i,\ v_j \text{ are connected by an edge} \\ 0, & \text{otherwise} \end{cases}$$
- 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.

Definitions

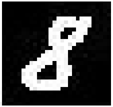
- Similarity graph: Given a data set $X_1, ..., X_n$ and a notion of "similar", a similarity graph is a graph where X_i and X_j have an edge between them if they are considered "similar". Some ways to determine if data points are similar are:
  - e-neighborhood graph
  - k-nearest neighborhood graph
  - Use Similarity Function
- Unnormalized Laplacian Matrix: $L = D - W$
- Normalized Laplacian Matrix: $L_{sym} = D^{-1/2}LD^{-1/2} = I - D^{-1/2}WD^{-1/2}$

Procedure


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

Database

- The database I will be using is the MNIST Handwritten digits database.
- The test set has 1000 of each digit 0-9. Each image is of size 28×28 pixels .
- Each image read into a 4-array $t(28, 28, 10, 1000)$



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

$$||X_i - X_j||_2^2 = \sum_{k=1}^{28} \sum_{l=1}^{28} (X_i(kl) - X_j(kl))^2$$

Implementation

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

Normalized Laplacian Matrix

Normalized Laplacian Algorithm

Set parameters: $n1$, $n2$, N, D, σ, ϵ.
Compute $||X_i - X_j||^2$ between any two images
Compute the Gaussian Similarity function $e^{\frac{-||x_i - x_j||^2}{2\sigma^2}}$
if similarity $> \epsilon$
    set $W(i,j)$ to 1
else as 0
D1=diag(sum(W,2) .^(-1/2))
B=D1*W*D1

Validation of Normalized Laplacian

Since we know the smallest eigenvalue of the Unnormalized laplacian will be zero with eigenvector $\mathbb{1}$, we can validate our computation of the Unnormlized laplacian or equivalently the Normalized laplacian with eigenvector $D^{1/2}\mathbb{1}$

Validation of Normalized Laplacian

Since we know the smallest eigenvalue of the Unnormalized laplacian will be zero with eigenvector $\mathbb{1}$, we can validate our computation of the Unnormlized laplacian or equivalently the Normalized laplacian with eigenvector $D^{1/2}\mathbb{1}$ ✓

Computing first K Eigenvectors

Power Method Algorithm (A)

Start with an initial nonzero vector, v_0.Set tolerance, max iteration and iteration$= 1$

Repeat

$v_0 = A * v_0$;

$v_0 = v_0/norm(v_0, 2)$;

lambda$= v_0' * A * v_0$;

converged $=$ (norm($A * v_0 -$ lambda$*v_0$,2) $<$ tol);

iter=iter+1;

if iter $>$ maxiter

warning('Did Not Converge')

Until Converged

Computing first K Eigenvectors (Con't)

Deflation Algorithm

Initialize d = length(A); V = zeros(d,K); lambda=zeros(K,1);
for j from $1, \ldots, K$
  [lambda(j), V(:,j)] = power-method(A,v_0);
  $A = A-$ lambda(j)$*V(:,j) * V(:,j)'$;
  $v_0 = v_0 - \frac{v_0 \cdot V(:,j)}{v_0 \cdot v_0} * v_0$
end

Challenges

$$L_{sym} = I - D^{-1/2}WD^{-1/2} = I - B$$

- In using the power method we want to ensure that our matrix is positive semidefinite in order to efficiently compute the eigenvalues.
- Add a multiple of the Identity to B
- Choose parameters σ and ϵ in order to ensure this

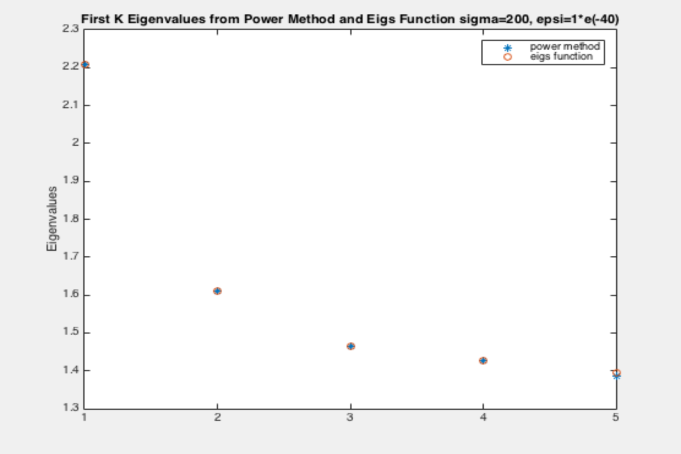
Adjusting B Matrix

Theorem

A Hermitian diagonally dominant matrix A with real non-negative diagonal entries is positive semidefinite.

Let $B_{mod} = B + \mu I$
If we let $\mu = \max(\text{sum}(B,2))$, this will allow B_{mod} to be positive semidefinite.

Eigenvalues Found



Eigenvectors Found

	λ_1	λ_2	λ_3	λ_4	λ_5
r	1.05E-10	9.54E-7	4.11E-1	7.30E-1	6.83E-1

$r =$norm$(\frac{B}{\lambda}v - \frac{B}{\lambda*}v*$,2)

(λ, v) came from power method

$(\lambda*, v*)$ came from eigs function

Computational Time

- Computing Normalized Laplacian (10,000 images) $\sim$ 25 mins
- Computing eigenvectors using power method with deflation (5,000 images) $\sim$ 18 secs
- Computing eigenvectors using eigs function (5,000 images) $\sim$ 7 secs

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 entire 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