

Scalability of Spectral Clustering

Course: CSE-5095-Big Data Analytics
Prof. Fei Wang

- Neha Gadigi & Spandana Vallabhaneni

Introduction

- Clustering :
 - method of grouping a set of objects.
 - Similar objects into one group
 - Dissimilar objects into different groups
 - Unsupervised learning – No predefined classes
- Traditional clustering algorithms work well only on convex shape data.
- Spectral clustering works well on non-convex shaped data
 - Pair wise similarity.

General Spectral Clustering

Construction of Similarity graph : Local neighborhood relations

\mathcal{E} – neighborhood

K-nearest neighbors

Fully connected

Similarity function: Gaussian kernel function = $w(i, j) = e^{\frac{-||x_i^2 - x_j^2||}{2\sigma^2}}$



Construct Laplacian Matrix L (Normalized or Un-Normalized)



Eigen Value Decomposition on L



Choose K smallest eigenvectors to define a K-dimensional subspace



Cluster data points in this subspace using K-means

Spectral Clustering

- Many versions of Spectral Clustering algorithms have been proposed with slight variances to the general method.
 - Different Laplacian Matrices
 - $L = D - W \rightarrow$ Un-normalized
 - $L_{sym} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}} \rightarrow$ Normalized (symmetric)
 - $L_{rw} = D^{-1} L \rightarrow$ Normalized (Close to Random-walk)
- Obj1: minimize inter-cluster similarity
- Obj2: maximize intra-cluster similarity
- Issues :
 - Choosing the scaling parameter, sigma (σ)
 - Value of k – number of clusters

Spectral Clustering

- Issues :
 - choosing the scaling parameter sigma
 - Local scaling parameter
 - $A = e^{\frac{-d^2(s_i - s_j)}{\sigma_i \sigma_j}}$
 - $\sigma_i = d(s_i, s_K)$
 - Value of k – number of clusters
 - Eigen gap heuristic – difference between two consecutive eigenvalues
 - $\nabla_k = |\lambda_k - \lambda_{k+1}| \rightarrow$ value of k which maximizes
 - Works well when data contains good clusters

Scalability

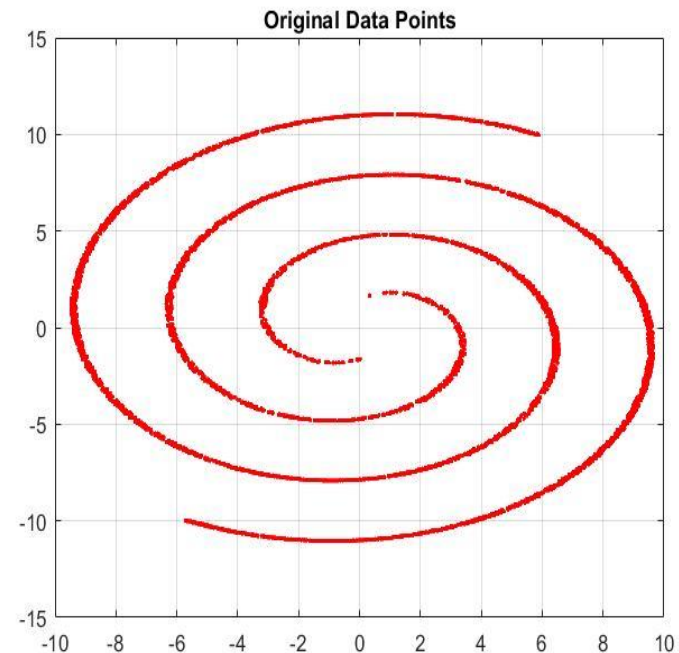
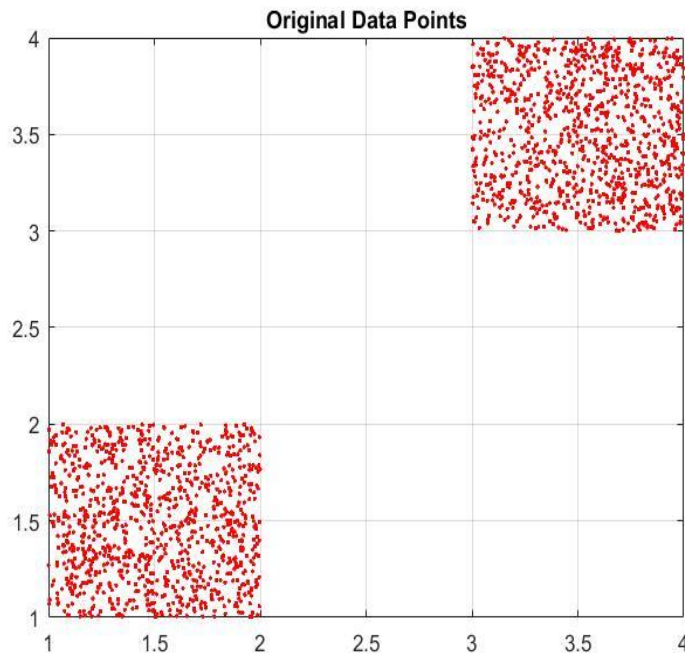
- Fails to work efficiently on large datasets
 - Construction of similarity matrix and storing it : Memory inefficient $O(n^2)$.
 - Eigen Value Decomposition :
 - Computationally expensive $O(n^3)$.

Scalability Approaches

- Goal: How many data points can it scale up to on a single machine.
- Basic Approaches:
 - Sample-out p points as representative data points ($p \ll n$).
 - Zero out some elements in the similarity matrix - Sparse
 - Call sparse Eigen solver – calculate k largest or smallest eigenvectors.
 - Lanczos method: It is direct method used to compute k outermost eigen values and eigenvectors approximately.
 - Arnoldi Iterative: An iterative method used to find out most useful eigenvalues and eigenvectors with limited number of operations.
 - Iterative methods: procedure which generates a sequence of improving approximate solutions.

Tools and Data

- Matlab R2015 a
- Processor specifications: Intel®Core™ i7-3540M CPU @ 3.00GHz 3.00 GHz, RAM: 8.00 GB System type: 64-bit Operating System.
- Synthetic Data

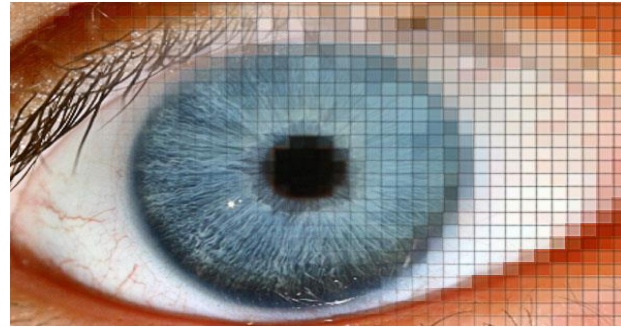


Data

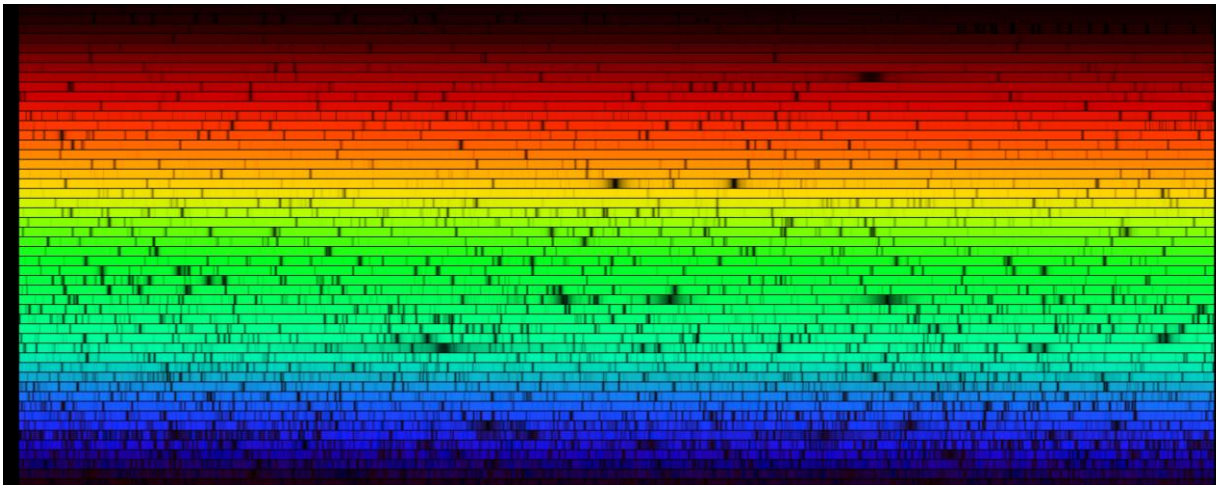
- Real Data - Images



- Points = 6,670
- Pixel = $130 * 132$



Points = 82,893
Pixel = $600 * 375$



Points = 461,847
Pixel = $5464 * 8192$

NJW Algorithm



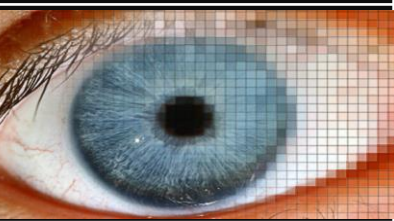
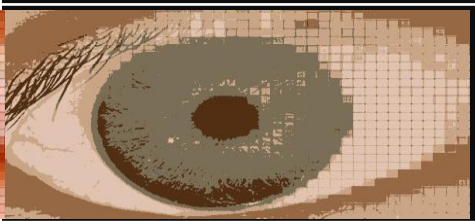
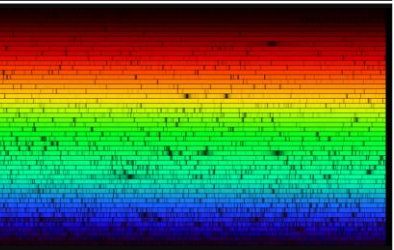
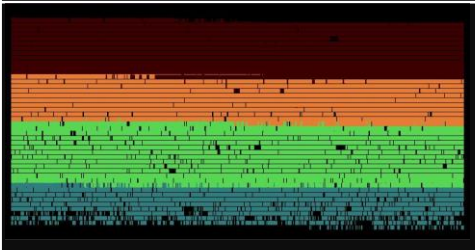
- Ng, Jordan and Weiss : $L_{sym} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}}$
- Experimentation results on synthetic dataset:

Data points	Time	Comment
1000	2mins	Successful
5000	3.35 mins	Successful
10000	85% memory used & 40% CPU 25 minutes	Successful
20000		System crash

Parallel and Distributed Spectral Clustering (PSC)

- Parallel and Distributed Spectral Clustering:
 - Construct sparse similarity matrix using k-nearest neighbors
 - Call Sparse Eigen Solver – Arnoldi Factorization(eigs())
 - Cluster using k-means
 - Reason for using the above:
 - Efficient construction of similarity matrix by dividing the data matrix into blocks and process the blocks sequentially, which alleviates the memory use
 - Its an NJW algorithm, just using a sparse similarity matrix and sparse eigensolver improved the scalability.
- Results on Synthetic data:
Scaled up to 290,000 points – 3hours.




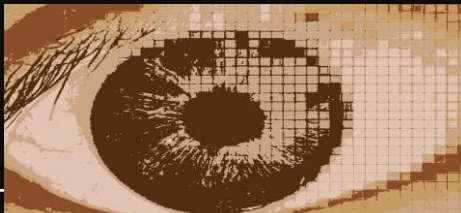
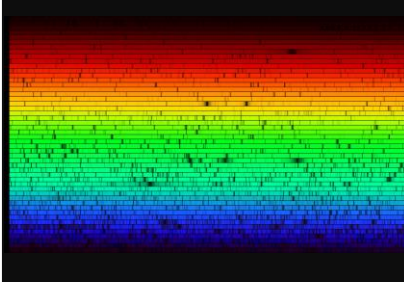
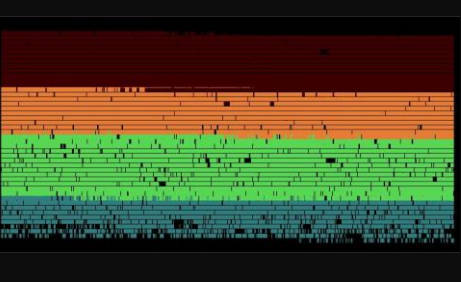
Results

Scenario	After clustering	Data points	No. of unique data points	Simulation time	No. of clusters
		17,160	6,670	8 sec	5
		225,000	82,893	9 mins	5
		44,761,088	461,847	7Hrs (aprox)	5




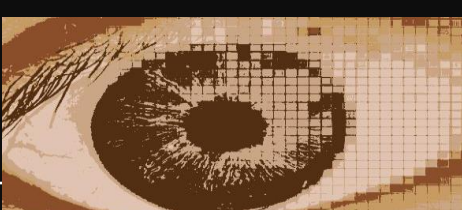
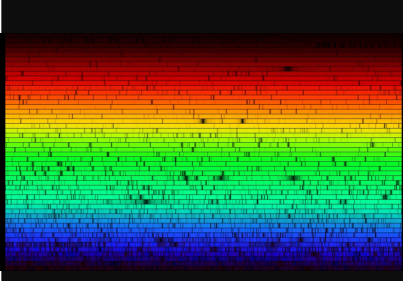
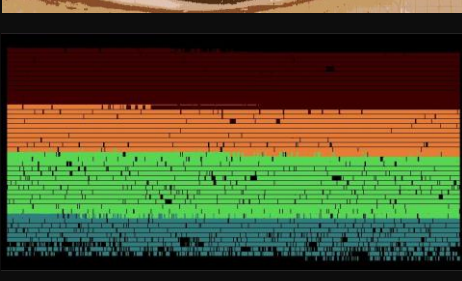
Landmark Based Spectral Clustering

- p data points ($p \ll n$) are chosen as landmarks and represent the original data points as linear combinations of landmark.
 - Algorithm:
 - ❖ Construct affinity matrix between data points and the chosen landmark points (using k-means or random sampling)
 - ❖ Carry out Eigen value decomposition
 - ❖ Cluster the data points in subspace using k-means
- Scales linearly with size of the problem.

Results (Landmark Based SC using k-means)

Scenario	After clustering	No. of unique data points	Simulation time	No. of clusters
		6,670	1.4 sec	5
		82,893	14 sec	5
		461,847	Out of memory issue	5

Results (Landmark Based SC using Random Sampling)

Scenario	After clustering	No. of unique data points	Simulation time	No. of clusters
		6,670	2 sec	5
		82,893	30 sec	5
		461,847	Out of memory issue	5

Comparison of Results

Data points	Parallel SC	LSC using K-means	LSE using random sampling
6,670	8 seconds	1.4 seconds	2 seconds
82,893	9 minutes	14 seconds	30 seconds
461,847	7 hours	Out of memory	Out of memory

LSC failed as the system ran out of memory while constructing pairwise distances using k-means, k-means suffered from the scalability of large dataset that was used.

Future Work:

To work on k-means issue in LSC by dividing the construction of pairwise distance matrix into blocks as the time taken by LSC to cluster points is far less when compared to PSC.

Conclusion

- Efficiently constructs the similarity matrix by dividing the dataset into blocks which alleviates the memory use.
- Sparsify the similarity matrix and it requires less memory to store.
- Call a sparse Eigen solver.
- We applied Parallel Spectral Clustering and Landmark based spectral clustering on both synthetic data as well as images.
- Compared the performance of two algorithms.

Disadvantages of Approximation techniques:

- Approximation techniques have risk of loss of data associated with them.
- Trade-off between Scalability of clustering and quality of clustering



QUESTIONS!!