IE 529 - FINAL PROJECT

EFFICIENT SPECTRAL CLUSTERING: A COMPARATIVE STUDY OF CLASSICAL AND FAST SPECTRAL CLUSTERING METHODS

Sarath Saroj - ssaroj2

Suvrata Gayathri Kappagantula - sk108

Safin Akash - santon21

May 9, 2024

Abstract

In our exploration, we assess the efficiencies of classical and fast spectral clustering techniques using the MNIST dataset as a benchmark. By implementing and rigorously comparing these methodologies, we gauge their performance across execution times and clustering metrics such as the Adjusted Rand Index (ARI) and Normalized Mutual Information (NMI). The results illuminate the substantial computational speedup achieved by the fast spectral clustering method without compromising, and occasionally enhancing, the clustering quality.

1 INTRODUCTION

Background Spectral clustering excels in identifying intrinsic groupings within complex datasets by utilizing principles of graph theory. This method distinguishes itself from traditional approaches like k-means by employing matrix eigenvectors for dimensionality reduction, significantly improving its efficiency in revealing intricate data structures. This makes it especially suitable for applications such as image segmentation, social network analysis, and bioinformatics.

Problem Statement While spectral clustering is good at managing non-convex clusters, it is inherently computationally demanding, particularly with large datasets. The primary computational challenge lies in calculating the eigenvalues and eigenvectors of the Laplacian matrix, which does not scale well with increased data size, thus restricting its practical deployment in large-scale applications.

This project draws inspiration from a recent paper presented by Peter Macgregor, which utilizes the power method to reduce computational demands significantly. By leveraging these innovative approaches, we aim to explore practical solutions that maintain the algorithm's effectiveness while minimizing its computational overhead.

Objective This project introduces and evaluates an innovative spectral clustering technique that uses a power method to approximate eigenvectors of the Laplacian matrix in the form of dominant vectors, reducing computational complexity drastically. Our study involves comparing this power method against traditional spectral clustering to assess differences in execution times, clustering accuracy, and robustness, employing the MNIST dataset of handwritten digits to test these attributes under complex data conditions.

2 LITERATURE REVIEW

Traditional Spectral Clustering Spectral clustering is renowned for its capacity to uncover complex patterns within data, attributed to its foundation in graph theory. The seminal work by Ng, Jordan, and Weiss outlines a framework that involves constructing a similarity graph, calculating the graph's Laplacian, and clustering the derived eigenvectors using standard algorithms like k-means. Significant advancements have been made since the initial conceptualization of spectral clustering. Ng, Jordan, and Weiss, in their pivotal 2001 study, outlined an approach that fundamentally integrated graph theory into clustering techniques, setting a standard for subsequent research. Their method emphasized optimizing the cut on the graph to segment data into distinct groups, a principle that remains integral to the most advanced forms of spectral clustering today.

Computational Challenges The computational intensity of deriving the Laplacian's eigenvalues and eigenvectors often renders spectral clustering impractical for large datasets or real-time applications. Efforts to mitigate these challenges focus on reducing dimensionality and enhancing eigen-decomposition efficiency or finding alternative methods to eigen decomposition.

Advances in Fast Spectral Clustering Recent innovations propose methods that either approximate the traditional eigen-decomposition steps or bypass them entirely, significantly reducing computational demands. Notably, the power method has emerged as an effective technique for approximating dominant eigenvalues and their corresponding eigenvectors, streamlining the spectral clustering process.

Contribution of Peter Macgregor Peter Macgregor's innovative adaptation of the power method significantly diminishes the time complexity of spectral clustering while preserving clustering accuracy. This method, which computes a low-dimensional vertex embedding using logarithmically few random vectors, enhances both the speed and applicability of spectral clustering.

3 METHODOLOGY

3.1 Data Description

The MNIST dataset, comprising 70,000 images of handwritten digits, provides a robust platform for demonstrating the efficiency of spectral clustering techniques. Each 28x28 pixel image is transformed into a 784-dimensional vector, representing complex and high-dimensional data ideal for this study.

3.2 Classical Spectral Clustering Implementation

- **Data Preprocessing**: Standardization is employed to normalize the data, ensuring uniformity in feature contribution.

- **Graph Construction**: We construct a similarity graph using Gaussian similarity function and/or k nearest neighbor structure based on the data set.

- **Laplacian Matrix and Eigen Decomposition**: The graph's Laplacian is computed and decomposed to obtain spectral embeddings.

- **Clustering**: The k-means algorithm is applied to cluster the embeddings into groups representing the different digit classes.

3.3 Fast Spectral Clustering Implementation

- **Power Method for Spectral Embedding**: Instead of calculating numerous eigenvectors, the power method approximates the spectral embedding using logarithmically few random vectors. This step significantly reduces the computation time by focusing on a smaller, representative subset of vectors.

- **Projection and Normalization**: The vectors (dominant vectors) obtained from the power method are normalized and used directly for clustering, eliminating the need for further eigen-decomposition.

- **Clustering with k-means**: Similar to the classical method, k-means clustering is then applied to these approximated embeddings to determine the cluster memberships of the data points.

Peter Macregor, in his paper, discusses the algorithmic complexity of the two methods under study. Traditional spectral clustering typically operates with a time complexity of $O(n^3)$ due to the eigen-decomposition of the Laplacian matrix, where $'n'$ represents the number of data points. In contrast, our fast spectral clustering method reduces this complexity significantly by approximating the top eigenvectors through the power method, which can operate in sub-quadratic time under certain conditions.

3.4 Pseudocode Explanation

The pseudo code for classical spectral clustering is not discussed since we have already discussed it during the lectures and implemented it in one of the assignments. The following pseudocode outlines the fast spectral clustering algorithm, emphasizing the simplicity and efficiency of the process. This has 2 parts, The Power Method algorithm to find dominant vectors and the main clustering part.

Algorithm 1 POWER METHOD

```
Input: Signless Laplacian ($M$), Random Vector ($x_0 \in R^n$), number of iterations (t)
Output: Dominant Vectors
for ($i = 1, \ldots, t$) do
    ($x_i = Mx_{i-1}$)
return ($x_t$)
```

This algorithm effectively reduces the computational overhead by simplifying the eigenvector calculation stage, making it feasible for larger datasets. Each step is designed to maintain the integrity of the spectral clustering approach while enhancing speed and reducing computational demands.

Algorithm 2 FAST SPECTRAL CLUSTERING

```
Input: Graph ($G(V, E)$), number of clusters ($k$)
Output: Clustering labels
Compute the normalized graph Laplacian, ($L = D^{-1/2}A * D^{-1/2}$)
Compute the Signless Laplacian by using ($N = I - 0.5 \times L$)
Initialize random vectors ($x_0$) of size ($O(\log(k))$)
for ($i = 1$) to ($\log(k)$) do
    ($x_i = \text{PowerMethod}(L, x_0, \text{iterations})$)
Form matrix ($Y$) from vectors ($x_i$)
Cluster ($Y$) using k-means into ($k$) clusters
return Clustering labels
```

4 IMPLEMENTATION AND RESULTS

To validate the theoretical advantages of fast spectral clustering, we implemented both classical and fast spectral clustering techniques using Python, leveraging popular libraries such as Scikit-learn for machine learning tasks and NumPy for numerical computations. Before analysing the MNIST dataset we implemented the method on the 'Shaped Data.csv' provided to us as a dataset for the computational assignment 2. We added this step to visualise the clustering performance (since ShapedData.csv is a 2-D data).

This section details the implementation process and showcases the results, demonstrating the practical efficacy of the fast clustering method.

5 CODE IMPLEMENTATION

Our approach involved several key steps, detailed below:

- **Data Preprocessing**: We started by loading the MNIST dataset, a collection of handwritten digits widely used in machine learning benchmarks. The dataset was normalised by dividing the pixel values of each 784 pixel by 255 (since the pixel value ranges from 0-255) to ensure each feature contributed equally.

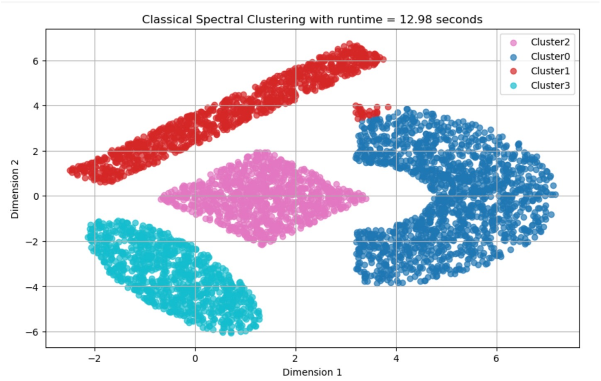
- **Classical Spectral Clustering**: We constructed a similarity graph using K-Nearest Neighbours and computed the Laplacian matrix, followed by its eigen decomposition to derive spectral embeddings. These embeddings were then clustered using the k-means algorithm.

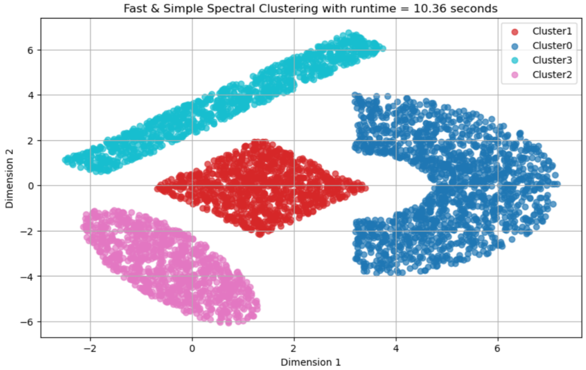
- **Fast Spectral Clustering**: Leveraging the power method, we approximated the spectral embeddings more efficiently. This involved fewer computations by approximating the top eigenvectors directly from the normalized Laplacian matrix, thus reducing both time and resource usage.

6 RESULTS AND DISCUSSION

Visualising clustering results and runtime complexity on a 2-D Data - ShapedData.csv

We prepared the Laplacian using a weighted adjacency matrix created using Gaussian Similarity Function, with a k-nearest neighbor structure. We selected 100 nearest neighbors and sigma =2 for the Gaussian Similarity Function.

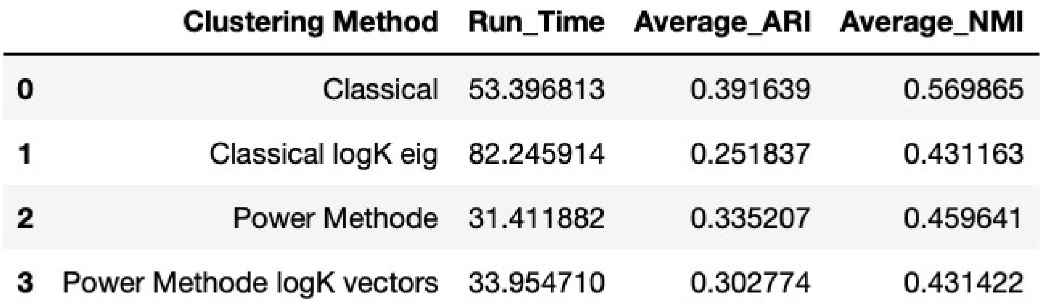




We got a clustering performance as good as/ better than the classical clustering algorithm at a shorter run time while implementing the power method. This visualisation plot attached shows that the power method doesn't deviate significantly in terms of clustering accuracy as compared to the classical clustering algorithm.

6.1 Performance Metrics Based on MNIST Data

The implementation of both classical and fast spectral clustering methods on the MNIST dataset revealed:

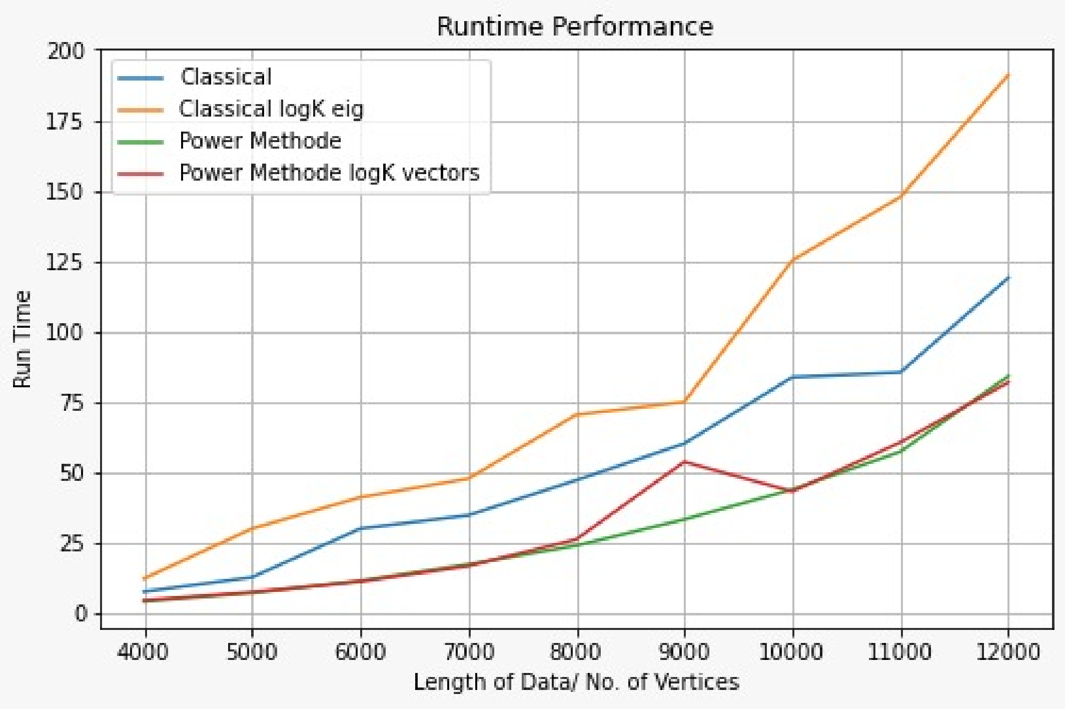


	Clustering Method	Run_Time	Average_ARI	Average_NMI
0	Classical	53.396813	0.391639	0.569865
1	Classical logK eig	82.245914	0.251837	0.431163
2	Power Methode	31.411882	0.335207	0.459641
3	Power Methode logK vectors	33.954710	0.302774	0.431422

- **Fast Spectral Clustering (with $log(k)$ or k dominant vectors)** has a significant performance improvement in runtime complexity (performing equally well) compared to classical clustering and the difference becomes highly dominant as the number of data points increases.
- **Classical Spectral Clustering with $log(k)$ eigenvectors**performs worse than with **k eigenvectors**
- **ARI/NMI scores** are not stable and fluctuate randomly during the experiment. We believe that we need to do a better tuning of the affinity matrix/similarity matrix going forward. But the average ARI and NMI performance show that the clustering accuracy doesn't drop significantly by using the power method.

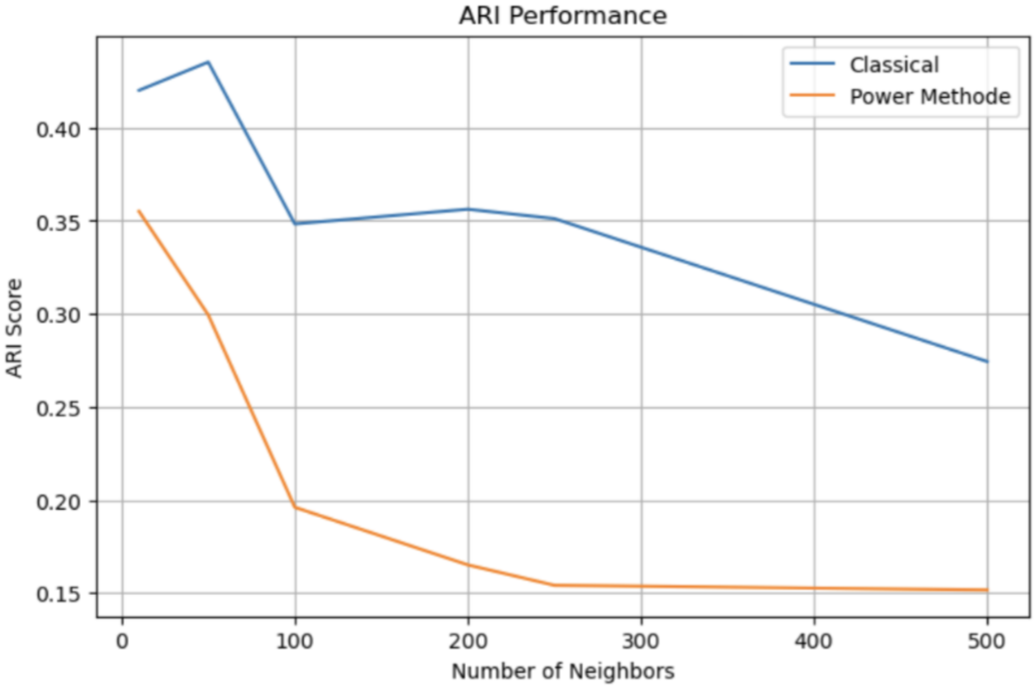
6.2 Time Complexity Analysis

The below line plot compares the time efficiency of the classical and power method spectral clustering algorithms. The plot for our analysis agrees with the performance discussed by Peter Macgregor in his paper with an exception that the power method with k and log k dominant vectors performed equally well for us (against a better performance of $logk$ dominant vectors in the paper).



6.3 Spectral Clustering Performance vs Parameters

One of the important parameters we used for the clustering algorithm was the number of nearest neighbors while building the affinity matrix. We visualised the effect of Clustering Performance in terms of ARI against different values of nearest neighbors ranging from 10 to 500.



We see that the clustering performance (ARI) goes down as the nearest neighbors increase. On a side note, we used 10 nearest neighbors for our previous algorithm comparisons as we had done this analysis in advance to pick the best nearest neighbor parameter.

7 SUMMARY

The results from our implementation demonstrate that the fast spectral clustering method achieves significant performance improvement in runtime complexity compared to classical clustering, particularly as the dataset size increases which is in complete agreement with the paper that we referred to for this project.

The study confirms that fast spectral clustering is a viable alternative to traditional methods, offering significant computational efficiency without sacrificing accuracy. Future work could explore further enhancements and applications of this method in various domains of machine learning and data analysis.

References

[1] Peter Macgregor. *Fast and Simple Spectral Clustering in Theory and Practice.*

[2] Andrew Ng, Michael Jordan, and Yair Weiss. *On spectral clustering: Analysis and an algorithm.* 15th Advances in Neural Information Processing Systems.

[3] Prof. Carolyn L Beck. *IE 529 Stats of Data and Clustering Lectures.*