DA324 Course Project

Aryan Singh & Varun Nagpal
210150008 & 210150020

---

1 Introduction

In today's data-driven landscape, extracting valuable insights from large datasets is essential for informed decision-making across various fields. Clustering, a key task in unsupervised learning, plays a vital role in organizing data into meaningful groups based on similarity, aiding in analysis and interpretation.

In this report, we outline our approach to clustering a dataset of 11,952 samples into 10 distinct clusters. We utilize spectral and K-means clustering techniques to effectively partition the data. Additionally, we explore the use of neural networks for predicting cluster labels for unlabeled samples.

Our goal is to offer a comprehensive analysis of the dataset, covering preprocessing steps, model development, and evaluation. Through this project, we aim to showcase the effectiveness of clustering algorithms in revealing hidden patterns within complex datasets, while also identifying areas for improvement and future exploration.

The report is structured into several sections, starting with an overview of the dataset and proceeding to discuss data visualization methods. We then delve into the model description, detailing our clustering and prediction methodologies. Following sections address hyperparameter selection, evaluation metrics, and any challenges encountered during the project. Finally, we conclude with key insights gained and potential directions for future research.

2 Dataset

The dataset provided for this project comprises three main files: attributes.csv, adjacency.csv, and seed.csv. It consists of 11,952 samples, each represented by a set of features captured in the attributes file. These features encapsulate various characteristics or attributes of the samples, which are essential for clustering and analysis.

The attributes.csv file contains 11,952 rows and 103 columns, with each row corresponding to a single sample and each column representing a specific attribute or feature. These attributes may include numerical values representing measurements, categorical variables, or other types of data.

In addition to the attributes file, the dataset includes an adjacency matrix stored in the adjacency.csv file. This matrix provides information about the relationships or connections between different samples in the dataset. Each entry in the adjacency matrix indicates the presence or absence of a connection between two samples, thereby encoding the underlying structure or topology of the dataset.

Furthermore, the seed.csv file contains information about initial cluster seeds, which serve as the starting points for clustering algorithms. These seeds help guide the clustering process and ensure consistency in the formation of clusters across different runs of the algorithm.

Overall, the dataset presents a rich and diverse set of samples, each characterized by a unique combination of attributes. By leveraging this data, we aim to explore patterns, similarities, and relationships between samples, ultimately leading to the identification of coherent clusters within the dataset.

2.1 Data Visualization

Data visualization plays a crucial role in understanding the underlying structure and patterns within a dataset. In this project, we employed the t-distributed Stochastic Neighbor Embedding (t-SNE) technique to visualize the clusters formed by our clustering algorithm.

Using the t-SNE algorithm, we transformed the high-dimensional data into a two-dimensional space while preserving the local and global structure of the data as much as possible. This enabled us to visualize the inherent clusters and relationships between samples in a more interpretable manner.

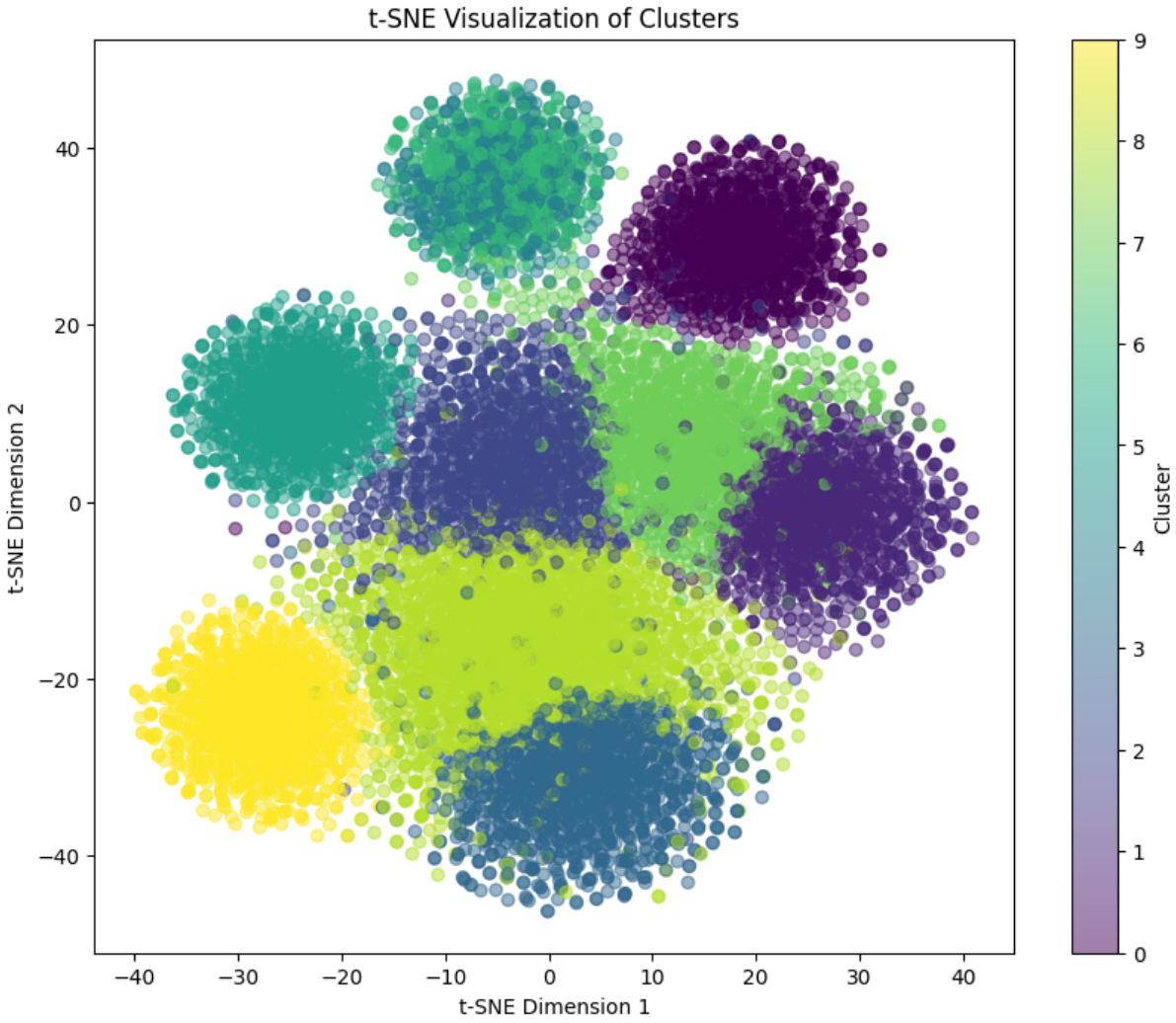

Figure 1: Visualization of clusters using t-SNE

3 Methodology

In this section, we outline the detailed pipeline of our framework, encompassing the model description, hyperparameter selection and tuning, as well as the evaluation metrics employed to assess the performance of our approach.

3.1 Model Description

Our model leverages a multi-step approach to cluster the dataset into distinct groups. Firstly, we apply Spectral Decomposition on the adjacency matrix, capturing the structural relationships

between data points in the form of eigenvalues and eigenvectors. Simultaneously, we employ Principal Component Analysis (PCA) on the attributes matrix to reduce its dimensionality while retaining the most informative features.

Subsequently, we combine the spectral decomposition results with the reduced-dimensional attributes using PCA, thereby integrating both structural and attribute-based information. Finally, we apply K-means clustering to the combined data to partition the dataset into clusters based on similarity.

3.2 Hyperparameter Selection and Tuning

The hyperparameters in our model include the number of principal components to retain in PCA and the mode parameter while constructing the K-nearest neighbors graph for spectral decomposition.

To determine the optimal number of principal components, we conducted experiments varying the number of components and evaluated their impact on clustering performance. Although the number of clusters for K-means clustering is fixed to 10 as per the task requirement, we explored different values for the number of principal components to identify the most informative representation of the attribute data.

Additionally, in the spectral decomposition step, we experimented with different modes for constructing the K-nearest neighbors graph, such as 'rbf' and 'nearest_neighbors'. This allowed us to explore different methods of capturing the local neighborhood structure of the data points and assess their influence on the final clustering results.

Through rigorous experimentation and evaluation, we determined the optimal hyperparameter settings that resulted in the most effective clustering performance, enabling us to generate meaningful clusters representative of the underlying data distribution.

3.3 Evaluation Metrics

In assessing the performance of our clustering model, we aimed to employ evaluation metrics that provide insights into the quality and consistency of the generated clusters. While we strived to implement various evaluation metrics, including silhouette score, adjusted Rand index, and Davies-Bouldin index, to quantitatively measure the clustering effectiveness, the actual implementation of these metrics may vary depending on the specific requirements and constraints of the task.

The silhouette score is a measure of how similar an object is to its own cluster compared to other clusters, ranging from -1 to 1, where a higher score indicates better-defined clusters. The adjusted Rand index quantifies the similarity between two clusterings, considering all pairs of samples and counting pairs that are assigned in the same or different clusters. The Davies-Bouldin index assesses the average 'similarity' between each cluster and its most similar cluster, with lower values indicating better clustering.

By exploring these evaluation metrics, we sought to gain deeper insights into the performance of our clustering model and validate its ability to accurately partition the dataset into meaningful clusters. However, the specific implementation and utilization of these metrics may be subject to the availability of ground truth labels and other considerations inherent to the clustering task.

4 Failed Attempts

Our exploration involved numerous attempts at leveraging various algorithms and techniques to improve clustering accuracy and prediction performance. However, many of these attempts did not yield the desired outcomes.

4.1 Clustering Algorithms

We experimented with several clustering algorithms, including KMeans, Agglomerative clustering, and Spectral Clustering. Despite our efforts, these algorithms did not consistently deliver significant improvements in clustering accuracy.

4.2 Dimensionality Reduction Techniques

In an effort to mitigate noise and enhance clustering, we explored dimensionality reduction techniques such as Principal Component Analysis (PCA), Random Projection, ISOMAP, and t-SNE. However, these techniques did not consistently lead to substantial improvements in clustering performance.

4.3 Spectral Clustering Hyperparameters

We encountered challenges in selecting and optimizing hyperparameters for spectral clustering, such as the mode for constructing the affinity matrix and tuning parameters like the number of nearest neighbors in the k-nearest neighbors graph. Despite our efforts to fine-tune these parameters, the resulting clustering outcomes did not demonstrate significant enhancements in accuracy or stability.

4.4 Feature Extraction

While experimenting with feature extraction methods, including node-level, link-level, and graph-level features, we found that traditional methods alone did not sufficiently capture the underlying patterns and relationships within the data.

4.5 Classifier Selection

We explored various machine learning classifiers, including Multilayer Perceptron (MLP) and Graph Convolutional Network (GCN), to predict the cluster labels of the remaining 1000 samples.

Despite our extensive exploration and experimentation, many of our attempts did not yield the desired improvements in clustering accuracy and prediction performance. These challenges underscore the complexity of the task and the importance of continued exploration and refinement in developing effective machine learning models.

5 Results

After extensive experimentation and refinement of our clustering and prediction framework, we achieved a final score of 0.12148 on the Kaggle leaderboard. While we aimed for higher performance, our results reflect the culmination of iterative improvements and strategic adjustments throughout the development process.

Our framework utilized a combination of Spectral Clustering applied to the adjacency matrix, along with dimension-reduced attributes obtained through Principal Component Analysis (PCA). We then employed KMeans clustering to partition the combined data into distinct clusters, followed by the utilization of a fine-tuned Multilayer Perceptron (MLP) as a classifier for the remaining 1000 samples.

Despite our efforts to optimize the model and fine-tune hyperparameters, achieving a higher score proved challenging. We speculate that the discrepancy between our expected performance and the obtained score may be attributed to various factors, including differences in the characteristics of the training and test data, as well as potential inconsistencies in the dataset used for clustering and prediction.

Furthermore, our analysis revealed limitations in the explanatory power of the adjacency data, particularly concerning the variance explained by the calculated eigenvectors. This observation suggests that the inherent complexity and structure of the dataset may not be fully captured by the adjacency matrix alone, contributing to challenges in achieving higher clustering accuracy.

In conclusion, while our framework yielded a respectable performance on the Kaggle leaderboard, the journey highlighted the inherent complexities and nuances involved in clustering and prediction tasks.

The code for our project can be found at: `https://drive.google.com/drive/folders/14ESxlCq4g2xofMHiy6hLVkf3mMJEIjVH?usp=sharing`

Figure 2: Kaggle leaderboard final position