MACHINE LEARNING

1. Which of the following is an application of clustering?

Answer-(d)

2. On which data type, we cannot perform cluster analysis?

Answer-(d)

3. Netflix's movie recommendation system uses?

Answer-(c)

4. The final output of Hierarchical clustering is ?

Answer-(b)

5. Which of the step is not required for K-means clustering?

Answer-(d)

6. Which is the following is wrong?

Answer-(c)

7. Which of the following metrics, do we have for finding dissimilarity between two clusters in hierarchical clustering?

i. Single-link

ii. Complete-link

iii. Average-link Options:

Answer-(d)

8. Which of the following are true?

i. Clustering analysis is negatively affected by multicollinearity of features

ii. Clustering analysis is negatively affected by heteroscedasticity

Answer-(a)

9. In the figure above, if you draw a horizontal line on y-axis for y=2. What will be the number of clusters formed?

Answer-(a)

10. For which of the following tasks might clustering be a suitable approach?

Answer-(b)

11. Given, six points with the following attributes: MACHINE LEARNING ASSIGNMENT – 3 Which of the following clustering representations and dendrogram depicts the use of MIN or Single link proximity function in hierarchical clustering:

Answer-(d)

12. Given, six points with the following attributes: Which of the following clustering representations and dendrogram depicts the use of MAX or Complete link proximity function in hierarchical clustering.

Answer-(c)

13. . What is the importance of clustering?

Answer- There are few steps importance of clustering:-

1. Having clustering methods helps in restarting the local search procedure and remove the inefficiency. In addition, clustering helps to determine the internal structure of the data.
2. This clustering analysis has been used for model analysis, vector region of attraction.
3. Clustering helps in understanding the natural grouping in a dataset. Their purpose is to make sense to partition the data into some group of logical groupings.
4. Clustering quality depends on the methods and the identification of hidden patterns.
5. They play a wide role in applications like marketing economic research and weblogs to identify similarity measures, Image processing, and spatial research.
6. They are used in outlier detections to detect credit card fraudulence.

14. How can I improve my clustering performance?

Answer-

Objective

To provide a parsimonious clustering pipeline that provides comparable performance to deep learning-based clustering methods, but without using deep learning algorithms, such as autoencoders.

Materials and methods

Clustering was performed on six benchmark datasets, consisting of five image datasets used in object, face, digit recognition tasks (COIL20, COIL100, CMU-PIE, USPS, and MNIST) and one text document dataset (REUTERS-10K) used in topic recognition. K-means, spectral clustering, Graph Regularized Non-negative Matrix Factorization, and K-means with principal components analysis algorithms were used for clustering. For each clustering algorithm, blind source separation (BSS) using Independent Component Analysis (ICA) was applied. Unsupervised feature learning (UFL) using

reconstruction cost ICA (RICA) and sparse filtering (SFT) was also performed for feature extraction prior to the cluster algorithms. Clustering performance was assessed using the normalized mutual information and unsupervised clustering accuracy metrics.

Results

Performing, ICA BSS after the initial matrix factorization step provided the maximum clustering performance in four out of six datasets (COIL100, CMU-PIE, MNIST, and REUTERS-10K). Applying UFL as an initial processing component helped to provide the maximum performance in three out of six datasets (USPS, COIL20, and COIL100). Compared to state-of-the-art non-deep learning clustering methods, ICA BSS and/or UFL with graph-based clustering algorithms outperformed all other methods. With respect to deep learning-based clustering algorithms, the new methodology presented here obtained the following rankings: COIL20, 2nd out of 5; COIL100, 2nd out of 5; CMU-PIE, 2nd out of 5; USPS, 3rd out of 9; MNIST, 8th out of 15; and REUTERS-10K, 4th out of 5.

Discussion

By using only ICA BSS and UFL using RICA and SFT, clustering accuracy that is better or on par with many deep learning-based clustering algorithms was achieved. For instance, by applying ICA BSS to spectral clustering on the MNIST dataset, we obtained an accuracy of 0.882. This is better than the well-known Deep Embedded Clustering algorithm that had obtained an accuracy of 0.818 using stacked denoising autoencoders in its model.

Conclusion

Using the new clustering pipeline presented here, effective clustering performance can be obtained without employing deep clustering algorithms and their accompanying hyper-parameter tuning procedure.

Introduction

Grouping observed data into cohesive clusters without any prior label information is an important task. Especially, in the era of big-data, in which very large and complex amounts of data from various platforms are collected, such as image content from Facebook or vital signs and genomic sequences measured from patients in hospitals [1]. Often, these data are not labeled and a significant undertaking is typically required (usually by individuals with domain knowledge). Even in simple tasks, such as labeling images or video data can require thousands of hours [2, 3]. Therefore, using the unsupervised learning technique of cluster analysis can aide in the process of providing labels to observed data [4].

Classical clustering algorithms

Classical clustering algorithms used for analysis are K-means [5], Gaussian Mixture Models [6], and hierarchical clustering [4], all of which are based on using a distance measure to assess the similarity of observations. The choice for distance measure is typically data dependent. For instance, in image data, the similarity between pixels can be represented by the Euclidean distance, where as in text documents cosine distance matrix is typically used [7]. Moreover, appropriate feature representation of the observations is even more critical in order to obtain correct clusters of the data [8], since improved features provide a better representative similarity matrix.

Deep learning-based clustering

Early approaches for learning the appropriate feature space in clustering algorithms implemented deep autoencoders (DAEs) [9]. Song et al. [10] used DAEs to directly learn the data representations and cluster centers. Huang et al. [11] employed a DAE with locality and sparsity preserving constraints, which is followed by a K-means to obtain the cluster memberships. A more recent and popular approach by Xie et al. [8] learned the feature space and cluster membership directly using a stacked denoising autoencoder [12]. Following Xie et al. [8], there have been many studies proposing deep clustering algorithms to learn the feature space and cluster membership simultaneously using some form of an autoencoder [13,14,15]. A departure from the autoencoder framework was demonstrated by Yang et al. [16], who used recurrent and convolutional neural networks with agglomerative (hierarchical) clustering.

Spectral clustering

Another class of clustering algorithms, called spectral clustering [17, 18], is based on embedding the graph structure of the data through eigendecomposition (also known as spectral decomposition) of the Laplacian matrix [19]. Spectral clustering usually performs better than K-means and the aforementioned classical algorithms due to its ability to cluster non-spherical data [4]. A key issue in spectral clustering is to solve the multiclass clustering problem. This is accomplished by representing the graph Laplacian in terms of *k* eigenvectors, *k* being the number classes [20]. Then, either K-means clustering [18], exhaustive search [17], or discretization [21] is applied to this lower dimensional representation of the Laplacian to determine the final cluster memberships. Recently, autoencoders have been applied on the Laplacian to obtain the spectral embedding provided by the eigenvectors [22]. Another approach has been to use a deep learning network that directly maps the input data into the lower dimensional eigenvector representation, which is then followed by a simple clustering algorithm [23].

Research aim

The drawback of deep learning methods is that they tend to have many hyper-parameters, such as learning rates, momentum, sparsity parameters, and number of features and layers [14, 24, 25]. All of which can make deep learning models difficult to train, since hyper-parameters can severely effect performance [24]. Typically, choosing the correct hyper-parameters requires expertise and ad hoc selection [14, 25]. However, the high degree of complexity in implementing deep learning-based algorithms [24] may be a limiting factor of their application in non-computer science based research fields. To have real-world applicability, clustering applications need to have as few hyper-parameters as possible [14].

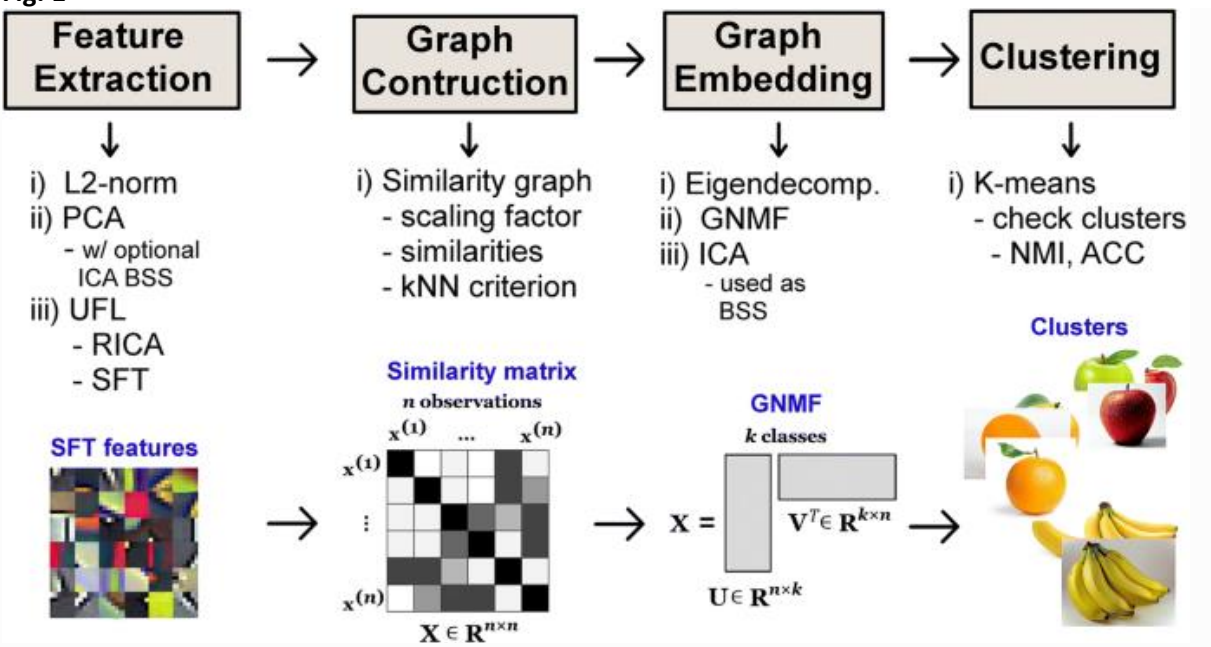
In this study, the aim is to provide a parsimonious and accessible clustering processing scheme that incorporates deep learning-style feature extraction, but without the complex hyper-parameter tuning procedure. The goal is to bridge the gap between deep learning-based clustering methods and widely available standard clustering techniques. This is accomplished by using two procedures. First, we improve the clustering accuracy of standard clustering algorithms by applying independent component analysis (ICA) [26] blind source separation (BSS) after the initial matrix factorization step in principal component analysis (PCA) and graph-based clustering algorithms. Second, we improve the features used for constructing the distance matrix in graph-based clustering techniques by performing feature extraction using deep learning-inspired feature learning techniques. Prior to any clustering algorithm, we implement the unsupervised feature learning (UFL) algorithms of ICA with reconstruction cost (RICA) [27] and sparse filtering (SFT) [28], both of which have only one tunable hyper-parameter—the number features [28].

By implementing these two procedures we demonstrate that effective clustering performance that is on par with more complex deep learning clustering models can be achieved. Thus, the clustering methodologies provided herein are designed to be simple to implement and train in different data applications.

Materials and methods

An overview of the clustering pipeline implementing unsupervised feature learning and ICA blind source separation is provided in Fig. 1. The clustering pipeline consists of four key components: (1) feature extraction, (2) graph construction, (3) graph embedding, and (4) K-means clustering. In the following, the datasets are first described and then the four components are introduced.

Fig. 1



Pipeline for processing. Each of the components contains the options available for implementation. The simplest processing pipeline to obtain clustering results consists of a L_2-normalization on the data, followed by K-means clustering. The processing stream with the most components would consist: (1) L_2-normalization followed by UFL using either RICA or SFT; (2) similarity graph construction; (3) GNMF or spectral decomposition followed by ICA blind source separation; and (4) K-means clustering