Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Evaluation of clustering methods for segmentation of hyperspectral remote sensing data

Ehsan Farahbakhsh
University of Sydney
Pulkit Sharma
Pingla Institute
Aman Agrawal
Indian Institute of Technology Kanpur
Rohitash Chandra
rohitash.chandra@unsw.edu.au

UNSW Sydney

Research Article

Keywords: Remote Sensing, Machine learning, clustering, hyperspectral data

Posted Date: July 21st, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10413878/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Evaluation of novel clustering methods for processing hyperspectral remote sensing

Ehsan Farahbakhsh,
Aman Agrawal, Pulkit Sharma, and Rohitash Chandra

***Abstract*—Hyperspectral remote sensing features more information than conventional remote sensing data by featuring a large number of bands of spectral data that capture ground-truth measurements against noise and uncertainties. A large amount of data is generally unlabelled in remote sensing applications; hence, unsupervised machine learning methods such as clustering have been prominent. However, there are several clustering methods with different strengths and weaknesses, and their suitability for processing hyperspectral data is unknown. In this paper, we provide an empirical study of prominent clustering methods, including variations of k-means clustering, balanced clustering, and agglomerative clustering. Our results reveal that xxxx**

***Index Terms*—Remote sensing; machine learning; clustering; computer vision; hyperspectral data**

I. Introduction

Remote sensing consists of using specialised instruments and technologies to measure and monitor the Earth's surface and atmosphere from a distance [1]. It allows scientists and resource managers to collect data for land cover, vegetation, water bodies, atmospheric conditions, and security and military operations [2], [3], [4]. One type of remote sensing data that has become increasingly popular in recent years is hyperspectral data [5], also known as imaging spectroscopy, which involves the acquisition of spectral signatures of objects or materials on the Earth's surface. These signatures are unique patterns of light reflectance or emission across a range of wavelengths in the electromagnetic spectrum and can provide detailed information about the composition and properties of different materials [6]. Hyperspectral data can be collected from a variety of platforms, including aircraft, satellites, and ground-based sensors [7]. It has a wide range of applications, including mineral exploration, crop monitoring, and environmental monitoring [8][9]. The high spatial and spectral resolution of hyperspectral data allows for identifying and mapping specific materials and features on the Earth's surface with greater accuracy and precision than other types of remote sensing data.

Machine learning involves the use of algorithms that can analyze large amounts of data and identify patterns and relationships within the data [10], [11], [12]. In the field of remote sensing, machine learning methods such as deep learning extract meaningful information to make predictions and classifications [13]. Machine Learning combined with hyperspectral remote sensing enables geoscientists to limit the negative impact of mineral prospecting activities on the ecology and environment with efficient and effective cost-cutting solutions. Machine learning methods can be used to classify different land cover types, such as forests, grasslands, or urban areas, based on the spectral signatures [14]. They can also be used to identify specific minerals [15], biomass and moisture retrievals in the environment [16], atmospheric environment studies [17], and predict crop yields [18]. The use of machine learning in the processing of hyperspectral data has the potential to improve the accuracy and efficiency of data analysis and to facilitate the development of new applications and insights.

Unsupervised machine learning is used for unlabelled training data, where the algorithms discover patterns and relationships within a dataset based on the inherent structure of the data [19], [20], [21]. Clustering is a prominent unsupervised learning method used in the processing of remote sensing [22] that includes multi-spectral and hyper-spectral data. Clustering methods partition a set of data points according to some measure of similarity (such as distance) to reveal subgroups within heterogeneous data such that each individual cluster has greater homogeneity than the whole [23]. The notion of a cluster cannot be precisely defined, which is one of the reasons why there are several clustering algorithms [24]. As clustering algorithms involve several parameters, mostly operated in high dimensional spaces, and have to cope with noisy, incomplete, and sampled data, their performance can vary substantially for different applications and types of data. Clustering algorithms are suited for application on unlabelled remote sensing data, and several applications show the effectiveness of clustering for multispectral [25] and hyperspectral data [26].

In the context of hyperspectral data, clustering algorithms can be used to identify different types of materials or land cover classes based on their spectral signatures [27]. Clustering has been used to group data points with a similar spectral signature for representing a particular type of vegetation [28]. Clustering algorithms can also be used to identify anomalous or unusual data points within the dataset, which may represent features of interest or potential errors in the data [29]. This can be useful in situations where the characteristics or identities of the materials or features present in the data are not known prior.

E. Farahbakhsh is with the EarthByte Group, School of Geosciences, The University of Sydney, Sydney, Australia (e-mail: e.farahbakhsh@sydney.edu.au).

P. Sharma is with Pingla Institute, Sydney, Australia (e-mail: pulkit.22sharma.22@gmail.com)

A. Agrawal is with the Indian Institute of Technology Kanpur, Kanpur, India (e-mail: amanag2019@gmail.com)

R. Chandra is with the Transitional Artificial Intelligence Research Group, School of Mathematics and Statistics, University of New South Wales, Sydney, Australia. (e-mail: rohitash.chandra@unsw.edu.au).

In comparison to multi-spectral remote sensing data, hyperspectral data [30] poses further challenges for clustering methods, since they have more than 100 bands of data. In large datasets (regions), it is difficult to ascertain which types of clustering methods are more suitable. Furthermore, often in hyperspectral data, dimension reduction methods such as principal component analysis (PCA) [31], and deep autoencoders [32], are used to compress the data bands so that it is easier for machine learning methods to process the data. Dimensionality reduction with PCA can reduce a large number of spectral banks into a small number that can be easily processed by clustering algorithms. Hence, it is important to evaluate the prominent clustering methods for hyperspectral data to make them more accessible to the remote sensing and machine learning community. Clustering algorithms can provide insight into the underlying structure of the data and facilitate the identification of patterns and relationships that may not be immediately apparent. However, it is important to note that the results of clustering algorithms are highly dependent on the parameters and assumptions used by the algorithm, and may require additional interpretation or validation by a human analyst.

The paramount objective of clustering in remote sensing data is to group similar pixels or spectra and has been prominent as an unsupervised machine learning method. Several prominent clustering methods exist, and therefore, it is important to evaluate their strengths and weaknesses. In this study, we evaluate prominent selected clustering methods to process hyperspectral data and map target features. Moreover, we propose a framework for applying five clustering methods in image segmentation. We present a framework that first processes the hyperspectral data using dimensionality reduction and then applies the clustering algorithms. We compare the performance of each method based on ground-truth data and clustered features using different metrics from the literature.

II. Materials and methodos

A. Datasets

1) Indian Pines dataset: The Indian Pines dataset [33] is a hyperspectral image segmentation dataset for a selected landscape in Indiana, United States, using 145 $\times$ 145 pixels. The dataset is captured by the Airborne Visible InfraRed Imaging Spectrometer (AVIRIS) satellite [34]. In each pixel, the data set contains 220 spectral reflectance bands which represent different portions of the electromagnetic spectrum in the wavelength range $0.4-2.5\mu$m (micro-meters) . The dataset features two-thirds agriculture and one-third of forest and other natural perennial vegetation. The ground truth available is designated into seventeen clusters and is not all mutually exclusive. The dataset has been prominently used in remote sensing and machine learning literature.

2) Pavia University dataset: This dataset was captured by Reflective Optics System Imaging Spectrometer (ROSIS) sensor on 610×610 pixels during a flight campaign over Pavia, northern Italy. In each pixel, the dataset contains 103 spectral bands. The geometric resolution is 1.3 meters. The image is divided into 10 classes with a total of 42,776 labelled samples, including asphalt, meadows, gravel, trees, metal sheet, bare soil, bitumen, brick, and shadow.

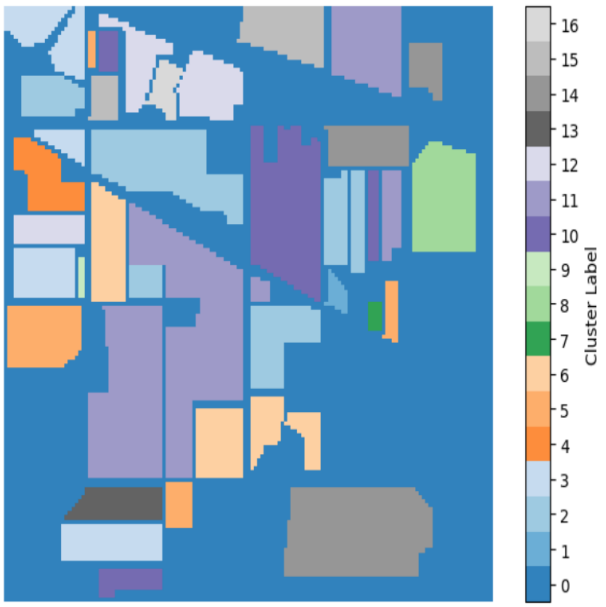


Fig. 1: Indian Pines ground-truth dataset with 17 clusters.

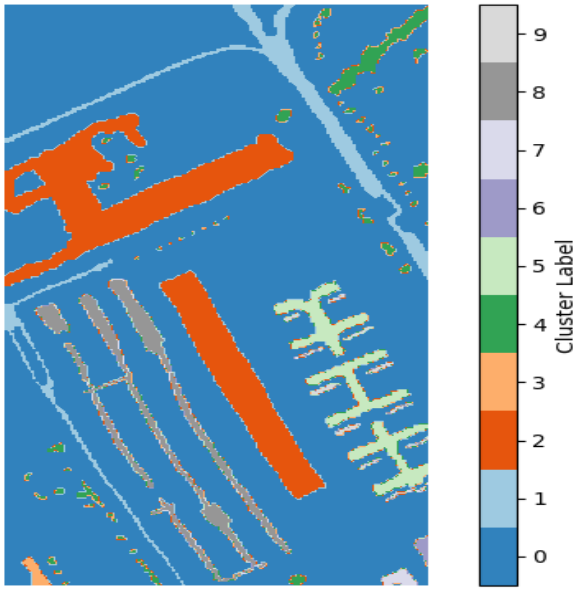


Fig. 2: Pavia University ground truth dataset with 10 clusters.

B. Dimensionality Reduction

1) t-SNE: t-SNE (t-Distributed Stochastic Neighbour Embedding) [35], [36] is a popular dimensionality reduction technique commonly used for visualising high-dimensional data in a lower-dimensional space. It initially computes pairwise similarities between data points using a Gaussian kernel, which captures the relationship between points in the original high-dimensional space. It then maps data points to generate a similar probability distribution in a lower-dimensional space. The approach iteratively employs gradient-based optimisation

to update the position of the data points in the low-dimensional embedding by reducing the divergence (Kullback-Leibler divergence) between the two distributions. t-SNE has been applied to a wide range of problems that require dimensionality reduction and data visualisation, including geology [37], [38] and remote sensing [39]. There have been extensions of t-SNE with automatic parameter optimisation for large datasets [40].

2) Incremental PCA: Incremental PCA has been designed to handle datasets that are too large to fit in memory. Incremental PCA performs PCA on large datasets by processing the data in smaller, manageable batches rather than the entire dataset at once. The idea is to update the eigenvectors and eigenvalues incrementally as new data becomes available, and hence reduces the memory usage and the computational time.

C. Clustering methods

Clustering algorithms aim to find hidden underlying structures in order to group the data and are used for unlabelled data and part of unsupervised machine learning [19], [21]. Broadly, the clustering algorithm makes clusters using a distance measure where similar data points are grouped together and hence can be used in the processing of unlabelled hyperspectral data [41].

1) K-means clustering: As a standard clustering algorithm, k-means clustering has been used to process hyperspectral data for recognising objects [42]. This algorithm distributes representative points of k classes in an n-dimensional space, where each spectrum in each pixel (data point) is assigned to the class whose representative point is the closest [43] and the centroids are calculated [44]. The k-means clustering continues this procedure until the pixel assignment becomes invariable from the previous assignment. The alignment of data points and cluster centres with the standard k-means algorithm can become very complicated due to mixed pixels. The objective (error) function of k-means clustering is defined as the sum of distances (typically Euclidean distance) between data points and their respective cluster centroids as given below:

$$\mathcal{L} = \sum_{k=1}^{K} \sum_{i=1}^{n} 1\{c_i = k\} ||x_i - \mu_k||^2$$

where x_i is ith training data point c_i cluster assignment to the data point x_i and μ_k is the centroid of the cluster k. $1\{c_i = k\}$ is a map which output 1 when $c_i = k$ else 0. The goal of k-means clustering is to find the optimal values for the centroids that minimise the objective function $\mathcal{L}$. The optimal solution $\boldsymbol{\mu}^*$ and $\mathbf{c}^*$ (where $\boldsymbol{\mu} = (\mu_1, \mu_2, ..., \mu_K)$ and $\mathbf{c}^* = (c_1, c_2, ..., c_n)$ are vectors) can be seen as follows.

$$\boldsymbol{\mu}^*, \mathbf{c}^* = argmin_{\mu,c} \mathcal{L}$$

We note that k-means clustering does not guarantee finding the globally optimal solution due to its sensitivity to initial centroid selection and potential convergence to local optima. Therefore, multiple runs with different initialisations are often performed to improve the chances of finding a good clustering solution.

2) Mini-batch k-means clustering: Mini-batch k-means is a variation of the traditional k-means clustering algorithm designed to handle large datasets [45]. In traditional k-means, the algorithm processes the entire dataset in each iteration, which can be computationally expensive for large datasets. Mini-batch k-means addresses this issue by processing only a small subset of the data, called a mini-batch, in each iteration. The mini-batch is randomly sampled from the dataset, and the algorithm updates the cluster centroids based on the data in the mini-batch. This allows the algorithm to converge faster and use less memory than traditional k-means clustering. Each mini-batch updates the clusters using a combination of data instances, applying a learning rate that decreases with the number of iterations. As the number of iterations increases, the effect of new examples is reduced, and convergence is determined when no changes in the clusters occur in several consecutive iterations [46].

3) Bisecting k-means clustering: Bisecting k-means [47] is a hierarchical clustering algorithm that begins with a single cluster containing all data points and iteratively splits clusters based on the k-means algorithm. The *bisection* procedure selects the best cluster using a cost function such as the sum of distances between data points assigned to that particular cluster and the centroid. The *k-means split* divides the selected cluster into two clusters, which involves finding two new centroids that minimise the objective function within the selected cluster. Banerjee et al. [48] provided an empirical evaluation of bisecting k-means and found it to have similar performance when compared to k-means, and also reported that fuzzy and genetic k-means were better in performance.

4) Hierarchical agglomerative clustering: The hierarchical agglomerative clustering (HAC) algorithm [49], [50] is a bottom-up clustering approach, where each data point is grouped into a cluster of its own and pairs of close clusters are merged until a single cluster is left. It begins by calculating the pairwise distances or similarities between all data points and then merges the two closest data points (or clusters) into a single cluster, updating the distance/similarity matrix accordingly. This process continues iteratively until a stopping criterion is satisfied. There are different linkage criteria to determine the distance/similarity between clusters at each iteration, such as single linkage (minimum distance), complete linkage (maximum distance), or average linkage (average distance). The choice of linkage criterion impacts the shape and structure of the resulting clusters [51], [52]. HAC creates a hierarchy of clusters, represented by a dendrogram [53], which illustrates the merging process and allows for exploring different levels of granularity. The dendrogram can be cut at a specific height to obtain the desired number of clusters. Goncalves et al. [54] used hierarchical agglomerative clustering with self-organising maps for classification in remote sensing application, Wilson et al. [55] provided a comparison with partitional clustering for classification of multi-spectral remote sensing data. Furthermore, Senthilnat et al. [56] used it for landcover mapping using satellite images.

5) BIRCH: BIRCH (balanced iterative reducing and clustering using hierarchies) [57] aims to efficiently cluster data points, while maintaining a compact summary of the

data. BIRCH can also be used to accelerate k-means clustering and Gaussian mixture modelling with the expectation–maximisation algorithm [58]. The advantage of BIRCH is its ability to incrementally and dynamically cluster incoming, multidimensional data points in an attempt to produce the best quality clustering for a given set of resources (memory and time constraints). In most cases, BIRCH only requires a single scan of the database, where the decision is made without scanning all data points and currently existing clusters. It exploits the observation that the data space is not usually uniformly occupied, and not every data point is equally important. It makes full use of available memory to derive the finest possible sub-clusters while minimising input/output costs. In the case of remote sensing data processing, BIRCH has been used for land cover categorisation [59].

6) Gaussian mixture models: Gaussian mixture models (GMMs) [60] are statistical techniques which assume that the data points in each cluster follow a Gaussian (normal) distribution and that the overall dataset is a mixture of these Gaussian distributions. In a GMM, the goal is to estimate the parameters of the Gaussian distributions, such as their mean and covariance, as well as the mixing proportions that determine the relative importance of each distribution in the mixture. This estimation process is typically performed using the expectation-maximisation (EM) algorithm. The EM algorithm alternates between calculating the expectation of the log-likelihood (expectation step, E-step) and maximizing the expected log-likelihood obtained by the E-step (maximization step, M-step). However, the EM algorithm typically converges to a local optimum and is highly dependent on the initialisation. The common solutions consist of one or a combination of the following strategies: 1) using multiple random starts and selecting the estimates with the highest likelihood; 2) the rough set model; and 3) initialisation with k-means or agglomerative clustering. GMM classifiers have also gained significant attention in the remote sensing community due to the capability of capturing non-Gaussian statistics of multivariate data and remarkable accuracy in classifying high-dimensional data [61]. GMMs do not consider spatial correlations and classify image data only based on their spectral information. Therefore, they do not employ the relevant information that can be extracted from the spatial domain. GMMs have been prominent in remote sensing applications [62], [63], [64]

D. Framework

Figure 3 presents the proposed framework that we developed to systematically compare clustering algorithms for hyperspectral remote sensing data. The framework consists of seven sequential stages that integrate dimensionality reduction, hyperparameter optimisation, clustering, and comprehensive performance evaluation.

In **Step 1**, we use two benchmark hyperspectral datasets, namely **Indian Pines** and **Pavia University**. We separate each dataset into hyperspectral imagery and the corresponding ground-truth labels, which we use only for external performance evaluation. To reduce computational complexity while preserving scene diversity, we use the top-left quarter of the Pavia University dataset.

In **Step 2**, we reduce the dimensionality of the hyperspectral data to alleviate the curse of dimensionality. For the Indian Pines dataset, which contains 220 spectral bands, we apply t-distributed Stochastic Neighbour Embedding (t-SNE) [35] to obtain a compact low-dimensional representation. For the Pavia University dataset, which contains 103 spectral bands, we apply Incremental Principal Component Analysis (Incremental PCA) to reduce the data to six principal components while preserving the dominant spectral information.

In **Step 3**, we employ Bayesian hyperparameter optimisation to determine the optimal parameter settings for each clustering algorithm. This enables a fair and consistent comparison by minimising the influence of manually selected hyperparameters.

In **Step 4**, we cluster the reduced feature representations using six representative clustering algorithms: K-Means, Mini-Batch K-Means, Bisecting K-Means, Hierarchical Agglomerative Clustering (HAC), BIRCH, and Gaussian Mixture Models (GMM).

In **Step 5**, we qualitatively assess the clustering results by analysing the generated segmentation maps together with cluster cardinality plots, deviation plots, and box plots. These visualisations allow us to examine cluster balance, compactness, and spatial consistency.

In **Step 6**, we quantitatively evaluate the clustering performance using both intrinsic and extrinsic metrics, including the Silhouette coefficient, Calinski–Harabasz index, Davies–Bouldin index, Homogeneity, Completeness, V-measure, Rand Index, Adjusted Rand Index, Mutual Information, and Adjusted Mutual Information.

Finally, in **Step 7**, we compare the clustering algorithms by jointly considering the quantitative metrics and qualitative visual analysis. Based on these results, we identify the clustering algorithm that provides the best balance between clustering quality, robustness, and computational efficiency for hyperspectral image segmentation.

E. Metrics

In order to check the performance of each clustering method used in our study, we compared it with ground truth data and several performance metrics. There are mainly two types of performance evaluation metrics for clustering, namely, Extrinsic Measures and Intrinsic Measures, where the former require ground truth labels while the latter do not. Extrinsic measures include the Rand Index, Mutual Information, V-Measure and Fowlkes-Mallows Scores. Whereas Intrinsic Measures include Silhouette Coefficient, Calinski-Harabasz Index and Davies-Bouldin Index. After careful evaluation, we found out that by keeping the number of clusters limited to 17, we can get the best of the scores while doing performance metrics evaluation for each clustering algorithm.

The *Rand Index* [65] and *Adjusted Rand Index* [66] measure the similarity between the cluster assignments by making pairwise comparisons. A higher score signifies higher similarity. The Rand Index (RI) and Adjusted Rand Index (ARI), range from zero to one, with zero equating to random labelling and one when the clusters are identical. The ARI is a correction

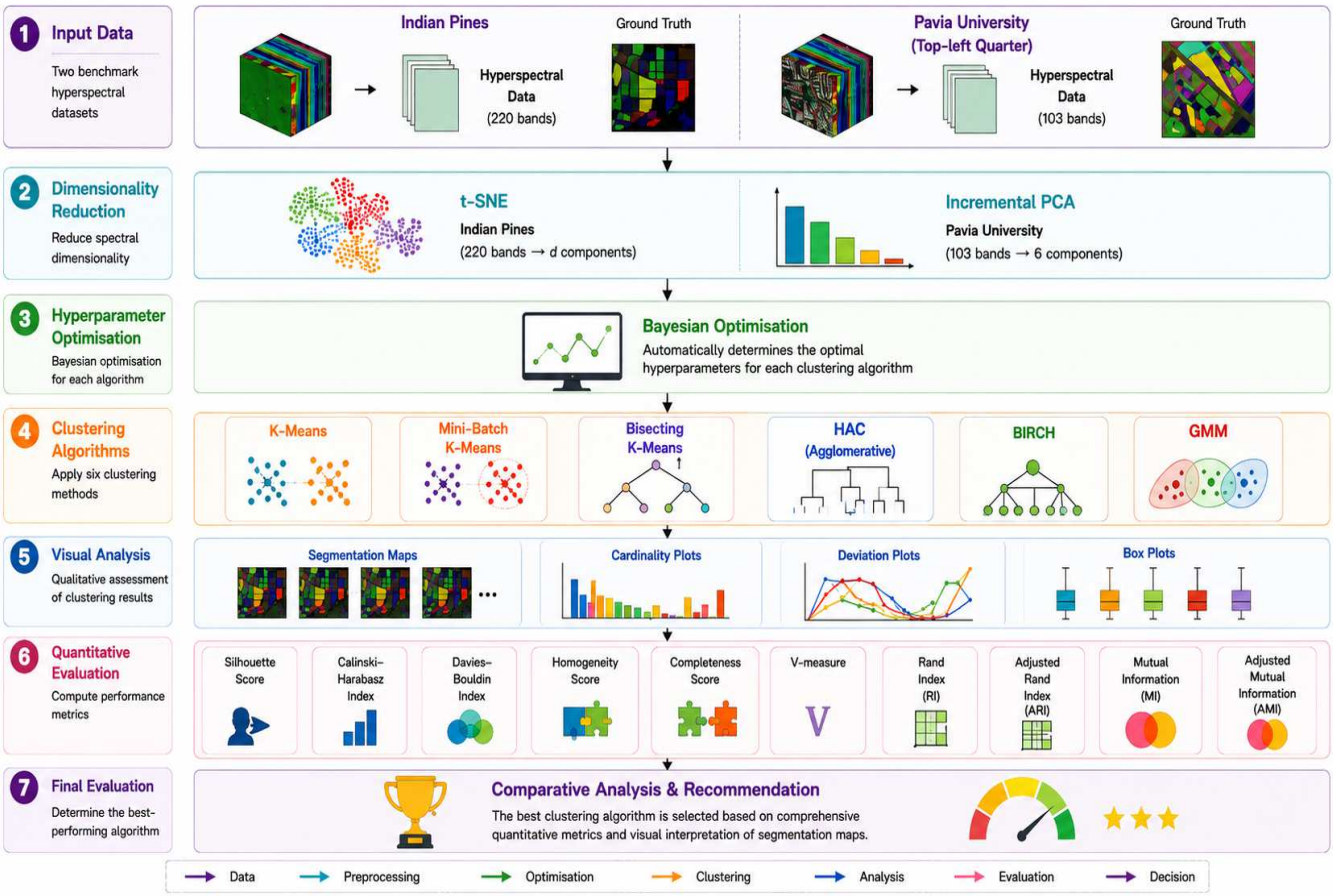


Fig. 3: Framework showing the combination of dimensionality reduction with clustering methods for image segmentation.

of the Rand Index, but it has the disadvantage of being sensitive to chance. The ARI takes into account the fact that some agreement between two clusterings can occur by chance, and it adjusts the Rand Index to account for this possibility. Furthermore, both RI and ARI are symmetric: swapping the argument does not change the scores. They can thus be used as consensus measures. For each pair, it is considered correct if the pair is predicted to be in the same cluster when they are in the same cluster (somewhat like "true positive") and correct if the pair is predicted to be in different clusters when they are indeed in different clusters (somewhat like "true negative"). However, Rand Index does not consider chance; if the cluster assignment was random, there can be many cases of "true negative" by fluke. Ideally, we want random (uniform) label assignments to have scores close to 0, and this requires adjusting for chance. ARI adjusts for chance by discounting a chance normalization term. If we carefully analyze the RI and ARI scores of each clustering algorithm in TABLE I, we can observe that our above statements truly justify the scores. That being said, K-Means and Bisecting K-Means are quite close in RI and ARI scores, respectively and have performed the best in comparison to other clustering algorithms, but barely by a small margin.

Mutual Information (abbreviated as MI) is another very popular extrinsic measure for comparing clusterings. It is beneficial to make an adjustment for chance to this measure, by subtracting an expected value and normalizing via an upper bound. This yields the constant baseline property that enhances intuitiveness [67]. The degree of agreement between clusters is computed by joint and marginal probabilities. There are two variations to Mutual Information: Normalized Mutual Information (NMI) and Adjusted Mutual Information (AMI). Normalised MI is MI divided by average cluster entropies and is often used in literature, while Adjusted MI is Normalized MI adjusted for chance by discounting a chance normalisation term. Mutual Information (MI, NMI, AMI) measures the agreement between the cluster assignments. A higher score signifies higher similarity. In our comparative study, we have just used Mutual Information Score and Adjusted Mutual Information Score because the usage of NMI measures has been argued because of the so-called selection bias problem, that is, they show the tendency to choose clustering solutions with more communities (Amelio and Pizzuti et. al., 2016). Now, by looking at performances of each clustering algorithms in TABLE II for MI score and AMI score, K-means and Gaussian Mixture have outperformed other algorithms, closely followed by BIRCH and Agglomerative clustering. But Mini-batch K-Means and Bisecting K-Means were not quite impressive in their MI and AMI scores.

Furthermore, we used *V-Measure* , Homogeneity and Completeness score. The calculation of the V-Measure first requires the calculation of two terms: (i) *Homogeneity* A perfectly

TABLE I: Performance metrics given by the clustering algorithms on Indian Pines dataset

Algorithm	Average Silhouette score	Calinski-Harabasz score	Davies-Bouldin score	Homogeneity score	Completeness score
K-Means	0.435094	27046.167058	0.724229	0.383581	0.249784
Mini-Batch	0.410561	25391.274908	0.784157	0.371982	0.241557
Bisecting K-means	0.369553	22572.193673	0.887189	0.365907	0.239102
BIRCH	0.403084	24609.124262	0.775742	0.379341	0.248238
Agglomerative	0.415769	24024.294808	0.718177	0.379266	0.253313
Gaussian Mixture	0.421448	27137.837030	0.749031	0.385646	0.250211

TABLE II: Performance metrics given by the clustering algorithms on Indian Pines dataset

Algorithm	V-measure	Rand index	Adjusted Rand index	Mutual Information score	Adjusted Mutual Information score
K-Means	0.302550	0.702769	0.064401	0.700711	0.300647
Mini-Batch	0.292907	0.699804	0.053890	0.679522	0.290980
Bisecting K-means	0.289216	0.705796	0.075604	0.668425	0.287272
BIRCH	0.300096	0.699926	0.057515	0.692965	0.298180
Agglomerative	0.303750	0.697218	0.056105	0.692829	0.301825
Gaussian Mixture	0.303505	0.701552	0.059032	0.704483	0.301608

homogeneous clustering is one where each cluster has datapoints belonging to the same class.(ii) *Completeness* A clustering result satisfies completeness if all the data points that are members of a given class are elements of the same cluster. The homogeneity and completeness of a clustering solution run roughly in opposition: Increasing the homogeneity of a clustering solution often results in decreasing its completeness [68]. V-measure is an entropy-based measure that explicitly measures how successfully the criteria of homogeneity and completeness have been satisfied. The 'V' stands for "validity", a common term used to describe the goodness of a clustering solution. V-measure is computed as the harmonic mean of distinct homogeneity and completeness scores, just as precision and recall are commonly combined into F-measure [69]. As F-measure scores can be weighted, V-measure can be weighted to favour the contributions of homogeneity or completeness. V-measure doesn't adjust for chance. This means that random labelling would not yield zero scores esp. if the number of clusters is large. If we closely look at the Homogeneity score and Completeness score in TABLE I, bisecting k-means didn't perform quite well in both of these scores, thereby affecting its v-measure score as well. Whereas every other clustering algorithm has quite comparable performance in these three performance metrics.

Calinski-Harabasz Index , or Variance Ratio Criterion, measures the sum of between-cluster dispersion against the sum of within-cluster dispersion, where dispersion is the sum of distance squared. measures the between-cluster dispersion against the within-cluster dispersion. A higher score signifies better-defined clusters. It is fast to compute but tends to be higher for density-based clustering algorithms, and would be unfair to compare against other types of clustering algorithms. K-Means and Gaussian Mixture have outperformed other clustering algorithms by a big margin due to the same fact mentioned above, whereas Bisecting K-Means

III. Results

Tables I–IV summarise the performance of the evaluated clustering algorithms on the Indian Pines and Pavia University datasets using both intrinsic and extrinsic clustering metrics. Overall, the results indicate that no single algorithm dominates every evaluation metric; however, clear trends emerge across both benchmark datasets.

For the Indian Pines dataset, K-Means consistently achieved among the highest scores in terms of the Silhouette coefficient (0.435), Calinski–Harabasz index ($27\,046$), Homogeneity (0.384), Mutual Information (0.701), and Adjusted Mutual Information (0.301). Gaussian Mixture Models (GMM) produced very similar performance, achieving the highest Calinski–Harabasz score ($27\,138$) while maintaining comparable external validation metrics. These results indicate that centroid-based and probabilistic clustering methods are well suited to the compact feature representation produced after dimensionality reduction.

Mini-Batch K-Means closely approximated the performance of standard K-Means across nearly all evaluation metrics while offering the advantage of reduced computational complexity. The relatively small reduction in clustering quality suggests that Mini-Batch K-Means provides an attractive alternative for large hyperspectral datasets where scalability is important.

Bisecting K-Means generally produced lower Silhouette and Calinski–Harabasz scores than standard K-Means, indicating that recursive binary partitioning was less effective in preserving the underlying spectral structure of the data. Although its Adjusted Rand Index was marginally higher than other methods, the remaining intrinsic and external metrics suggest that this improvement does not translate into consistently superior clustering quality.

BIRCH and Agglomerative clustering produced competitive but less consistent results. BIRCH generated relatively balanced mutual information scores but exhibited lower compactness according to intrinsic measures, while Agglomerative clustering achieved the lowest Davies–Bouldin index, suggesting well-separated clusters despite only moderate agreement with the ground-truth labels.

The results on the Pavia University dataset exhibit similar behaviour. K-Means again produced the highest Silhouette coefficient and Calinski–Harabasz score, indicating compact and well-separated clusters. Mini-Batch K-Means closely followed these results, whereas Gaussian Mixture Models obtained the highest Homogeneity and Mutual Information scores but considerably lower intrinsic clustering quality, as reflected by

TABLE III: Performance metrics given by the clustering algorithms on Pavia University dataset

Algorithm	Average Silhouette score	Calinski-Harabasz score	Davies-Bouldin score	Homogeneity score	Completeness score
K-Means	0.357191	41607.818818	0.806245	0.365578	0.153171
Mini-Batch	0.347300	39901.574953	0.844886	0.362743	0.151084
Bisecting K-means	0.287052	32913.858430	0.958083	0.325023	0.135972
BIRCH	0.341145	12364.167055	0.778865	0.200861	0.172901
Gaussian Mixture	0.180774	17678.538504	1.225791	0.375177	0.154916

TABLE IV: Performance metrics given by the clustering algorithms on Pavia University dataset

Algorithm	V-measure	Rand index	Adjusted Rand index	Mutual Information score	Adjusted Mutual Information score
K-Means	0.215889	0.425047	0.015315	0.322421	0.215469
Mini-Batch	0.213320	0.425269	0.016495	0.319920	0.212900
Bisecting K-means	0.191733	0.425035	0.014689	0.286653	0.191300
BIRCH	0.185835	0.514958	0.044201	0.177149	0.185355
Gaussian Mixture	0.219285	0.433931	0.031551	0.330886	0.218872

the Silhouette and Davies–Bouldin indices. This suggests that although GMM can better capture class distributions in certain cases, it is more sensitive to overlapping spectral distributions and parameter estimation.

Figures 4 and 5 provide qualitative comparisons of the resulting segmentation maps. K-Means and Gaussian Mixture Models generally produce more spatially coherent regions that closely resemble the reference maps, whereas Bisecting K-Means and BIRCH generate more fragmented segment boundaries. The cardinality plots further demonstrate that K-Means and Mini-Batch K-Means produce relatively balanced cluster sizes, while BIRCH tends to generate uneven cluster distributions with several dominant clusters. Similarly, the deviation and box plots indicate that K-Means yields tighter within-cluster distributions, reflecting greater cluster compactness and reduced variance.

Taken together, the quantitative metrics and qualitative visualisations consistently demonstrate that K-Means provides the best balance between clustering quality, robustness, and computational simplicity across both benchmark datasets, while Mini-Batch K-Means achieves comparable performance with improved scalability.

IV. Discussion

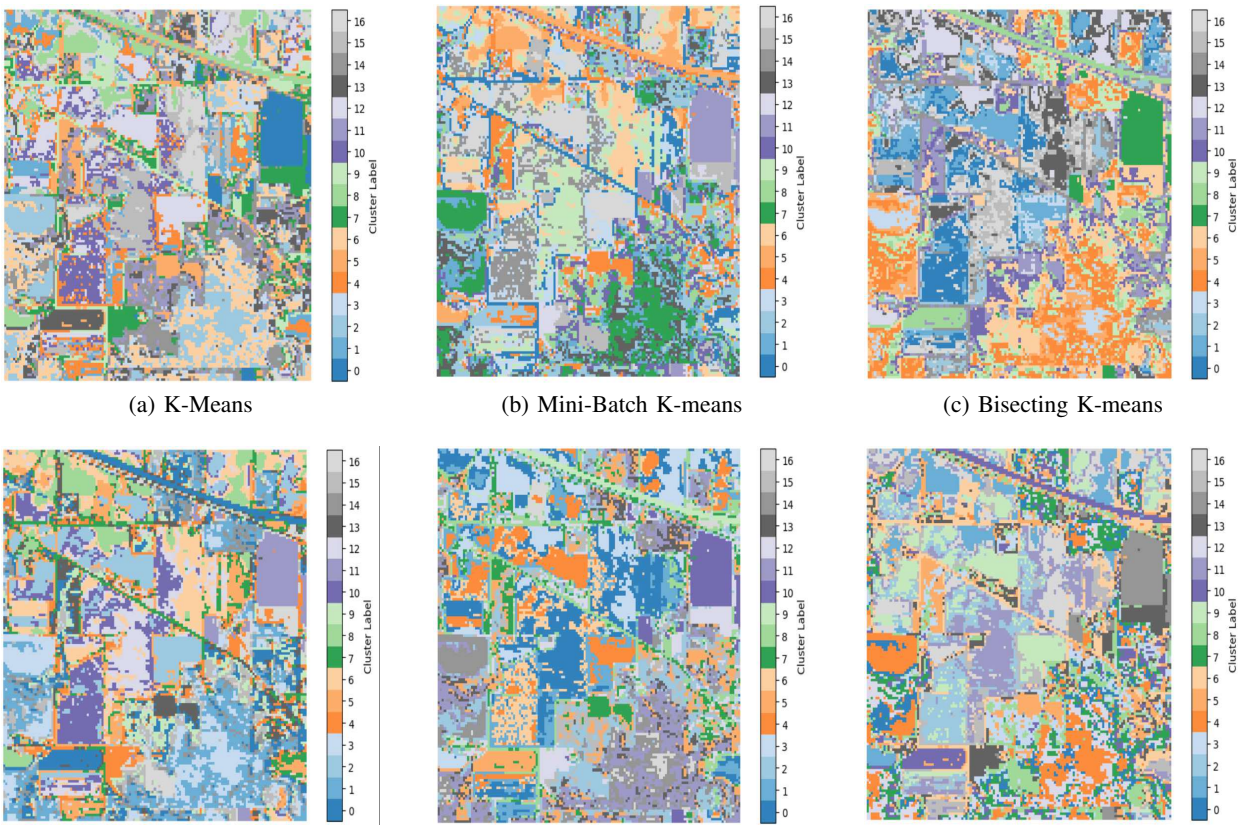
This study provides one of the few systematic empirical comparisons of widely used clustering algorithms under a common preprocessing and evaluation framework for hyperspectral remote sensing. While numerous studies have proposed novel clustering algorithms or application-specific methods for hyperspectral image analysis [41], [62], relatively few have examined how classical clustering methods compare when subjected to identical dimensionality reduction, hyperparameter optimisation, and evaluation protocols. Our framework therefore provides practical guidance for selecting clustering algorithms for hyperspectral image segmentation rather than introducing another specialised clustering technique.

Across both benchmark datasets, K-Means consistently achieved the best overall balance between intrinsic clustering quality and agreement with ground truth labels. In particular, it produced among the highest Silhouette coefficients, Calinski–Harabasz indices, Mutual Information scores and V-measure values while also generating visually coherent segmentation maps. These findings agree with previous studies that have shown that centroid-based clustering remains a strong baseline for hyperspectral image analysis because of its computational efficiency, robustness and ability to partition compact spectral manifolds after suitable preprocessing [42], [30].

The strong performance of K-Means can largely be attributed to the dimensionality reduction stage. Hyperspectral imagery is characterised by hundreds of highly correlated spectral bands, where Euclidean distance becomes less informative because of the curse of dimensionality [30]. By projecting the spectral signatures into a lower-dimensional manifold using t-SNE and Incremental PCA, the underlying class structure becomes more compact and better separated. Under these conditions, centroid-based optimisation naturally becomes more effective, allowing K-Means to identify representative spectral clusters with relatively simple distance calculations. This observation suggests that feature representation may play a more important role than clustering algorithm complexity for many hyperspectral applications.

Gaussian Mixture Models achieved performance comparable to K-Means on several external validation metrics, particularly Mutual Information and Homogeneity. Unlike K-Means, GMM explicitly models the covariance structure of each cluster and therefore provides greater flexibility for representing non-spherical data distributions [60]. However, the lower Silhouette coefficients and higher Davies–Bouldin indices observed on the Pavia University dataset indicate reduced cluster compactness. This behaviour is expected because expectation–maximisation optimisation is sensitive to parameter initialisation and can converge to local optima, particularly in high-dimensional data with overlapping spectral classes.

Mini-Batch K-Means produced results remarkably close to



(a) K-Means (b) Mini-Batch K-means (c) Bisecting K-means

(d) BIRCH (e) Agglomerative Clustering (f) Gaussian Mixture

Fig. 4: Predictions of various clustering algorithms on the Indian Pines dataset

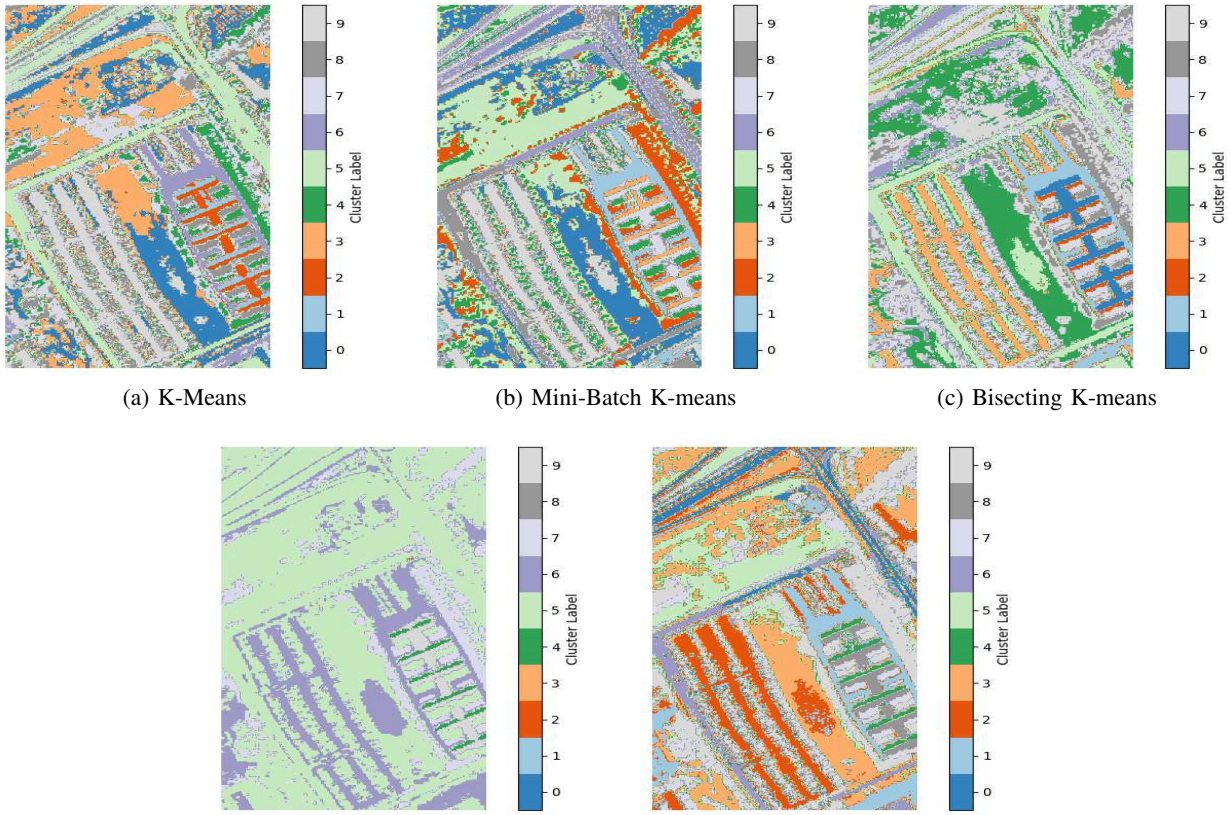
standard K-Means despite using stochastic optimisation over small subsets of the data. Previous work has demonstrated that Mini-Batch K-Means significantly reduces computational cost while preserving clustering quality for large datasets [45]. Our results confirm these observations for hyperspectral imagery and suggest that Mini-Batch K-Means represents an attractive choice for operational remote sensing applications involving increasingly large airborne and satellite datasets.

Hierarchical approaches, including Bisecting K-Means and Agglomerative Clustering, demonstrated more variable performance across the benchmark datasets. Although hierarchical methods have advantages in revealing multiscale data structures [49], [50], recursive partitioning appears less capable of representing the complex spectral variability introduced by mixed pixels and overlapping land-cover classes. Similarly, BIRCH achieved reasonable scalability but lower clustering consistency, reflecting the limitations of compressed cluster-feature summaries when representing highly heterogeneous hyperspectral observations [57].

An important contribution of this work is demonstrating that increasing algorithmic complexity does not necessarily translate into improved segmentation performance. Across nearly all evaluation metrics, relatively simple partition-based clustering methods performed as well as, or better than, more sophisticated probabilistic and hierarchical approaches. This finding has practical implications for remote sensing practitioners, suggesting that computational resources may be more effectively invested in feature extraction, dimensionality reduction and data preprocessing rather than increasingly complex clustering algorithms. The results therefore reinforce recent reviews arguing that representation learning has become the primary factor governing hyperspectral image analysis performance rather than the choice of clustering algorithm alone [41], [30].

Another contribution of this study is the comprehensive evaluation methodology. Instead of relying on a single performance metric, we combined intrinsic measures, external validation metrics, visual assessment, cluster cardinality analysis and within-cluster deviation statistics. This provides a more complete assessment of clustering behaviour and highlights that different algorithms exhibit different trade-offs between cluster compactness, separation and agreement with reference labels. Such multi-perspective evaluation is particularly valuable for hyperspectral remote sensing, where no single metric fully captures segmentation quality.

Several limitations should nevertheless be acknowledged. First, the experiments were conducted on two widely used benchmark datasets. Although Indian Pines and Pavia University are standard evaluation datasets, future work should include larger airborne and satellite hyperspectral datasets acquired under varying atmospheric conditions, spatial resolutions and land-cover complexity. Second, the number of clusters was fixed according to the available ground-truth labels. Automatic estimation of the optimal number of clusters



(a) K-Means (b) Mini-Batch K-means (c) Bisecting K-means

(d) BIRCH (e) Gaussian Mixture

Fig. 5: Predictions of various clustering algorithms on the Pavia University dataset

remains an important research problem for practical unsupervised remote sensing applications. Third, only spectral information was considered during clustering. Recent studies have shown that incorporating spatial context through spectral–spatial clustering or deep representation learning substantially improves segmentation accuracy by preserving object boundaries and reducing the effects of mixed pixels [**?**], [62]. Finally, although t-SNE and Incremental PCA provided effective feature representations, recent nonlinear manifold learning methods such as UMAP and deep autoencoders deserve further investigation within the proposed comparative framework.

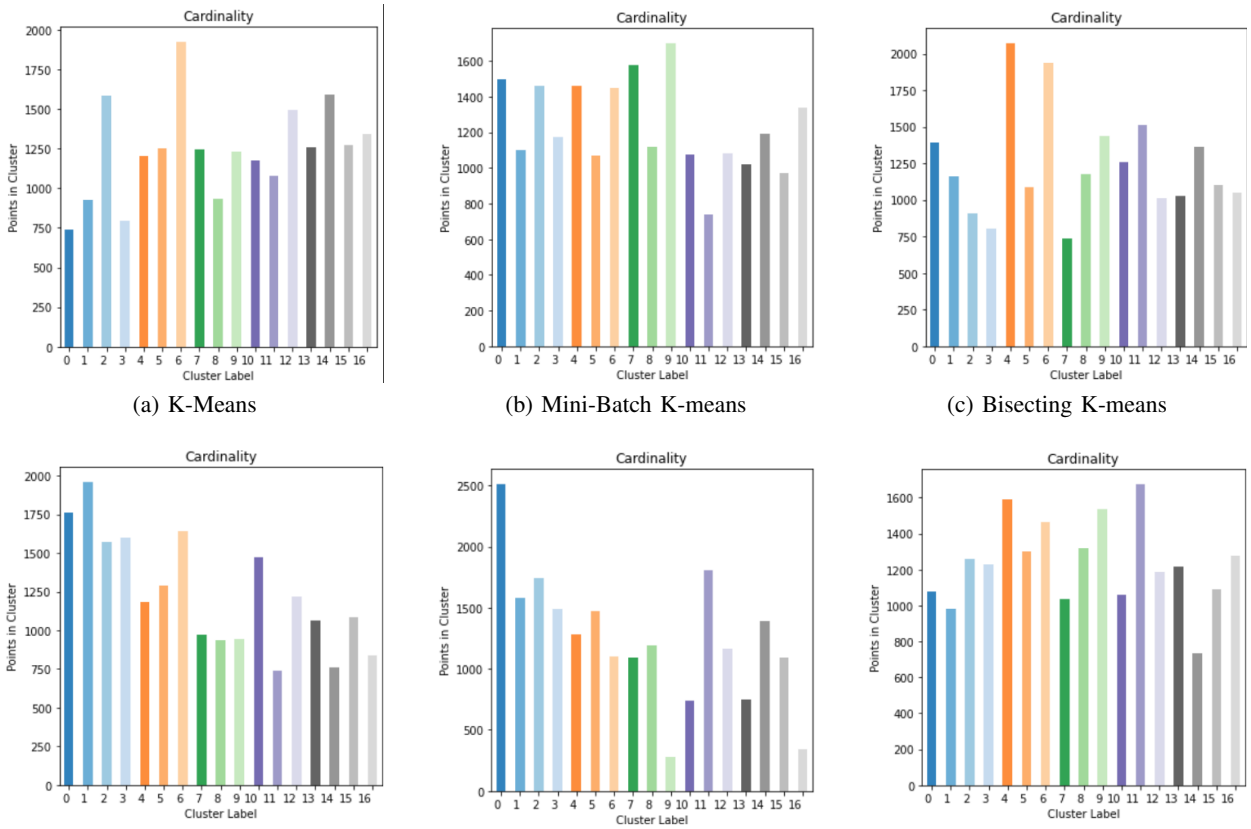
Overall, the results demonstrate that carefully designed preprocessing combined with computationally efficient clustering algorithms provides a strong baseline for hyperspectral image segmentation. Rather than focusing exclusively on increasingly sophisticated clustering algorithms, future research should investigate adaptive feature learning, automatic cluster estimation, spectral–spatial integration and large-scale benchmarking to better understand the relative importance of data representation and clustering methodology in modern hyperspectral remote sensing.

The primary contributions of this work are as follows:

- We present a unified experimental framework for evaluating classical clustering algorithms for hyperspectral image segmentation under consistent preprocessing, hyperparameter optimisation and evaluation protocols.
- We provide one of the most comprehensive empirical comparisons of six representative clustering algorithms across two widely used hyperspectral benchmark datasets using both intrinsic and extrinsic clustering metrics together with qualitative visual analysis.
- We demonstrate that computationally efficient centroid-based clustering methods remain highly competitive with more sophisticated probabilistic and hierarchical approaches when appropriate dimensionality reduction is employed.
- We identify practical trade-offs between clustering quality, robustness and computational efficiency, providing guidance for selecting clustering algorithms in operational hyperspectral remote sensing applications.

V. Conclusions

This study presented a systematic comparison of six widely used clustering algorithms for hyperspectral image segmentation under a common dimensionality reduction framework. Experiments conducted on the Indian Pines and Pavia University benchmarks demonstrated that conventional centroid-based approaches remain highly competitive despite the availability of more sophisticated clustering techniques. Among the evaluated methods, K-means consistently provided the strongest balance between clustering quality, robustness, and computational efficiency, while Mini-Batch K-means achieved comparable performance with substantially improved scalability for larger datasets. Gaussian Mixture Models also



(a) K-Means (b) Mini-Batch K-means (c) Bisecting K-means

(d) BIRCH (e) Agglomerative Clustering (f) Gaussian Mixture

Fig. 6: Cardinality Plots of various clustering algorithms on the Indian Pines dataset

achieved competitive external validation scores but showed greater sensitivity to initialisation and parameter estimation. In contrast, hierarchical approaches, including Bisecting K-means and BIRCH, generally produced less consistent segmentation quality across the evaluated datasets.

These findings suggest that effective dimensionality reduction plays a more influential role than increasing clustering algorithm complexity for many hyperspectral remote sensing applications. Consequently, practitioners can often obtain strong baseline performance using computationally efficient clustering methods combined with suitable feature extraction.

Future research can investigate automatic estimation of the optimal number of clusters, incorporation of spectral–spatial information, evaluation on larger hyperspectral benchmarks, and integration of deep representation learning with scalable clustering algorithms to improve robustness across diverse environmental conditions.

VI. Code and Data

Link GitHub: [1]

References

[1] J. B. Campbell and R. H. Wynne, *Introduction to remote sensing*. Guilford press, 2011.

[2] K. Wang, S. E. Franklin, X. Guo, and M. Cattet, "Remote sensing of ecology, biodiversity and conservation: A review from the perspective of remote sensing specialists," *Sensors*, vol. 10, no. 11, pp. 9647–9667, 2010.

[3] M. Weiss, F. Jacob, and G. Duveiller, "Remote sensing for agricultural applications: A meta-review," *Remote Sensing of Environment*, vol. 236, p. 111402, 2020.

[4] A. Bégué, D. Arvor, B. Bellon, J. Betbeder, D. De Abelleyra, R. P. D. Ferraz, V. Lebourgeois, C. Lelong, M. Simões, and S. R. Verón, "Remote sensing and cropping practices: A review," *Remote Sensing*, vol. 10, no. 1, 2018.

[5] J. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. Nasrabadi, and J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges," *Geoscience and Remote Sensing Magazine, IEEE*, vol. 1, pp. 6–36, 06 2013.

[6] D. Chutia, D. Bhattacharyya, K. K. Sarma, R. Kalita, and S. Sudhakar, "Hyperspectral remote sensing classifications: a perspective survey," *Transactions in GIS*, vol. 20, pp. 463–490, 2015.

[7] A. F. Goetz, "Three decades of hyperspectral remote sensing of the earth: A personal view," *Remote Sensing of Environment*, vol. 113, pp. S5–S16, 2009. Imaging Spectroscopy Special Issue.

[8] M. Govender, K. Chetty, and H. Bulcock, "A review of hyperspectral remote sensing and its application in vegetation and water resource studies," *Water S.A*, vol. 33, 05 2007.

[9] R. Sahoo, S. Ray, and M. R, "Hyperspectral remote sensing of agriculture," *Current science*, vol. 108, pp. 848–859, 03 2015.

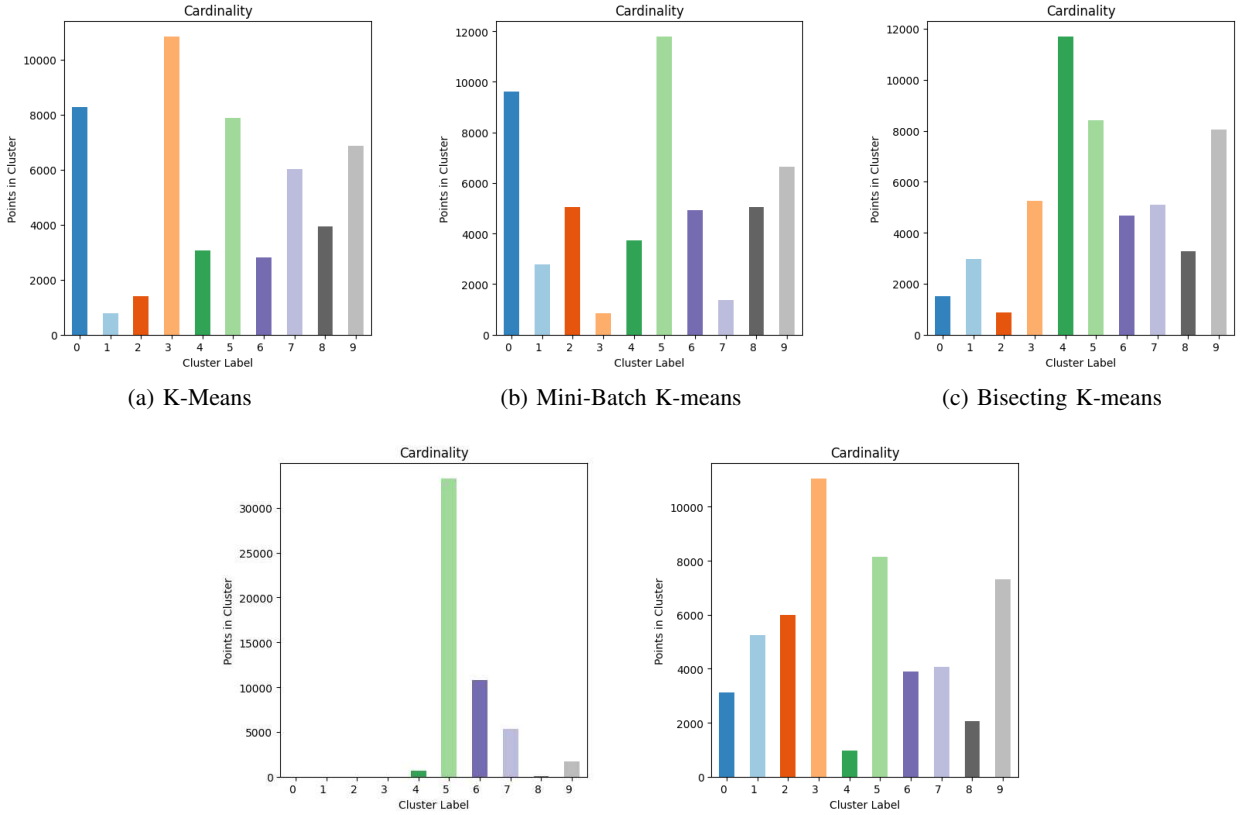
[10] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015.

[11] B. Mahesh, "Machine learning algorithms-a review," *International Journal of Science and Research (IJSR).[Internet]*, vol. 9, no. 1, pp. 381–386, 2020.

[12] Mitchell and T. M, *Machine Learning*. McGraw-Hill, 1997.

[13] Q. Yuan, H. Shen, T. Li, Z. Li, S. Li, Y. Jiang, H. Xu, T. Weiwei, Q. Yang, J. Wang, J. Gao, and L. Zhang, "Deep learning in environmental remote sensing: Achievements and challenges," *Remote Sensing of Environment*, vol. 241, p. 111716, 05 2020.

[1] https://github.com/DARE-ML/hyperspectralclustering



(a) K-Means (b) Mini-Batch K-means (c) Bisecting K-means

(d) BIRCH (e) Gaussian Mixture

Fig. 7: Cardinality Plots of various clustering algorithms on Pavia University dataset

[14] J. Wang, M. Bretz, M. Dewan, and M. Aghajani Delavar, "Machine learning in modelling land-use and land cover-change (lulcc): Current status, challenges and prospects," *Science of The Total Environment*, vol. 822, p. 153559, 01 2022.

[15] H. Shirmard, E. Farahbakhsh, R. D. Müller, and R. Chandra, "A review of machine learning in processing remote sensing data for mineral exploration," *Remote Sensing of Environment*, vol. 268, p. 112750, 2022.

[16] I. Ali, F. Greifeneder, J. Stamenkovic, M. Neumann, and C. Notarnicola, "Review of machine learning approaches for biomass and soil moisture retrievals from remote sensing data," *Remote Sensing*, vol. 7, 11 2015.

[17] L. Zheng, R. Lin, X. Wang, and W. Chen, "The development and application of machine learning in atmospheric environment studies," *Remote Sensing*, vol. 13, p. 4839, 11 2021.

[18] K. Kuwata and R. Shibasaki, "Estimating crop yields with deep learning and remotely sensed data," in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 858–861, 2015.

[19] A. K. Saxena, M. Prasad, A. Gupta, N. Bharill, O. P. Patel, A. Tiwari, M. J. Er, W. Ding, and C.-T. Lin, "A review of clustering techniques and developments," *Neurocomputing*, vol. 267, pp. 664–681, 2017.

[20] G. W. Milligan and M. C. Cooper, "Methodology review: Clustering methods," *Applied psychological measurement*, vol. 11, no. 4, pp. 329–354, 1987.

[21] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Comput. Surv.*, vol. 31, p. 264–323, sep 1999.

[22] C. Wemmert, A. Puissant, G. Forestier, and P. Gancarski, "Multiresolution remote sensing image clustering," *Geoscience and Remote Sensing Letters, IEEE*, vol. 6, pp. 533 – 537, 08 2009.

[23] C. Eick, N. Zeidat, and Z. Zhao, "Supervised clustering - algorithms and benefits," in *16th IEEE International Conference on Tools with Artificial Intelligence*, pp. 774–776, 2004.

[24] V. Estivill-Castro, "Why so many clustering algorithms: A position paper," *SIGKDD Explor. Newsl.*, vol. 4, p. 65–75, jun 2002.

[25] T. Tran, R. Wehrens, and L. Buydens, "Sparef: A clustering algorithm for multispectral images," *Analytica Chimica Acta*, vol. 490, pp. 303–312, 08 2003.

[26] T. Tran, R. Wehrens, and L. Buydens, "Clustering multispectral images: A tutorial," *Chemometrics and Intelligent Laboratory Systems*, vol. 77, pp. 3–17, 05 2005.

[27] S. Zhang and J. M. Murphy, "Hyperspectral image clustering with spatially-regularized ultrametrics," *Remote Sensing*, vol. 13, no. 5, 2021.

[28] H. Zhang, H. Zhai, L. Zhang, and P. Li, "Spectral–spatial sparse subspace clustering for hyperspectral remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, pp. 1–13, 03 2016.

[29] C. Cariou, S. Le Moan, and K. Chehdi, "Improving k-nearest neighbor approaches for density-based pixel clustering in hyperspectral remote sensing images," *Remote Sensing*, vol. 12, 11 2020.

[30] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. Nasrabadi, and J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geoscience and remote sensing magazine*, vol. 1, no. 2, pp. 6–36, 2013.

[31] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, no. 1, pp. 37–52, 1987. Proceedings of the Multivariate Statistical Workshop for Geologists and Geochemists.

[32] A. Krizhevsky and G. E. Hinton, "Using very deep autoencoders for content-based image retrieval.," in *ESANN*, vol. 1, p. 2, Citeseer, 2011.

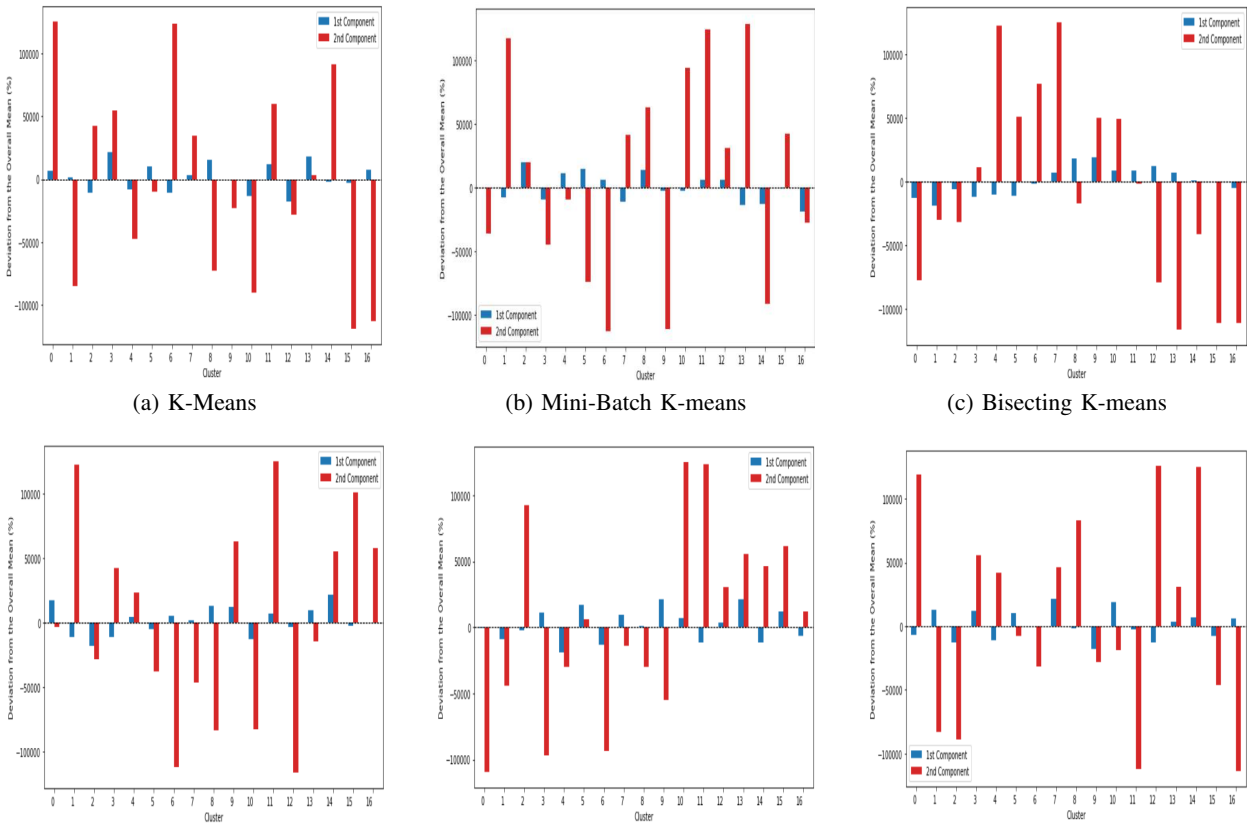
[33] M. F. Baumgardner, L. L. Biehl, and D. A. Landgrebe, "220 Band AVIRIS Hyperspectral Image Data Set: June 12, 1992 Indian Pine Test Site 3," Sep 2015.

[34] G. Vane, R. O. Green, T. G. Chrien, H. T. Enmark, E. G. Hansen, and W. M. Porter, "The airborne visible/infrared imaging spectrometer (AVIRIS)," *Remote sensing of environment*, vol. 44, no. 2-3, pp. 127–143, 1993.

[35] G. E. Hinton and S. Roweis, "Stochastic neighbor embedding," *Advances in neural information processing systems*, vol. 15, 2002.

[36] M. C. Cieslak, A. M. Castelfranco, V. Roncalli, P. H. Lenz, and D. K. Hartline, "t-distributed stochastic neighbor embedding (t-sne): A tool for eco-physiological transcriptomic analysis," *Marine genomics*, vol. 51, p. 100723, 2020.

[37] T. Horrocks, E.-J. Holden, D. Wedge, C. Wijns, and M. Fiorentini,



(a) K-Means (b) Mini-Batch K-means (c) Bisecting K-means

(d) BIRCH (e) Agglomerative Clustering (f) Gaussian Mixture

Fig. 8: Deviation Plots of various clustering algorithms on the Indian Pines dataset

"Geochemical characterisation of rock hydration processes using t-sne," *Computers & geosciences*, vol. 124, pp. 46–57, 2019.

[38] M. Balamurali, K. L. Silversides, and A. Melkumyan, "A comparison of t-sne, som and spade for identifying material type domains in geological data," *Computers & Geosciences*, vol. 125, pp. 78–89, 2019.

[39] W. Song, L. Wang, P. Liu, and K.-K. R. Choo, "Improved t-sne based manifold dimensional reduction for remote sensing data processing," *Multimedia Tools and Applications*, vol. 78, pp. 4311–4326, 2019.

[40] A. C. Belkina, C. O. Ciccolella, R. Anno, R. Halpert, J. Spidlen, and J. E. Snyder-Cappione, "Automated optimized parameters for t-distributed stochastic neighbor embedding improve visualization and analysis of large datasets," *Nature communications*, vol. 10, no. 1, p. 5415, 2019.

[41] H. Zhai, H. Zhang, P. Li, and L. Zhang, "Hyperspectral image clustering: Current achievements and future lines," *IEEE Geoscience and Remote Sensing Magazine*, vol. 9, no. 4, pp. 35–67, 2021.

[42] J. M. Haut, M. Paoletti, J. Plaza, and A. Plaza, "Cloud implementation of the k-means algorithm for hyperspectral image analysis," *The Journal of Supercomputing*, vol. 73, pp. 514–529, 2017.

[43] N. Ishidoshiro, Y. Yamaguchi, S. Noda, Y. Asano, T. Kondo, Y. Kawakami, M. Mitsuishi, and H. Nakamura, "Geological mapping by combining spectral unmixing and cluster analysis for hyperspectral data," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 41, pp. 431–435, 2016.

[44] J. D. Paola and R. A. Schowengerdt, "The effect of neural-network structure on a multispectral land-use/land-cover classification," *Photogrammetric Engineering and Remote Sensing*, vol. 63, no. 5, pp. 535–544, 1997.

[45] D. Sculley, "Web-scale k-means clustering," in *Proceedings of the 19th international conference on World wide web*, pp. 1177–1178, 2010.

[46] J. Béjar Alonso, "K-means vs mini batch k-means: a comparison," 2013.

[47] S. M. Savaresi and D. L. Boley, "On the performance of bisecting k-means and PDDP," in *Proceedings of the 2001 SIAM International Conference on Data Mining*, pp. 1–14, SIAM, 2001.

[48] S. Banerjee, A. Choudhary, and S. Pal, "Empirical evaluation of k-means, bisecting k-means, fuzzy c-means and genetic k-means clustering algorithms," in *2015 IEEE international WIE conference on electrical and computer engineering (WIECON-ECE)*, pp. 168–172, IEEE, 2015.

[49] A. Lukasová, "Hierarchical agglomerative clustering procedure," *Pattern Recognition*, vol. 11, no. 5-6, pp. 365–381, 1979.

[50] F. Murtagh and P. Legendre, "Ward's hierarchical agglomerative clustering method: which algorithms implement ward's criterion?," *Journal of classification*, vol. 31, pp. 274–295, 2014.

[51] O. Yim and K. T. Ramdeen, "Hierarchical cluster analysis: comparison of three linkage measures and application to psychological data," *The quantitative methods for psychology*, vol. 11, no. 1, pp. 8–21, 2015.

[52] F. Ros and S. Guillaume, "A hierarchical clustering algorithm and an improvement of the single linkage criterion to deal with noise," *Expert Systems with Applications*, vol. 128, pp. 96–108, 2019.

[53] M. Forina, C. Armanino, and V. Raggio, "Clustering with dendrograms on interpretation variables," *Analytica Chimica Acta*, vol. 454, no. 1, pp. 13–19, 2002.

[54] M. Goncalves, M. Netto, J. Costa, and J. Zullo Junior, "An unsupervised method of classifying remotely sensed images using kohonen self-organizing maps and agglomerative hierarchical clustering methods," *International Journal of Remote Sensing*, vol. 29, no. 11, pp. 3171–3207, 2008.

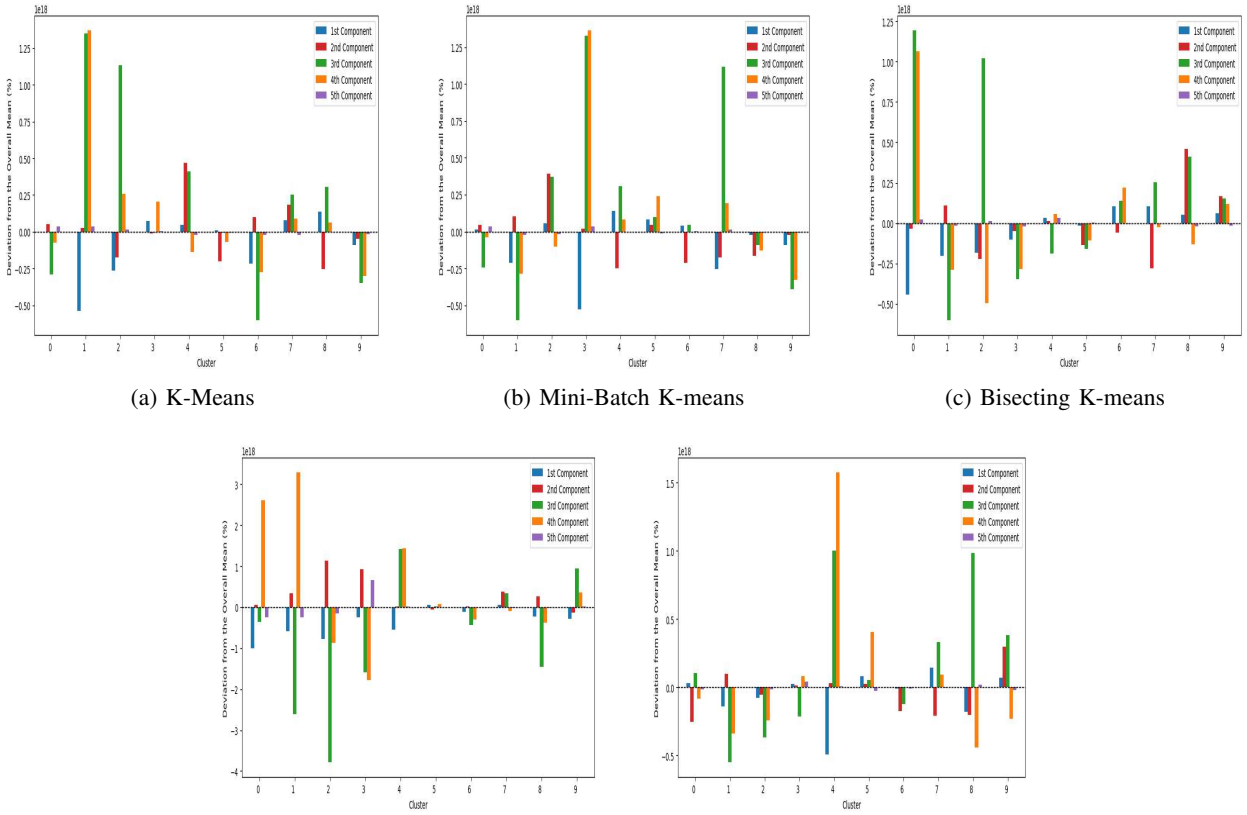
[55] H. Wilson, B. Boots, and A. A. Millward, "A comparison of hierarchical and partitional clustering techniques for multispectral image classification," in *IEEE International Geoscience and Remote Sensing Symposium*, vol. 3, pp. 1624–1626, Ieee, 2002.

[56] J. Senthilnath, S. Omkar, V. Mani, N. Tejovanth, P. Diwakar, and A. B. Shenoy, "Hierarchical clustering algorithm for land cover mapping using satellite images," *IEEE journal of selected topics in applied earth observations and remote sensing*, vol. 5, no. 3, pp. 762–768, 2012.

[57] T. Zhang, R. Ramakrishnan, and M. Livny, "Birch: an efficient data clustering method for very large databases," *ACM sigmod record*, vol. 25, no. 2, pp. 103–114, 1996.

[58] A. Lang and E. Schubert, "Betula: numerically stable cf-trees for birch clustering," in *International Conference on Similarity Search and Applications*, pp. 281–296, Springer, 2020.

[59] R. Fernandez-Beltran, J. M. Haut, M. E. Paoletti, J. Plaza, A. Plaza,



(a) K-Means (b) Mini-Batch K-means (c) Bisecting K-means

(d) BIRCH (e) Gaussian Mixture

Fig. 9: Deviation Plots of various clustering algorithms on Pavia University dataset

and F. Pla, "Remote sensing image fusion using hierarchical multimodal probabilistic latent semantic analysis," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 12, pp. 4982–4993, 2018.

[60] G. J. McLachlan and K. E. Basford, *Mixture models: Inference and applications to clustering*, vol. 38. M. Dekker New York, 1988.

[61] J. A. Benediktsson and P. Ghamisi, *Spectral-spatial classification of hyperspectral remote sensing images*. Artech House, 2015.

[62] B. Zhao, Y. Zhong, A. Ma, and L. Zhang, "A spatial gaussian mixture model for optical remote sensing image clustering," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 12, pp. 5748–5759, 2016.

[63] S. Yin, Y. Zhang, and S. Karim, "Large scale remote sensing image segmentation based on fuzzy region competition and gaussian mixture model," *IEEE Access*, vol. 6, pp. 26069–26080, 2018.

[64] S. Barve, J. M. Webster, and R. Chandra, "Reef-insight: A framework for reef habitat mapping with clustering methods using remote sensing," *Information*, vol. 14, no. 7, p. 373, 2023.

[65] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical association*, vol. 66, no. 336, pp. 846–850, 1971.

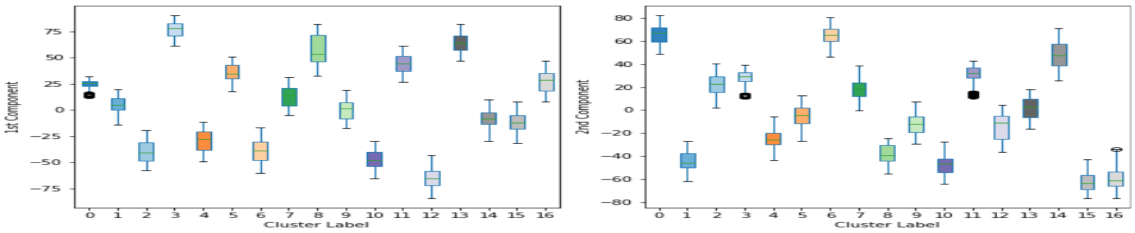
[66] L. Hubert and P. Arabie, "Comparing partitions," *Journal of classification*, vol. 2, pp. 193–218, 1985.

[67] S. Romano, J. Bailey, V. Nguyen, and K. Verspoor, "Standardized mutual information for clustering comparisons: one step further in adjustment for chance," in *International conference on machine learning*, pp. 1143–1151, PMLR, 2014.

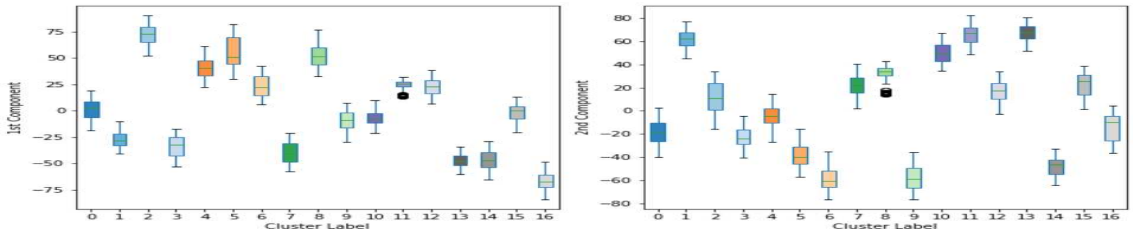
[68] A. Rosenberg and J. Hirschberg, "V-measure: A conditional entropy-based external cluster evaluation measure," in *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, pp. 410–420, 2007.

[69] C. Van Rijsbergen, "Information retrieval: theory and practice," in *Proceedings of the Joint IBM/University of Newcastle upon Tyne Seminar on Data Base Systems*, vol. 79, 1979.

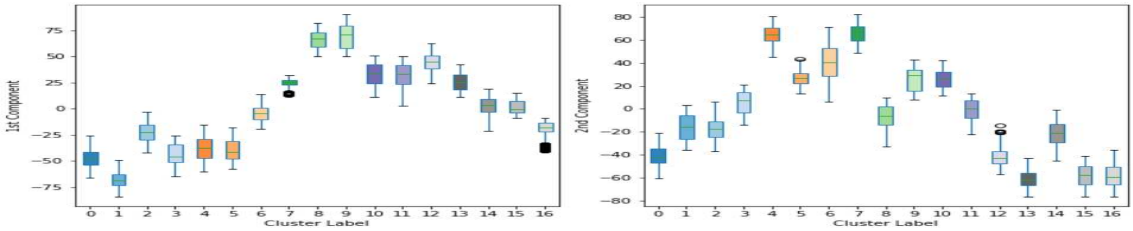
APPENDIX



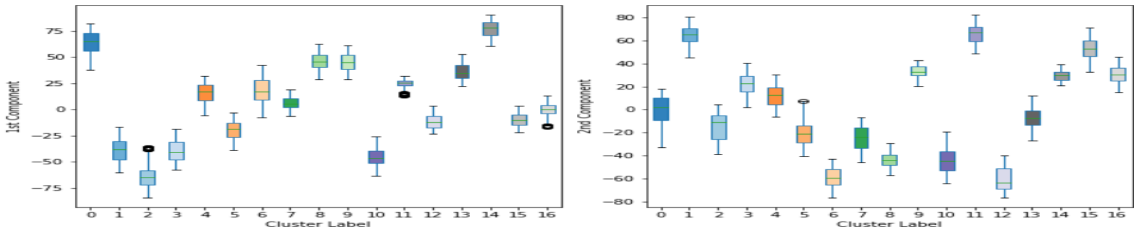
(a) K-Means



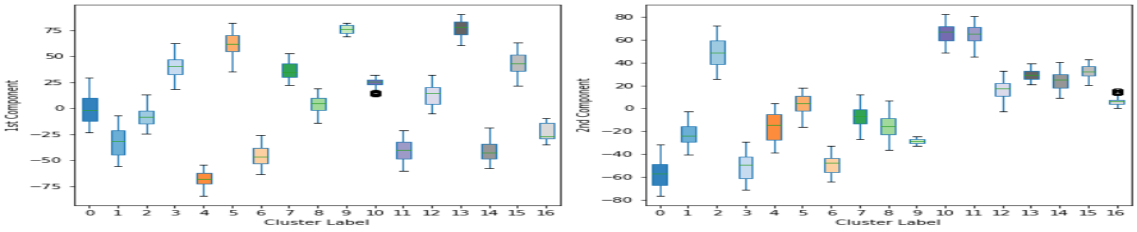
(b) Mini-Batch K-means



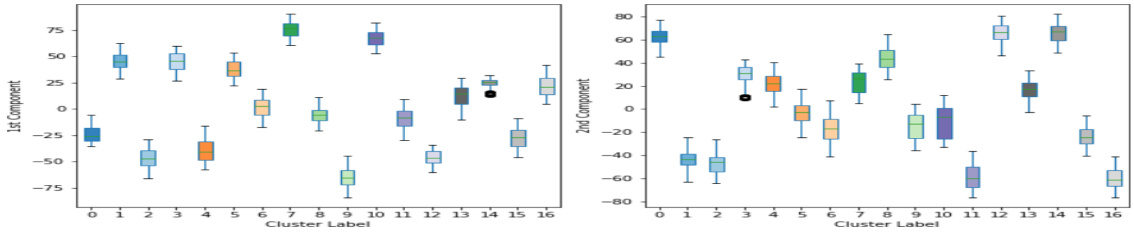
(c) Bisecting K-means



(d) BIRCH

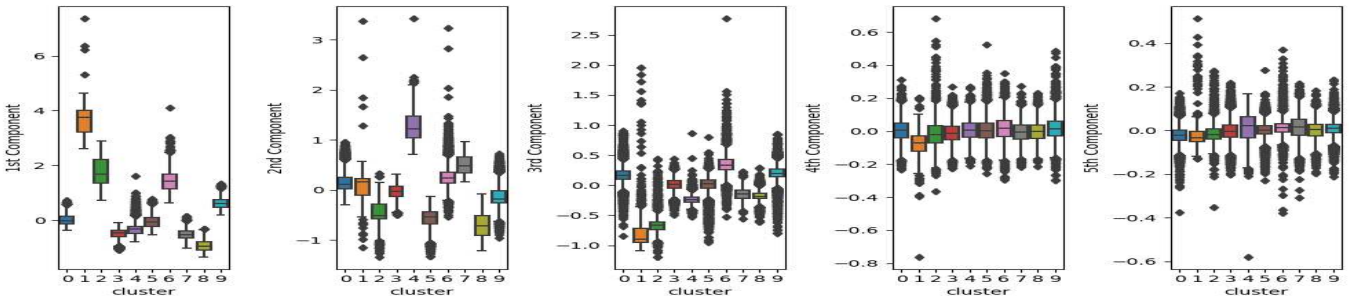


(e) Agglomerative Clustering

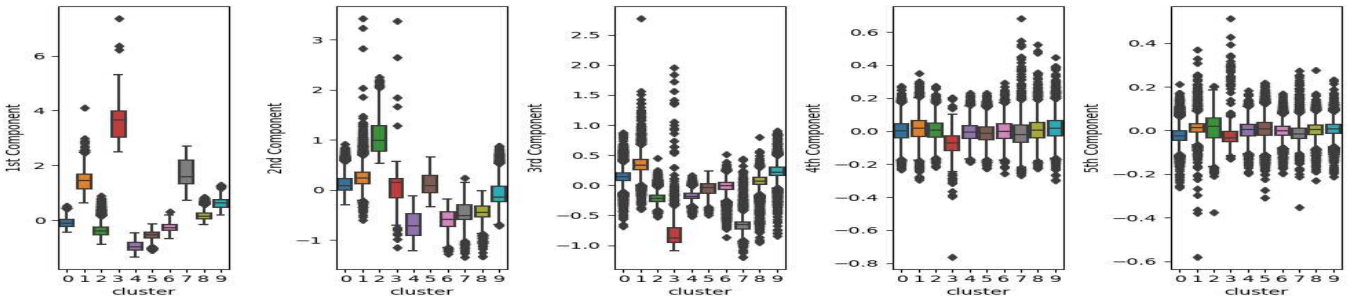


(f) Gaussian Mixture

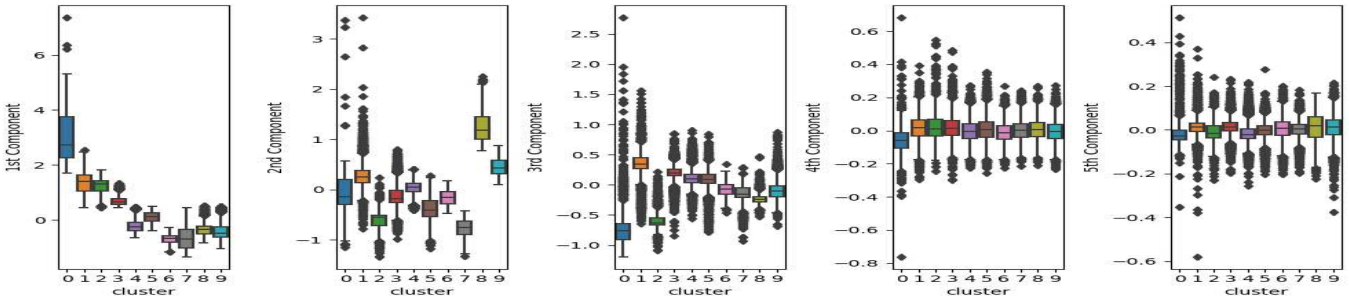
Fig. 10: Box Plots of various clustering algorithms on Indian Pines dataset



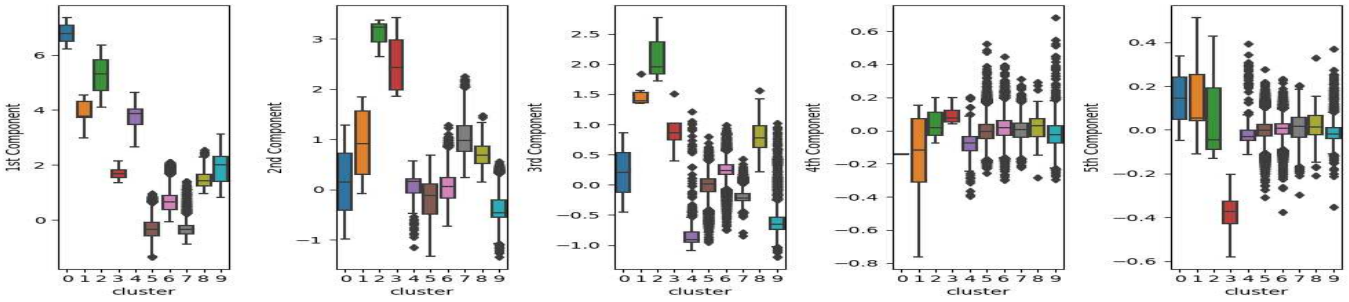
(a) K-Means



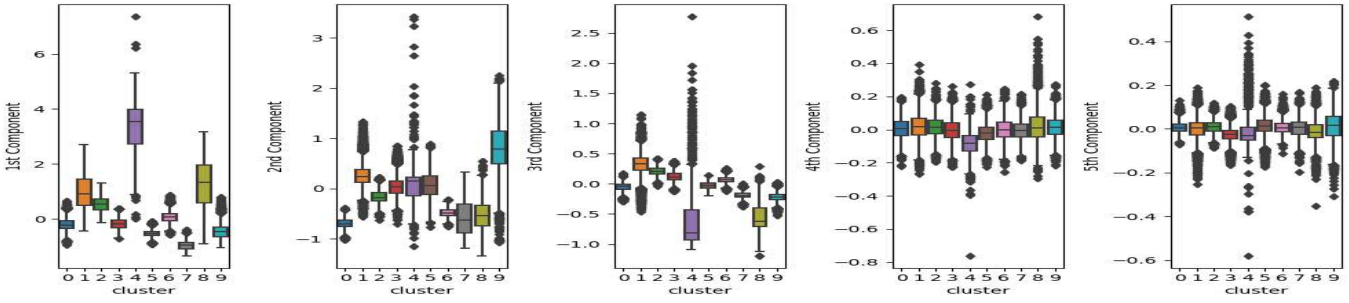
(b) Mini-Batch K-means



(c) Bisecting K-means



(d) BIRCH



(e) Gaussian Mixture

Fig. 11: Box Plots of various clustering algorithms on Pavia University dataset