ISSN 2616-5775 Vol. 7, Issue 6: 20-31, DOI: 10.25236/AJCIS.2024.070604

Convolutional Autoencoder-based Deep Embedded Fuzzy Clustering Using H-divergences

Tianzhen Chen*

EIT Data Science and Communication College, Zhejiang Yuexiu University of Foreign Languages, Shaoxing, China
**Corresponding author: terrychen17@outlook.com*

Abstract: *With the rapid development of deep learning technologies, the integration of fuzzy clustering methods with convolutional autoencoders has offered new perspectives for data clustering. This paper introduces a novel deep learning methodology, "Convolutional Autoencoder-based Deep Embedded Fuzzy Clustering Using H-divergences, (CADEFC)". In response to the rapid evolution of deep learning, this study combines fuzzy clustering methods with convolutional autoencoders to explore new dimensions in data clustering. We critically assess the limitations of the Wasserstein distance as a conventional loss function and propose the adoption of H-divergences as a more robust alternative. By integrating H-divergence and introducing fuzzy theory, our method transcends the traditional constraints of deep clustering techniques, offering substantial improvements in clustering accuracy and stability. The effectiveness and superiority of this approach are demonstrated through rigorous testing on several benchmark datasets, including Digits, Fashion-MNIST, MNIST, and USPS. Our results confirm that the proposed methodology not only enhances performance but also broadens the applicative landscape of deep embedded clustering.*

Keywords: *Unsupervised learning, pattern recognition, deep clustering, H-divergence*

1. Introduction

Data clustering, a pivotal component of unsupervised learning, plays a crucial role in various applications such as image recognition and speech processing. Fuzzy clustering, by allowing data points to belong to multiple clusters, offers a more flexible tool for data analysis compared to traditional clustering methods. In recent years, clustering techniques that integrate deep learning—particularly those based on convolutional autoencoders—have demonstrated remarkable capabilities in handling complex datasets. However, traditional loss functions like the Wasserstein distance exhibit limitations, especially under conditions of insufficient distributional support, leading to decreased performance.

Background on Fuzzy Clustering and Convolutional Autoencoders: Fuzzy clustering extends traditional approaches by incorporating the concept of partial membership, which provides a nuanced understanding of data points' affiliations with multiple clusters [1]. Convolutional autoencoders, renowned for their efficacy in feature extraction from structured data like images, serve as the backbone for our deep clustering framework. By harnessing the power of deep convolutional networks, these autoencoders efficiently capture the intricate patterns and relationships within data, facilitating more accurate and robust clustering [2].

Limitations of Wasserstein Distance in Clustering: While the Wasserstein distance has been a popular choice for measuring discrepancies between distributions in clustering scenarios, it encounters significant challenges when the underlying data distributions lack sufficient overlap or are sparsely supported [2]. Such scenarios often lead to suboptimal clustering outcomes, reflecting the need for alternative approaches that can provide more reliable results across diverse distributional characteristics.

Introduction of H-divergences: To address these shortcomings, we introduce H-divergences as an innovative divergence metric within our clustering methodology. H-divergences are adept at measuring differences between distributions, even in cases where traditional metrics like the Wasserstein distance falter [3]. By incorporating H-divergences into our deep embedded fuzzy clustering framework, we aim to enhance the model's ability to discern and adapt to complex distributional shifts, thereby improving

ISSN 2616-5775 Vol. 7, Issue 6: 20-31, DOI: 10.25236/AJCIS.2024.070604

clustering accuracy and robustness.

This paper explores the integration of H-divergences into deep convolutional autoencoder-based clustering, aiming to provide a sophisticated approach that not only addresses the inherent limitations of previous metrics but also leverages the strengths of modern deep learning architectures to advance the field of data clustering.

2. Related work

This section provides a comprehensive review of the advancements in fuzzy clustering, deep convolutional autoencoder-based clustering, and their associated loss functions, with a particular focus on the application of Wasserstein distances and H-divergences in clustering. By analyzing the strengths and limitations of existing methods in specific scenarios, we establish a theoretical foundation for the research presented in this paper.

2.1 Advancements in deep learning for clustering

Recent advancements in the field of deep learning have substantially influenced the development of clustering methods, particularly through the integration of deep convolutional networks with fuzzy clustering techniques. This synthesis of deep learning and fuzzy logic in clustering not only improves the robustness and adaptability of clustering methods but also opens new avenues for research and application. For instance, in fields like bioinformatics, customer segmentation, and image processing, where data ambiguity and complexity are prevalent. This fusion, known as deep convolutional embedded fuzzy clustering (DCEFC) [4], has marked a significant step forward in both the theory and application of clustering methods.

Deep convolutional networks, renowned for their superior capability in feature extraction from complex datasets, such as images and videos, have been effectively adapted for clustering purposes. These networks facilitate the discernment of intricate patterns and relationships within data, which traditional clustering methods might overlook. The work of Alqahtani et al. (2018) and Du et al. (2023) exemplifies the application of these deep learning architectures, demonstrating enhanced clustering performance in terms of both accuracy and computational efficiency [5,6].

Moreover, the incorporation of fuzzy logic into deep learning-based clustering frameworks addresses the often rigid cluster assignment problem inherent in many conventional methods. Fuzzy clustering allows for soft cluster membership, which is a more realistic representation of the ambiguities present in real-world data. As highlighted by Tan et al. (2023), this approach provides a more flexible and realistic clustering process, where data points can belong to multiple clusters to varying degrees, thus reflecting the probabilistic nature of many real-world scenarios [7].

The progress in this area suggests a promising direction for further research, particularly in exploring the scalability of these models to larger datasets and their applicability across different domains. Future studies might focus on enhancing the interpretability of the clusters generated by such models, thus making them more actionable for decision-making processes in business and science [2].

2.2 Review of Wasserstein distance and other loss functions

The Wasserstein distance, recognized for its capacity to measure the discrepancies between probability distributions, has emerged as a preferred loss function in the realm of clustering methods. This metric, grounded in the principles of optimal transport, has been shown to effectively capture the true distances between complex distributions, which is often a challenging feat for traditional metrics. As highlighted by Bischoff et al. (2024), the Wasserstein distance offers a robust framework for assessing dissimilarities in data characterized by varied probability distributions, enhancing the precision of clustering outcomes [8].

The Deep Convolutional Embedded Fuzzy Clustering with Wasserstein Loss (DCEFC) [4] stands as a sophisticated progression from traditional Deep Embedding Clustering (DEC). This advanced model integrates the concept of Wasserstein distance as the clustering phase's loss function. The selection of the Wasserstein distance is informed by its proficiency in accommodating the geometric intricacies of data distributions, which significantly enhances the accuracy of cluster assignments. Furthermore, DCEFC incorporates the fuzzy parameter m, a concept pivotal in fuzzy clustering. This addition allows for a more flexible membership assignment within clusters, thereby refining the

ISSN 2616-5775 Vol. 7, Issue 6: 20-31, DOI: 10.25236/AJCIS.2024.070604

clustering process by acknowledging and managing the degrees of uncertainty inherent in data points. This dual incorporation of Wasserstein distance and fuzzy logic considerably elevates the clustering performance of DCEFC over traditional methods.

Despite its advantages, the Wasserstein distance is not devoid of limitations. One significant challenge arises when this metric is employed in datasets with sparse or disjoint distributional supports. In such cases, the Wasserstein distance can struggle to accurately gauge the true disparities between clusters. This issue is particularly problematic in scenarios where clusters are not well-defined or are separated by non-overlapping supports, leading to potentially misleading clustering results. Cai et al. (2024) explore these challenges in depth, suggesting that while the Wasserstein distance is powerful, its effectiveness may be contingent upon the structural characteristics of the data [9].

In addition to the Wasserstein distance, other divergence measures like the Kullback-Leibler (KL) divergence and the Jensen-Shannon (JS) divergence have also been investigated for their potential in clustering applications. Both divergences, rooted in information theory, provide alternative means of measuring the information loss when one probability distribution is used to approximate another. However, as Cai and Yuhang (2024) discuss, these measures also encounter limitations, particularly in handling data complexities such as multimodal distributions where multiple subgroups exist within a single cluster [10].

The exploration of these loss functions highlights a critical area of research in clustering: the need for a versatile, accurate, and computationally feasible approach to measure distances between distributions in varied data environments. Future research might focus on developing new metrics or enhancing existing ones to overcome these challenges. Enhancements could involve hybrid approaches that combine the strengths of Wasserstein and other divergences, or entirely novel methods that are designed to be more adaptive to the specific properties of the dataset [11].

2.3 Applications of H-divergences in various domains

The application of H-divergences in various domains, stemming from their origin in statistical hypothesis testing, has garnered considerable attention in machine learning. Particularly noteworthy are their roles in domain adaptation, generative adversarial networks (GANs), and clustering frameworks [11].

Shui et al. (2020) shed light on the utility of H-divergences in assessing the similarity between source and target distributions, crucial for ensuring the efficacy of learning models in domain adaptation scenarios [12]. Their studies demonstrate that models adapted using H-divergence-based strategies exhibit superior performance, especially when faced with significant disparities between the underlying data distributions of training and testing sets.

In the realm of generative adversarial networks, Goel et al. (2020) showcase the instrumental role of H-divergences in refining the training process. By leveraging H-divergences, they enhance the generator's capability to produce synthetic data points that closely resemble real data, thus advancing the applicability of GANs in areas such as image synthesis and data augmentation [13].

Furthermore, Rey et al. (2022) delve into the adaptability of H-divergences in clustering frameworks, emphasizing their effectiveness in environments with non-overlapping distributions. This characteristic renders H-divergences a robust alternative to traditional metrics like Euclidean distance or the Kullback-Leibler divergence, particularly when capturing the true nature of data clusters becomes challenging [14].

Zhao et al. (2022) provide a comprehensive analysis of H-divergences across various machine learning tasks, including clustering, classification, and generative modeling. Their findings underscore the superiority of H-divergences over Kullback-Leibler and Wasserstein distances, especially in scenarios involving non-overlapping distributions or noisy data [15]. Additionally, Goel et al. (2020) contribute to this discourse with their review on advancements in divergence measures for machine learning, highlighting the superior performance of H-divergences in capturing intricate dependencies between distributions [13].

Collectively, these studies bolster the burgeoning literature supporting the adoption of H-divergences as a viable alternative to traditional distance measures. They underscore the potential of H-divergences to enhance the performance of machine learning methods, particularly in scenarios characterized by complex or non-standard distributional characteristics.

ISSN 2616-5775 Vol. 7, Issue 6: 20-31, DOI: 10.25236/AJCIS.2024.070604

The insights gained from these studies provide crucial context for our investigation into the incorporation of H-divergences into deep convolutional autoencoder-based clustering. The next section details the methodology developed in this research, building upon the theoretical insights and empirical findings discussed herein.

3. Proposed method

In this section, we introduce a novel method named Convolutional Autoencoder-based Deep Embedded Fuzzy Clustering Using H-divergences (CADEFC). This section details the network architecture of the method, presents the core mathematical formulations, and provides illustrative diagrams and pseudocode to elucidate the method's implementation.

In our proposed method, we continue to apply the clustering framework of DCEFC [4], which is divided into two distinct phases: the pre-training phase and the clustering phase. Initially, the pre-training phase employs a network architecture comprising three convolutional layers followed by a symmetric mirror structure. This configuration is meticulously designed to batch process datasets and subsequently construct the requisite convolutional autoencoder for pre-training. During this phase, both the parameters (weights and biases) and the hidden features from the final layer of the pre-training encoder are retained.

Following the pre-training phase, the clustering phase commences. Here, the convolutional autoencoder's architecture from the pre-training phase is preserved, and the network parameters from this phase are inherited. The clustering model is then formed by appending a clustering layer subsequent to the pre-training encoder. In our approach, clustering centers are defined by the weights of the hidden features from the last pre-training encoder. Unlike other DCEFC implementations, our clustering is conducted using Fuzzy C-means clustering to derive cluster prediction labels and centers. These labels and centers are utilized to initialize the weights of the clustering layer.

To better introduce our, we provide a schematic diagram of our method as shown in Figure 1, along with a detailed description of the method outlined below.

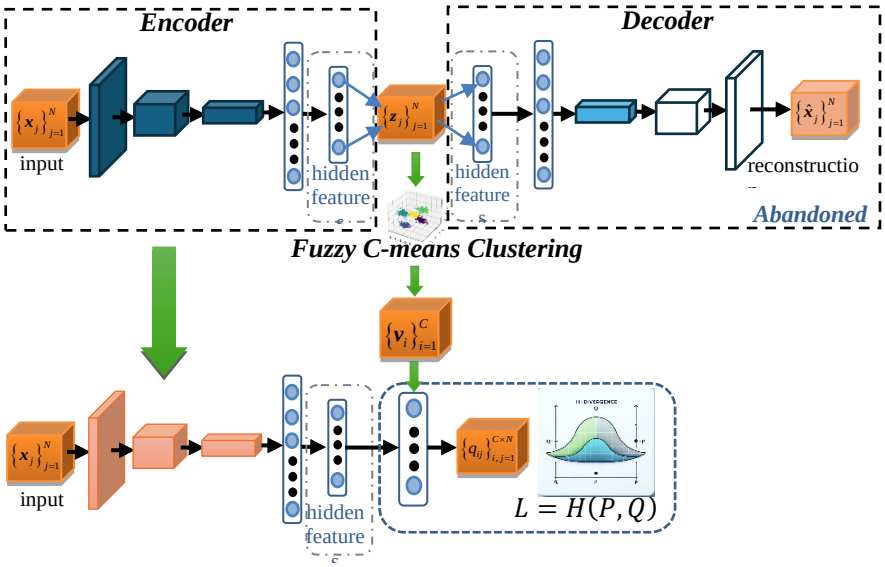

Figure 1. Schematic Diagram of Convolutional Autoencoder-based Deep Embedded Fuzzy Clustering Using H-divergences

To optimize DCEFC outcomes and mitigate the risk of Wasserstein divergence failure, we introduce an H-divergence based approach. This approach provides a rigorous framework for clustering analysis, effectively integrating the novel concept of H-divergences to enhance model performance and reliability. The fuzzy distribution p of embedding points is computed using the formula:

$$p(z_i) = \mathbf{softmax}(-\frac{\|z_i - c_j\|^2}{2\sigma^2}) \tag{1}$$

where z_i represents the embedding points, c_j denotes the cluster centers, and σ is the scale parameter.

ISSN 2616-5775 Vol. 7, Issue 6: 20-31, DOI: 10.25236/AJCIS.2024.070604

Subsequently, the auxiliary distribution $\{q_{ij} \in Q\}_{i,j=1}^{C\times N}$ is calculated as:

$$q_{ij} = \frac{\left(m+\|v_i - z_j\|^2\right)^{\frac{-2}{m-1}}}{\sum_{k=1}^{C}\left(m+\|v_k - z_j\|^2\right)^{\frac{-2}{m-1}}},\ \sum_{i=1}^{C} q_{ij} = 1,\ \forall j \tag{2}$$

To better facilitate the minimization of the H-divergence between the predicted and target distributions.

The minimization of H-divergence is realized through the loss function:

$$L = \sum_i p_i \boldsymbol{log} \frac{p_i}{q_i} + \lambda \mathbf{H} - \mathbf{div}(P\|Q) \tag{3}$$

Where $\sum_i p_i$ and $\lambda\mathbf{H}$ are the distributions of target and clustering output, respectively. Here, $\lambda\mathbf{H}$ acts as a regularization parameter, and the H-divergence $H - \mathrm{div}(P\|Q)$ is defined as:

$$\mathbf{H} - \mathbf{div}(P\|Q) = \sum_{(x,y)} \pi(x,y)\ell(x,y) \tag{4}$$

Which $\pi(x,y)$ denotes the joint distribution that minimizes the expected loss $\ell(x,y)$ between pairs (x,y).

Label updates are governed by the minimization of H-divergence:

$$\mathrm{Label}_i = \mathbf{arg}\min_k \mathbf{H} - \mathbf{div}(p_{ik} \parallel q) \tag{5}$$

Where p_{ik} is the distribution corresponding to the q-th data point for the q -th cluster.

Training is halted when the label difference rate between consecutive updates falls below a predetermined threshold q, reflecting minimal change and convergence:

if $\frac{\sum_i \left|\mathrm{Label}_i^{(t)} - \mathrm{Label}_i^{(t-1)}\right|}{N} < \epsilon$, stop training

Where N is the total number of data points.

The pseudocode is provided below.

Convolutional Autoencoder-based Deep Embedded Fuzzy Clustering Using H-divergences, (CADEFC)

Input Specifications

- **Dataset (X):** A set of data points to be clustered.
- **Number of Samples per Batch (*B*):** The number of data points in each batch during training.
- **Number of Pre-training Iterations (*T*):** The total iterations for the initial encoding and decoding process to prepare the model.
- **Maximum Number of Main Clustering Iterations (*N*):** The upper limit on the number of iterations in the clustering phase.
- **Update Frequency for Target Distribution (*P*):** The number of iterations after which the target distribution is updated.
- **Stopping Threshold (ε):** A predefined threshold for stopping the training process if the change in clustering results or loss falls below this value.

Output Specifications

- **Clustering Centers:** The centroids of the clusters after the completion of the training, representing the typical data points within each cluster.
- **Clustering Pseudo-Labels:** The labels assigned to each data point, indicative of their respective cluster memberships.

Detailed Process Description

1) **Batch Preparation:**
   - Divide the dataset ***X*** into batches, each containing ***B*** samples. Each batch is represented as $^{(b)}$, where b indexes the batch.
2) **Pre-training Loop (Initialization):**
   - For each iteration from 1 to ***T***:
     - For each batch $X^{(b)}$:
       - **Encoder:** Map $^{(b)}$ to a latent space to obtain embeddings $^{(b)}$.

ISSN 2616-5775 Vol. 7, Issue 6: 20-31, DOI: 10.25236/AJCIS.2024.070604

- **Decoder:** Reconstruct $^{(b)}$ from $^{(b)}$, obtaining $\hat{X}^{(b)}$.
- Compute the Mean Squared Error (MSE) between $^{(b)}$ and $\hat{X}^{(b)}$ and optimize the encoder and decoder parameters.

- After pre-training, initialize the clustering centers and pseudo-labels using fuzzy *C*-means on the embeddings obtained from the encoder.

3) **Main Clustering Loop:**
   - For each iteration up to N:
     - **Target Distribution Update:** Every P iterations:
       - Compute embeddings for all data points using the Encoder.
       - Update the fuzzy membership matrix and auxiliary target distribution using predefined formulas (referred to as formula (1) and formula (2)).
       - Store the last set of pseudo-labels and update the current pseudo-labels using the update rule (formula (5)).
       - Compute the stopping criterion (usually some function of the change in cluster assignments or loss).
       - If the stopping criterion (difference $< \epsilon$), terminate training.
     - **Batch Processing:**
       - Shuffle and divide the dataset X into batches again.
       - For each batch $X^{(b)}$:
         - Calculate partial embeddings $^{(b)}$ using the Encoder.
         - Update the fuzzy membership for $^{(b)}$ using the membership update formula (1).
         - Segment $^{(b)}$ into clusters.
         - Calculate the H-divergence and optimize parameters and cluster centers based on this distance (using formula (4)).

This section introduced the Convolutional Autoencoder-based Deep Embedded Fuzzy Clustering Using H-divergences (CADEFC). We outlined the architecture, provided mathematical formulations, and explained the innovative integration of H-divergences for optimizing clustering performance. The upcoming section will empirically validate CADEFC, demonstrating its efficacy across various benchmark datasets.

4. Experimental design and validation

This section outlines the experimental design and validation of our study, focusing on the application of clustering methods across several benchmark datasets. We detail the datasets used, describe the evaluation metrics, and discuss comparative methods to assess the performance and robustness of the proposed CADEFC method against established clustering techniques.

4.1 Introduction to datasets

In this paper, our experimental framework revolves around the utilization of four well-established benchmark datasets, namely Digits, Fashion-MNIST, MNIST, and USPS. These datasets were meticulously chosen for their prominence and representativeness within the domain of digital image processing. Despite sharing the commonality of comprising digital images, each dataset presents unique characteristics, particularly in terms of dimensionality, thereby enriching the breadth of our study.

4.1.1 Digits dataset

The Digits dataset constitutes a foundational component of our experimental setup, characterized by its diverse collection of hand-written digit images. Renowned for its role in machine learning research, this dataset offers a rich assortment of digit images, spanning various writing styles and complexities.

4.1.2 Fashion-MNIST dataset

The Fashion-MNIST emerges as another cornerstone in our experimental design, featuring a curated ensemble of fashion-related images. Renowned for its emulation of the structure and complexity of the original MNIST dataset, Fashion-MNIST presents a challenging yet realistic

ISSN 2616-5775 Vol. 7, Issue 6: 20-31, DOI: 10.25236/AJCIS.2024.070604

benchmark for evaluating clustering methods in the context of fashion image analysis.

4.1.3 MNIST dataset

The MNIST dataset stands as a seminal benchmark in the realm of machine learning, renowned for its pivotal role in benchmarking clustering methods. Comprising a vast array of hand-written digit images, MNIST serves as a standard reference for evaluating the performance and robustness of various clustering techniques.

4.1.4 USPS dataset

The USPS dataset, derived from the United States Postal Service, represents an essential component of our experimental paradigm. Renowned for its practical relevance and real-world applicability, this dataset encompasses a diverse collection of hand-written digit images, offering insights into the performance of clustering methods in real-world scenarios.

Table 1. Attributes of Selected Benchmark Datasets

Datasets	Samples	Features	Clusters
Digits	1797	64	10
Fashion-MNIST	70000	784	10
MNIST	70000	784	10
USPS	9298	356	10

The selection of these datasets is underpinned by their diverse attributes and widespread adoption in the research community. As detailed in Table 1, the distinct characteristics of each dataset serve to enhance the comprehensiveness and validity of our experimental evaluations, thereby providing valuable insights into the performance and generalizability of our proposed methodology.

4.2 Introduction to evaluation methods

In the experimental evaluation of clustering methods, the choice of appropriate evaluation metrics is crucial to accurately assess the efficacy of the methods in producing meaningful clusters. Here, we delve into the intricacies of three fundamental evaluation metrics: Accuracy (ACC), Normalized Mutual Information (NMI), and Adjusted Rand Index (ARI).

4.2.1 Accuracy (ACC)

Accuracy, denoted as ACC, measures the proportion of correctly classified data points within a clustering result. Mathematically, ACC is computed as the ratio of the number of correctly classified data points to the total number of data points. Formally,

$$ACC = \frac{\sum_{i=1}^{n} \mathbf{1}(y_i = \hat{y}_i)}{n}$$

where y_i represents the true label of data point i, $\hat{y}_i$ denotes the assigned cluster label for data point i, and n signifies the total number of data points. The indicator function 1 yields a value of 1 if $y_i = \hat{y}_i$ and 0 otherwise.

4.2.2 Normalized Mutual Information (NMI)

NMI quantifies the similarity between two clusterings, considering both the mutual information and the entropies of the individual clusterings. It provides a normalized measure of clustering quality, taking into account the inherent structure of the data. Formally, NMI is defined as:

$$NMI = \frac{2 \cdot I(Y, \hat{Y})}{H(Y) + H(\hat{Y})}$$

where $I(Y, \hat{Y})$represents the mutual information between the true clustering $\hat{Y}$ and the predicted clustering $\hat{Y}$, and $H(Y)$ and $H(\hat{Y})$ denote the entropies of Y and $\hat{Y}$ respectively.

4.2.3 Adjusted Rand Index (ARI)

The Adjusted Rand Index, denoted as ARI, offers a robust measure of agreement between two clustering by accounting for chance grouping. It adjusts the Rand Index to provide a normalized measure that considers random clustering assignments. Formally, ARI is calculated as follows:

$$ARI = \frac{RI - Expected(RI)}{Max(RI) - Expected(RI)}$$

ISSN 2616-5775 Vol. 7, Issue 6: 20-31, DOI: 10.25236/AJCIS.2024.070604

where RI represents the Rand Index, $Expected(RI)$ is the expected value of the Rand Index, and $Max(RI)$ denotes the maximum possible value of the Rand Index.

These evaluation metrics play a pivotal role in quantifying the performance of clustering methods and are instrumental in guiding method selection and parameter tuning. In the subsequent sections, we present empirical results utilizing these metrics to assess the performance of various clustering methods on benchmark datasets, providing insights into their effectiveness and suitability for real-world applications.

4.3 Introduction to comparative methods

In order to assess the effectiveness of our proposed clustering approach, we conducted comparative analyses with several established clustering methods, each characterized by its unique attributes and methodologies. These methods are as follows:

K-means [16]: This is a well-established partition-based hard clustering method where each data point is assigned exclusively to one cluster. This method serves as a baseline due to its simplicity and widespread usage in clustering tasks.

DEC [17]: Deep Embedding Clustering (DEC) is an advanced deep clustering model that employs a Student's-t distribution for cluster assignment and notably omits the decoder component in its architecture. The experiments employ the original implementation of the DEC method as published by its creators.

IDEC [18]: An Improved version of DEC, IDEC maintains the local structure of the data, thereby enhancing the fidelity of the clustering outcomes relative to the underlying data distribution.

DCEC [19]: Deep Convolutional Embedded Clustering (DCEC) integrates a convolutional autoencoder (CAE) with a structure that preserves local data characteristics, along with a dedicated clustering layer, to refine the quality of the clustering.

GrDNFCS [20]: Utilizing a deep fuzzy clustering method, this approach reconstructs original data through an autoencoder, emphasizing the delineation between clusters and the regularization of affinity based on pseudo-labels, which aims to optimize the cohesiveness within clusters and the distinction between them.

DECCA [21]: This method leverages the Frobenius norm as a penalty term and integrates it with a deep embedding clustering framework that is based on a shrinkage autoencoder. This combination is designed to enhance clustering performance by effectively managing the compactness and separation of clusters.

DCEFC [4]: Standing as a sophisticated advancement over traditional Deep Embedding Clustering (DEC), the Deep Convolutional Embedded Fuzzy Clustering with Wasserstein Loss (DCEFC) utilizes Wasserstein distance as the loss function during the clustering phase. This choice is motivated by the Wasserstein distance's ability to account for the geometry of the data distribution, thereby facilitating more meaningful cluster assignments compared to traditional methods.

For the comparative analysis, we maintained consistent parameters across all methods as per the referenced methodologies, with the exception of setting m to a default value of 1.8. This ensured a fair and standardized comparison of the different clustering approaches. The evaluation was rigorously carried out through ten random executions of each method, and the results were averaged to derive robust conclusions regarding the efficacy of each clustering technique.

4.4 Experimental results and analysis

Table 2 illustrates the comprehensive evaluation of clustering methods, including our novel CADEFC method, alongside established methodologies, through the averaging of results from 10 independent runs. This meticulous assessment offers valuable insights into the stability and efficacy of the CADEFC method when compared with existing approaches. Particularly noteworthy is the superior clustering performance demonstrated by our CADEFC method across all metrics on the four-digit image datasets, as denoted by the bold annotations.

ISSN 2616-5775 Vol. 7, Issue 6: 20-31, DOI: 10.25236/AJCIS.2024.070604

Table 2: Average Clustering Results of CADEFC Method and Comparative Methods over 10 Runs

Dataset	Metrics	K-means	DEC	IDEC	DCEC*	GrDNFCS*	DECCA*	DCEFC	CADEFC
Digits	ACC	0.7525	0.8019	0.8182	0.8529	0.8608	0.8705	0.8818	0.8839
	NMI	0.7469	0.8257	0.8231	0.8452	0.8593	0.8672	0.8680	0.8702
	ARI	0.6687	0.7219	0.7300	0.8041	0.8149	0.8264	0.851	0.8576
Fashion-MNIST	ACC	0.5123	0.5791	0.5772	0.6332†	0.6351	0.6099	0.6370	0.6399
	NMI	0.5178	0.6275	0.6029	0.6636†	0.6609	0.6698	0.6698	0.6672
	ARI	0.3643	0.4558	0.4481	-	0.5028	-	0.5142	0.5181
MNIST	ACC	0.5324	0.8847	0.8851	0.8897	0.9145	0.9637	0.9589	0.9608
	NMI	0.4997	0.8525	0.8637	0.8849	0.9074	0.9074	0.9152	0.9258
	ARI	0.3652	0.8243	0.8382	-	0.8626	-	0.9119	0.9211
USPS	ACC	0.6681	0. 7277	0.7541	0.7900	0.7652	0.7731	0.8037	0.8089
	NMI	0.6265	0. 7368	0.7362	0.8257	0.7761	0.8053	0.8329	0.8401
	ARI	0.5463	0. 6639	0.6796	-	0.6903	-	0.7593	0.7618

The results reveal that the CADEFC method consistently outperforms its counterparts in terms of clustering accuracy (ACC), normalized mutual information (NMI), and adjusted Rand index (ARI). Specifically, on the MNIST dataset, CADEFC achieves an ACC of 0.9608, NMI of 0.9258, and ARI of 0.9211, surpassing the performance of competing methods such as K-means, DEC, IDEC, and DCEC. Similarly, on the Digits, Fashion-MNIST, and USPS datasets, CADEFC exhibits notable superiority across all metrics, highlighting its robustness and effectiveness in diverse clustering tasks.

These findings underscore the potential of the CADEFC method as a promising tool for clustering tasks, particularly in the domain of image data analysis. The consistently superior performance of CADEFC reaffirms its efficacy in accurately identifying clusters within complex datasets, thus offering valuable insights for various applications in pattern recognition, data mining, and image processing.

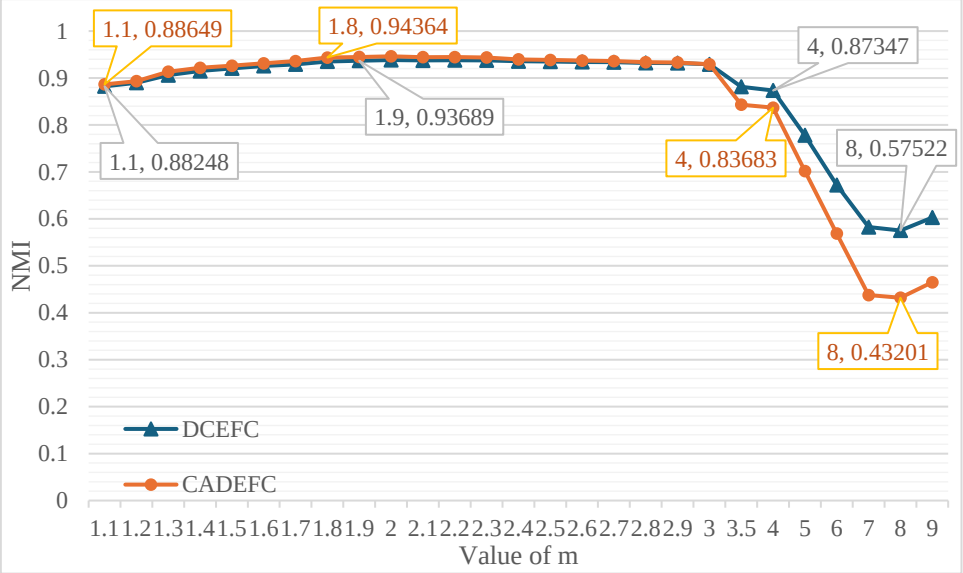

Figure 2. Normalized Mutual Information (NMI) Outcomes of CADEFC and DCEFC Methods across Various Fuzziness m Values on the MNIST Dataset.

Figure 2 illustrates the Normalized Mutual Information (NMI) outcomes of the CADEFC and DCEFC methods when applied to the MNIST dataset, with respect to varying values of the fuzziness parameter m. This graph provides a clear depiction of the clustering methods' sensitivity and adaptability under different operational conditions, as m plays a critical role in influencing the clustering results.

From the graph, it is apparent that the CADEFC method attains its peak performance with an NMI score of 0.94364 at an m value of 1.8, highlighting its robustness in accurately capturing the underlying data structure with minimal fuzziness. In comparison, the DCEFC method reaches its highest NMI of 0.93689 at an m value of 1.9, indicating a slightly higher tolerance for fuzziness but at a minor sacrifice in clustering precision.

The superior performance of the CADEFC method at lower m values not only underscores its effectiveness in discerning the intrinsic cluster structures within the dataset but also suggests a potential

ISSN 2616-5775 Vol. 7, Issue 6: 20-31, DOI: 10.25236/AJCIS.2024.070604

reduction in computational costs. This efficiency is crucial, especially in scenarios where quick and accurate data categorization is necessary. Moreover, the ability to perform optimally at lower m values may reduce the likelihood of overfitting, thus enhancing the generalizability of the clustering model.

The sensitivity of the clustering outcomes to the fuzziness parameter m is a significant observation, as it aligns with the fundamental principles of fuzzy clustering, where m determines the degree of cluster membership as fuzzy or crisp. This parameter's optimization is essential for achieving precise clustering results and can greatly influence the practical applications of fuzzy clustering methods in real-world scenarios, such as image recognition and data segmentation.

In summary, the data presented in Figure 2 advocate for the CADEFC method's deployment in applications requiring high precision and computational efficiency. The findings also open avenues for further research into the effects of the fuzziness parameter on clustering performance, with an aim to refine and optimize clustering methods for broader machine learning tasks.

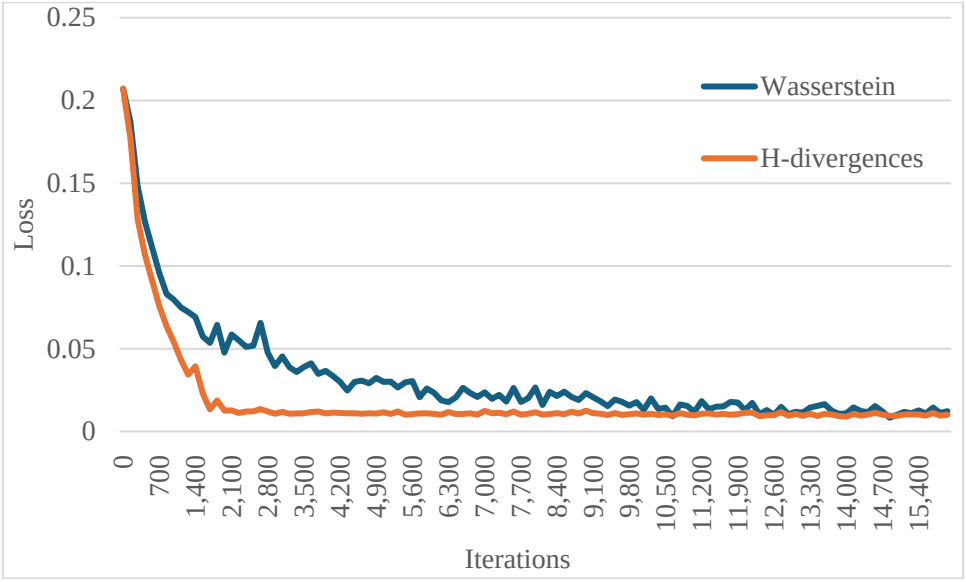

Figure3. Comparative Loss Dynamics of CADEFC and DCEFC Methods on the MNIST Dataset Using H-divergences and Wasserstein Distance Respectively as Loss Functions.

Figure 3 illustrates the comparative loss values derived from employing H-divergences and Wasserstein Distance as loss functions within the CADEFC and DCEFC methods, respectively, on the MNIST dataset. This study specifically chooses the MNIST dataset due to its widespread acknowledgment as a robust benchmark in digital image processing. The graph distinctly highlights the performance of our proposed CADEFC method, which leverages H-divergences, against the traditional DCEFC method that utilizes Wasserstein Distance as its loss function.

As depicted, the CADEFC method exhibits a notably faster reduction in loss values, effectively demonstrating the superiority of H-divergences in optimizing clustering outcomes. This rapid convergence of the CADEFC method is particularly advantageous for complex machine learning tasks, where reducing computational time without sacrificing accuracy is paramount. Furthermore, the use of H-divergences addresses some of the intrinsic weaknesses observed with the Wasserstein Distance, such as sensitivity to model architecture changes and computational inefficiency in higher dimensions.

Our analysis not only validates the effectiveness of H-divergences in enhancing the clustering method's performance but also sets a precedent for further explorations into loss functions that could potentially yield even more robust and efficient machine learning models. This comparative study underscores the pivotal role of advanced loss functions in the evolution of clustering methods, providing a clear pathway for future research in this domain.

5. Conclusion

This paper has detailed the development and evaluation of a CADEFC. Our study underscores the substantial contributions and empirical results of this innovative approach, particularly highlighting its effectiveness in clustering digital image data.

ISSN 2616-5775 Vol. 7, Issue 6: 20-31, DOI: 10.25236/AJCIS.2024.070604

Effectiveness of the Fuzzy Parameter m: Our experimental results further affirm that the inclusion of the fuzzy parameter m in the CADEFC model remains effective. The ability to adjust the fuzziness level within our clustering algorithm enables more nuanced control over membership degrees in cluster assignments, which has proven beneficial across our tests with various datasets. This flexibility enhances the model's capacity to handle overlapping data points and ambiguous classifications, contributing significantly to its robustness and accuracy in real-world scenarios.

Summary of Research Findings and the Significance of H-divergences in Deep Fuzzy Clustering: The implementation of H-divergences as a loss function within our deep clustering framework has demonstrated notable improvements in handling complex data distributions compared to traditional methods. The ability of H-divergences to effectively measure discrepancies between non-overlapping distributions has proven critical in enhancing the accuracy and robustness of cluster assignments. Our experiments across various datasets, such as Digits, Fashion-MNIST, MNIST, and USPS, have validated the superiority of this method in capturing the underlying patterns within data more precisely than previously possible.

Emphasis on the Potential Impact for Practical Applications: The methodology presented herein not only advances theoretical understanding but also offers substantial practical implications. Industries reliant on precise data categorization, such as healthcare for medical imaging, retail for customer segmentation, and autonomous driving for object recognition, could benefit immensely from the refined clustering capabilities provided by our approach. Furthermore, the adaptability of our method to different types of data and its resilience to distributional challenges make it a valuable tool for researchers and practitioners alike.

Acknowledgment of Limitations: Despite its strengths, the CADEFC method introduces a higher level of complexity, which may lead to increased computational demands. This complexity can affect the scalability and speed of the algorithm, particularly when applied to very large datasets or in real-time applications where computational resources are limited. Addressing these challenges in future iterations of the method will be crucial to enhancing its applicability and efficiency.

In conclusion, the convolutional autoencoder-based deep embedded fuzzy clustering method using H-divergences stands as a significant advancement in the field of machine learning for unsupervised data analysis. Future research will focus on further refining this approach, exploring its scalability to larger datasets, and extending its application to other challenging domains beyond image data. This work not only contributes to the academic discourse but also paves the way for enhanced data-driven decision-making in various industries.

References

[1] Lu, Jie, Guangzhi Ma, and Guangquan Zhang. "Fuzzy Machine Learning: A Comprehensive Framework and Systematic Review." IEEE Transactions on Fuzzy Systems, 2024.
[2] Wei, Xiuxi, et al. "An overview on deep clustering." Neurocomputing, 2024: 127761.
[3] Zhao, Shengjia, et al. "H-divergence: A Decision-Theoretic Probability Discrepancy Measure." IEEE Access, 2020.
[4] Chen, Tianzhen, and Wei Sun. "Deep Convolutional Embedded Fuzzy Clustering with Wasserstein Loss." International Conference on Artificial Intelligence and Big Data in Digital Era. Cham: Springer International Publishing, 2021.
[5] Alqahtani, Ali, et al. "A deep convolutional auto-encoder with embedded clustering." 2018 25th IEE E international conference on image processing (ICIP). IEEE, 2018.
[6] Du, Guowang, et al. "Neighbor-aware deep multi-view clustering via graph convolutional network." Information Fusion, 93, 2023: 330-343.
[7] Tan, Dayu, et al. "Deep adaptive fuzzy clustering for evolutionary unsupervised representation learning." IEEE Transactions on Neural Networks and Learning Systems, 2023.
[8] Bischoff, Sebastian, et al. "A Practical Guide to Statistical Distances for Evaluating Generative Models in Science." arXiv preprint arXiv:2403.12636, 2024.
[9] Cai, Jinyu, et al. "Wasserstein Embedding Learning for Deep Clustering: A Generative Approach." IEEE Transactions on Multimedia, 2024.
[10] Cai, Yuhang, and Lek-Heng Lim. "Distances between probability distributions of different dimensions." IEEE Transactions on Information Theory, 68.6, 2022: 4020-4031.
[11] Li, Yanchun, et al. "The theoretical research of generative adversarial networks: an overview." Neurocomputing, 435, 2021: 26-41.

ISSN 2616-5775 Vol. 7, Issue 6: 20-31, DOI: 10.25236/AJCIS.2024.070604

[12] Shui, Changjian, et al. "Beyond $\mathcal {H} $-Divergence: Domain Adaptation Theory With Jensen-Shannon Divergence." arXiv preprint arXiv:2007.15567, 2020.
[13] Goel, Karan, et al. "Model patching: Closing the subgroup performance gap with data augmentation." arXiv preprint arXiv:2008.06775, 2020.
[14] Rey, Andrea, et al. "Automatic Delineation of Water Bodies in SAR Images with a Novel Stochastic Distance Approach." Remote Sensing, 14.22, 2022: 5716.
[15] Zhao, Shengjia, et al. "Comparing distributions by measuring differences that affect decision making." International Conference on Learning Representations. 2022.
[16] MacQueen, James. "Some methods for classification and analysis of multivariate observations." Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. Vol. 1. No. 14. 1967.
[17] Xie, Junyuan, Ross Girshick, and Ali Farhadi. "Unsupervised deep embedding for clustering analysis." International conference on machine learning. PMLR, 2016.
[18] Guo, Xifeng, et al. "Improved deep embedded clustering with local structure preservation." IJCAI. Vol. 17. 2017.
[19] Guo, Xifeng, et al. "Deep clustering with convolutional autoencoders." Neural Information Processing: 24th International Conference, ICONIP 2017, Guangzhou, China, November 14-18, 2017, Part II 24. Springer International Publishing, 2017.
[20] Feng, Qiying, et al. "Deep fuzzy clustering—a representation learning approach." IEEE Transactions on Fuzzy Systems, 28.7, 2020: 1420-1433.
[21] Diallo, Bassoma, et al. "Deep embedding clustering based on contractive autoencoder." Neurocomputing, 433, 2021: 96-107.