

# 基于最大熵的深度模糊聚类方法研究

黄皓宇, 李少勇, 陈傲天

五邑大学智能制造学部, 广东 江门

收稿日期: 2024年3月21日; 录用日期: 2024年4月21日; 发布日期: 2024年4月29日

## 摘要

高维数据聚类是数据挖掘和模式识别研究领域的一项关键且具有挑战性的任务。深度聚类方法借助神经网络高效地特征提取能力, 往往比传统聚类方法具有更好的性能。因此, 本文提出了一种基于最大熵的深度模糊聚类算法(DFMEC)。该算法通过构建神经网络来表示模糊聚类, 具有算法模型的可解释性。联合深度自动编码器模型, DFMEC通过梯度下降实现了深度特征学习和聚类中心的同步更新, 解决了硬聚类由于其离散性而不能更新梯度的问题。此外, 在所提出方法的目标函数的优化中, 添加了基于模糊分配的最大熵正则项来提高聚类模型的鲁棒性。在各个高维数据集上的综合实验表明, 与其他先进的深度聚类方法相比, 该方法在重构表示和聚类质量方面都有更好的性能, 大量实验的深入分析证明了这一点。

## 关键词

模糊聚类, 深度聚类, 可解释性

# Research on Deep Fuzzy Clustering Based on Maximum Entropy

Haoyu Huang, Shaoyong Li, Aotian Chen

Faculty of Intelligent Manufacturing, Wuyi University, Jiangmen Guangdong

Received: Mar. 21<sup>st</sup>, 2024; accepted: Apr. 21<sup>st</sup>, 2024; published: Apr. 29<sup>th</sup>, 2024

## Abstract

Clustering of high-dimensional data is a crucial but challenging task in data mining and pattern recognition. The deep clustering usually have better performance than traditional methods due to

more efficient feature extraction capability of neural networks. Therefore, this paper proposes a deep fuzzy clustering algorithm based on maximum entropy (DFMEC). The method represents fuzzy clustering by constructing neural networks and has the interpretability of the model. Jointly the deep autoencoder model, DFMEC through the network gradient realizes deep feature extraction and updating the clustering centers simultaneously, solving the problem that hard clustering cannot update gradients due to the discrete. Moreover, in the optimization of the objective function of the proposed method, a maximum entropy regularity term based on fuzzy assignment is added to improve the robustness of the clustering model. Comprehensive experiments on various high-dimensional datasets show that, in contrast to other advanced deep clustering methods, the method has better performance in terms of reconstruction representation and clustering quality, it is demonstrated by extensive experimental analyses.

## Keywords

Fuzzy Clustering, Deep Clustering, Interpretability

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 介绍

聚类是根据数据对象的相似性进行区分和分类的过程，是数据挖掘[1]、图像处理[2]和模式识别[3]的重要研究方法。基于划分的传统聚类算法主要分为两种：硬聚类和软聚类。两者的区别在于将数据分配到类簇的方法不同。硬聚类按照严格的归属关系，将样本分配到某个类中，样本对象与类簇属于“非此即彼”的隶属关系，其中 K-Means 聚类是典型的硬聚类算法。然而，现实中部分数据对象并无严格的类属，在这些场合下依据硬划分的方法并不能真正反映数据对象与类别的实际关系。随着模糊集理论[4]的广泛应用，许多基于模糊划分的软聚类方法被提出[5] [6] [7]。模糊聚类通过建立样本类属的不确定性描述，将硬聚类的严格约束转化为连续变量问题[8]，能够比较客观地反映现实世界。

高维数据聚类是目前聚类分析具有挑战性的问题之一。大数据背景下的数据集一般具有几十维甚至更高维的特征属性，往往使得传统的聚类算法失效[9]。这是由于高维空间中数据的稀疏性、“维数灾难”效应和空空间现象(Empty space phenomenon)的影响。数据在全维空间中不可能密集，但是许多传统聚类算法对距离的定义是针对全局空间的。为此，最常见的策略是通过从原始数据中找到合适的新表示方法来缓解在高维数据上的聚类困难。针对模糊聚类，单核模糊聚类[10]和多核模糊聚类[11]使用基于内核的方法将数据表示在由一些核函数设计的特征空间中，隐式地获得更具有表现性的特性。然而，设计合适的聚类核空间是一个比较困难的问题，核方法在处理大规模数据方面也存在缺陷。通过对原始数据进行显式转换的方法也能缓解高维数据上模糊聚类的困难，例如 Zhu 等人[12]直接对数据进行非线性变换来执行模糊 C 均值聚类[5]；文献[13] [14]提出随机投影和随机特征的方法能够在低维数据上近似得到一些预定义的核，基于此，Rathore 等人[15]根据随机映射提出一种快速聚类算法，提升了高维数据聚类的性能。模糊子空间聚类[16]也是一种直接的数据变换方法，因为它为不同的簇选择了正确的特征子集。然而，这些方法虽然能够在一定程度上缓解高维数据聚类的困难，但是在大规模复杂数据上仍然难以取得显著有效的聚类结果。

为了对大规模的高维复杂数据进行聚类，将无监督聚类与深度学习相结合的深度聚类方法是一个新

的研究方向,这是由于深度学习方法能够高效地捕获高维复杂数据的内在非线性结构。Ji 等人[17]使用深度神经网络对复杂数据降维,将深度聚类框架转换为子空间聚类问题,Peng 等人[18]在此基础上基于稀疏子空间聚类[19]进行深度扩展,提出深度子空间聚类。Xie [20]等人将自动编码器与聚类相结合,提出深度嵌入聚类算法(Deep Embedded Clustering, DEC),通过预训练自动编码器,学习原始数据的低维表征,采用辅助目标分布来优化基于 Kullback-Leibler (KL)散度的聚类损失,大幅提升了高维数据聚类的聚类性能。Yang 等人[21]首次将自动编码器与 K-Means 聚类相结合,相比传统 K-Means 聚类,显著地提高了聚类性能。Guo 等人[22]在损失函数中加入重构项来改进深度嵌入聚类。Li [23]将 DEC 的预训练网络置换为 CNN,提取了包含像素空间信息的高质量特征,提升了整体聚类性能。除此之外,与深度学习相结合的聚类还有基于判别信息的软聚类[24],谱聚类[25],图聚类[26]等等。

现有的深度聚类方法大多是将深度特征学习与传统的聚类任务简单地相结合,其中特征学习与聚类任务被分为两个独立的过程。具体地说,特征学习是通过梯度下降来更新优化神经网络参数,而聚类则通过特定的目标函数来实现聚类中心的迭代更新。两个任务优化过程的相关性较小。结合模糊聚类的特性和深度自动编码器模型,我们的工作动机是研究一个可以充分利用神经网络端到端优化的深度模糊聚类模型,使得聚类任务能够随着神经网络的参数更新一起优化。因此,本文介绍了一种基于最大熵的深度模糊聚类方法(Deep fuzzy clustering based on maximum entropy, DFMEC)。该方法旨在用神经网络来表示模糊聚类,联合自动编码器网络,能够从基于潜在的表示空间、相似性度量和最大熵正则化三个方面来提升高维数据聚类的聚类性能,同时所提出的模糊聚类网络具有可解释性。本研究的主要特点如下:

(1) 解决了硬聚类由于其离散性而不能更新梯度的问题。DFMEC 利用模糊聚类中的隶属度的性质,设计了一个类隶属度指标变量来代替硬划分,使得聚类中心能够通过梯度下降随着神经网络的参数一起更新。

(2) 通过 DFMEC 定义的目标函数,能够从三个方面提升聚类性能:潜在表示空间、相似性度量和类隶属度指标的最大熵。

(3) 提出的模糊聚类网络具有可解释性。神经网络作为一个“黑盒”模型,通常无法理解其内部的工作机制。此外,现有的可解释神经网络大多都是基于监督任务设计的[27],本研究试图对无监督模糊聚类任务设计一个可解释的网络框架。

## 2. 基础知识

### 2.1. 模糊 C 均值聚类

模糊 C 均值聚类(Fuzzy C-means, FCM) [5]算法是软聚类方法的一种。作为基于目标函数的经典的聚类算法之一,FCM 根据数据点与聚类中心之间的隶属度大小关系,建立加权相似性度量函数,通过最小化目标函数来实现模糊聚类,很大程度上提高了算法的模糊性和不确定性的处理能力。

对于给定的数据集  $X = \{x_i \in R^d, i = 1, 2, \dots, N\}$ , 其中数据点  $x_i$  表示第  $i$  个样本,  $N$  和  $d$  分别代表数据样本的数量和维数。假定数据集存在  $C$  个类别,则 FCM 的目标函数为:

$$J_{FCM}(U, V) = \min \sum_{j=1}^C \sum_{i=1}^N u_{ij}^m \|x_i - v_j\|^2 \quad (1)$$

其中,  $V$  为聚类中心矩阵,  $v_j$  是第  $j$  类的聚类中心;  $U$  是隶属度矩阵,其中隶属度  $\mu_{ij}$  满足  $\sum_{j=1}^C u_{ij} = 1$  且  $u_{ij} \in [0, 1]$ , 描述数据点  $x_i$  属于第  $j$  类簇的概率,反映了数据点与各类簇之间的相似程度,隶属度值越接

近 1 代表数据点与该类簇的相似程度越高，越接近 0 代表与该类簇的相似程度越低，最终根据比较隶属度的大小来决定每个样本所属的类别； $m$  是模糊加权指数，用于调整聚类算法的模糊水平，不同的  $m$  对算法的性能(如算法收敛速度)产生不同的影响，通常情况下根据实验经验  $m$  取值为 2。对于 FCM 算法的优化，实际上是一个双目标优化问题，即约束其中一个变量，极值化另外一个变量。因此，结合隶属度函数的约束条件，利用拉格朗日乘子法得到隶属度  $\mu_{ij}$  以及聚类中心  $v_j$  的计算表达式，即：

隶属度  $\mu_{ij}$  的迭代公式为：

$$u_{ij} = \sum_{k=1}^C \left( \frac{\|x_i - v_j\|}{\|x_i - v_k\|} \right)^{\frac{-2}{m-1}} \quad (2)$$

聚类中心  $v_j$  的迭代公式为：

$$v_j = \frac{\sum_{i=1}^N u_{ij}^m x_i}{\sum_{i=1}^N u_{ij}^m} \quad (3)$$

FCM 聚类的迭代求解过程是通过不断交替更新  $U$  和  $V$  直到目标函数收敛，即根据公式(2)先计算出隶属度，再根据公式(3)计算出聚类中心，再重新计算隶属度，如此循环往复进行迭代直到达到算法终止条件并得到目标函数的解。传统 FCM 算法的时间复杂度为  $O(NC^2dt)$ ，其中  $t$  为 FCM 的收敛迭代次数。不难发现，在样本数据体量和特征维数越大时，FCM 算法优化迭代所需的计算时间开销越大，算法收敛迭代次数也不易控制。

## 2.2. 基于自动编码器的深度聚类

自动编码器(Autoencoder, AE) [28]是一种由编码器(Encoder)和解码器(Decoder)组成的无监督神经网络模型，是目前应用最广泛的网络架构之一。编码器通常由多层神经网络堆叠组成，可以学习输入数据的隐式特征；解码器的作用是将编码器学习的新特征映射回原始数据空间。自动编码器的目标是最小化重构误差，即输入数据与重构数据之间的差异。通过这种方式，自动编码器可以学习到数据的有效表示，同时也可以用于降维、特征学习、去噪等任务。自动编码器模型的重构损失函数一般可表示为：

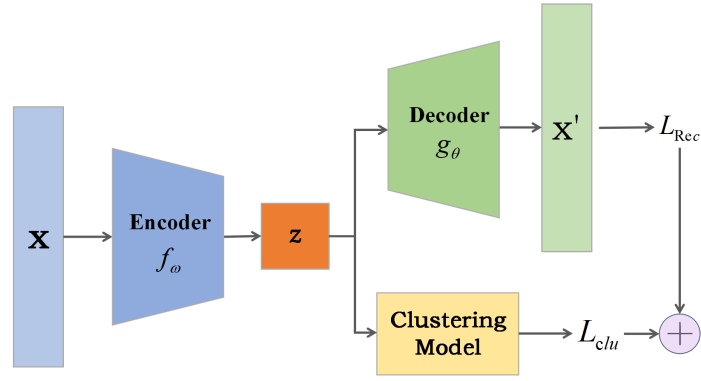
$$L_{AE} = \min \sum_{i=1}^N \|X_i - g_{\theta}(z_i)\|^2 \quad (4)$$

其中  $f_{\omega}()$  和  $g_{\theta}()$  分别代表编码器和解码器， $z_i = f_{\omega}(X_i)$  是输入数据经过编码器编码后得到的有效隐向量。训练过程中，自动编码器通过编码器学习到数据的有效表示，再通过解码器将学习到的有效表示进行重构输出。

基于自动编码器的深度聚类是一种结合自动编码器和聚类算法的方法，可以实现数据的特征学习和聚类分析的联合优化，从而得到更加准确和鲁棒的聚类结果。在构造编码器和解码器时，通常会根据特定的任务采用不同的编码层和解码层来构建特征学习网络，例如多层感知机、卷积神经网络、长短时记忆网络(LSTM) [29]、多头自注意力网络[30]等。图 1 是基于自动编码器的深度聚类模型的一般结构流程图，该类模型的损失目标函数通常是无监督表示学习损失(即特征学习网络损失  $L_{Rec}$  和聚类过程的损失  $L_{clu}$ )的线性组合，即自动编码器的深度聚类模型损失函数的一般范式可表示为：

$$\min L = \alpha L_{Rec} + \beta L_{clu}, \quad \alpha \geq 0, \beta \geq 0 \quad (5)$$

其中， $\alpha$  和  $\beta$  是平衡两个损失影响的超参数。



**Figure 1.** The structure of the deep clustering model based on the autoencoder  
**图 1.** 基于自动编码器的深度聚类模型结构

相比传统聚类算法，深度聚类算法将深度神经网络的特征学习优势与适当的聚类模型相结合，在处理规模庞大的复杂数据时具有更好的聚类性能。

### 3. 基于最大熵聚类的深度模糊聚类

#### 3.1. 模糊聚类的神经网络实现

基于 FCM 聚类算法的思想，本文尝试构建一个表示模糊聚类的神经网络框架，所建立的聚类层实现了可端到端优化的深度模糊聚类。因此，首先对模糊聚类的目标函数公式(1)进行重定义：

$$\min \sum_{j=1}^K \sum_{i=1}^N \phi_j(x_i) \cdot \|x_i - v_j\|^2 \quad (6)$$

其中  $\phi(x)$  是一个新定义的一类隶属度指标变量， $\phi_j(x_i)$  可以看作是数据点  $x_i$  对第  $j$  个集群的关注程度，且  $\phi_j(x_i) \in [0, 1]$ 。与传统模糊 C 均值聚类不同的是， $\phi(x)$  作为集群软分配的指标变量，不需要模糊加权指数  $m$  的参与，避免了模糊加权指数在对目标函数优化时造成的复杂运算。

为了实现用神经网络来表示模糊聚类，将对公式(6)进行展开：

$$\|x_i - v_j\|^2 = \|x_i\|^2 - 2v_j^T x_i + \|v_j\|^2 \quad (7)$$

对于展开式(7)，定义：

$$\|x_i\|^2 = a_i, W_j = 2v_j, \|v_j\|^2 = b_j \quad (8)$$

其中， $a_i$  和  $b_j$  分别为  $x_i$  和  $v_j$  对应的长度非负常数； $W_j$  为矩阵  $W$  的第  $j$  列，与  $v_j$  在数值上呈数量关系。

结合公式(7)和公式(8)，可以等价地将数据点  $x_i$  和聚类中心  $v_j$  之间的距离表示重构为：

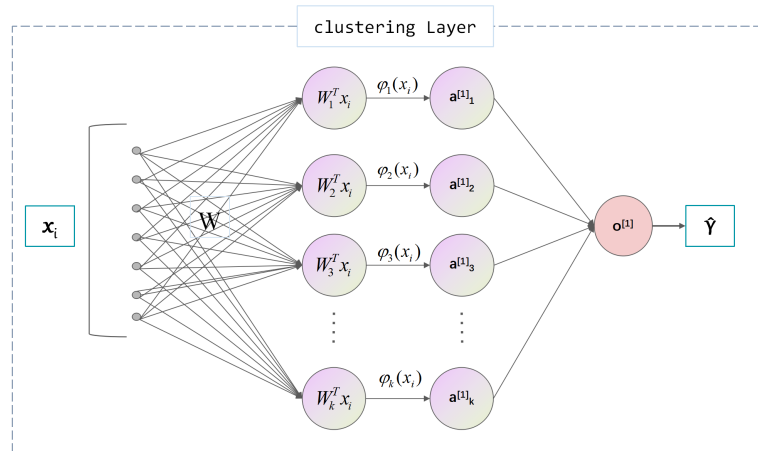
$$\|x_i - v_j\|^2 = a_i + b_j - W_j^T x_i \quad (9)$$

令：

$$y = \sum_{j=1}^K W_j^T x_i \quad (10)$$

根据全连接神经网络计算原理，将作为神经网络输入层， $W$  作为神经网络层权值矩阵，可构建一层全连接神经网络层实现对公式(10)的计算。如图 2 所示，该神经网络结构包含一层输入层，一层隐含层和激活函数以及最后一层输出层。





**Figure 2.** The structure of cluster layer's neural network  
**图 2.** 聚类层神经网络结构

给定调节因子  $\eta > 0$ , 分配变量  $\phi_j(x_i)$  可由下式得到:

$$\phi_j(x_i) = \frac{\exp\left(-\|x_i - v_j\|^2 / \eta\right)}{\sum_{k=1}^K \exp\left(-\eta \|x_i - v_k\|^2 / \eta\right)} \quad (11)$$

### 3.2. 目标函数优化分析

假设样本之间相互独立, 那么分别计算单个样本的聚类损失并不会对整个数据集的最终聚类结果产生变化和影响, 因此, 下面将根据单个样本的情况进行分析而不失一般性原则。为了方便表示, 将  $\phi_j(x)$  简写为  $\phi_j$ 。根据公式(6)和公式(9), 聚类层损失目标函数可以表示为:

$$L'_{clu} = \sum_{j=1}^K \phi_j (a + b_j - W_j^T x) \quad (12)$$

根据公式(8)的定义, 有:

$$b_j = \frac{\|W_j^T\|^2}{4} \quad (13)$$

在对聚类层网络进行训练优化时, 由于  $a$  是常量, 通过公式(13)可知, 如果直接最小化目标函数公式(12)将导致  $W_j^T$  和  $b_j$  的长度趋于无穷大, 此时聚类层的优化将难以收敛, 导致发散和不稳定的结果。为此, 本文的策略是通过归一化的方法来提高训练的稳定性。

如图 3 所示, 图中  $W_0$  表示初始的权值,  $W_1$  代表经过一次更新优化后的权值,  $G$  表示梯度。其中图 3(a)直接利用梯度  $G$  更新优化  $W_0$ , 可能导致  $W_1$  趋于无穷大而无法收敛; 图 3(b)将更新后得到的  $W_1$  归一化, 可以阻止其趋于无穷大, 但是如果梯度过大则容易导致每次更新的变化较大, 使得网络训练不稳定; 图 3(c)在每次更新后同时对权值和梯度进行归一化, 既能阻止  $W$  在训练过程趋于无穷大, 也能使得每次更新的变化幅度更加稳定。

因此, 为了防止聚类层权值更新后趋于无穷大而导致无法收敛, 以及梯度更新发生比较大的变化而导致更新时出现较大的偏差进而导致不稳定的优化。在训练过程中, 采用梯度和聚类层权值同时归一化的策略, 以保证聚类中心的适度优化, 确保稳定的收敛。具体而言, 本文策略是将数据点  $x_i$  归一化为 1 个单位长度(即  $\|x_i\|^2 = 1$ ), 聚类中心也被归一化为 1 个单位长度(即  $\|v_j\|^2 = 1$ )。由此, 公式(12)可以重写为:

$$L'_{clu} = \sum_{j=1}^K \phi_j \cdot (2 - W_j^T x) \tag{14}$$

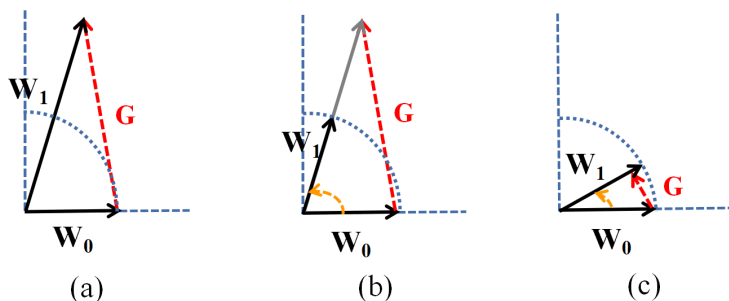


Figure 3. Updating weights and gradients with different normalization strategies  
 图 3. 网络权值和梯度的在不同归一化策略下的更新方式

### 3.3. 基于最大熵的深度模糊聚类网络可解释模型

为了约束类隶属度指标变量  $\phi(x)$  对聚类结果产生较大的影响, 本文基于最大熵理论[31], 对  $\phi(x)$  施加一个熵正则, 以此来保证对聚类中心关注程度的值的分布更加均匀, 减少模型对噪声和异常值的干扰, 增加模糊聚类的鲁棒性。因此, 基于最大熵的模糊聚类网络 DFMEC 的目标函数可定义如下:

$$\begin{aligned} \min Loss &= L_{Rec} + \lambda L_{clu} \\ &= \sum_{i=1}^N \|X_i - g(z_i)\| + \lambda \left( \sum_{j=1}^K \sum_{i=1}^N \phi_j(z_i) \cdot (2 - W_j^T z_i) + \eta \sum_{j=1}^K \sum_{i=1}^N \phi_j(z_i) \cdot \ln \phi_j(z_i) \right) \end{aligned} \tag{15}$$

其中  $\lambda$  是重构损失和聚类损失的权重占比参数,  $\eta$  是熵正则化系数且  $0 \leq \eta \leq 1$ 。整个模型框架如图 4。DFMEC 算法流程如下:

首先, 将数据集  $X$  输入到自动编码器网络结构中, 通过编码器  $f_{\omega}()$  将原始数据编码映射到低维特征空间, 学习到原始数据的隐式特征表示  $z$ , 并通过解码器  $g_{\theta}()$  将映射回原始特征空间(即重构原始输入数据), 计算自动编码器网络的重构损失  $L_{Rec}$ 。同时, 在学习到隐式特征表示  $z$  之后, 将归一化后的  $z$  输入到模糊聚类网络, 再通过类隶属度指标对  $z$  进行软分配选择合适的聚类中心, 然后对聚类网络施加最大熵正则来计算聚类损失  $L_{clu}$ , 最后联合  $L_{Rec}$  和  $L_{clu}$  训练整个模型。整个训练过程贯穿聚类模块和数据重构模块, 两个部分通过梯度下降同时进行参数更新。

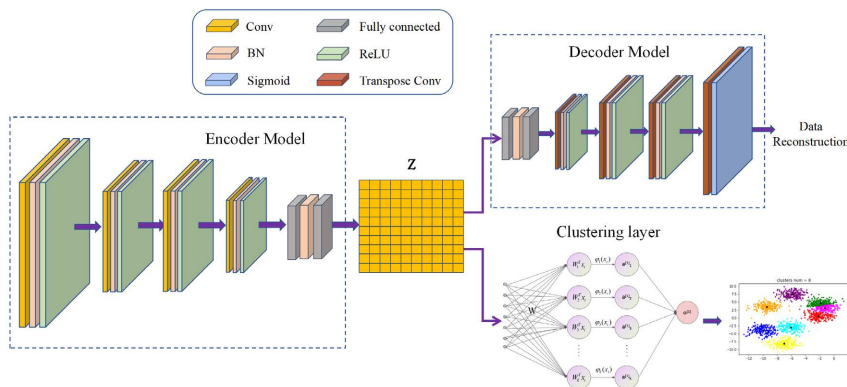


Figure 4. The flow of the DFMEC model framework  
 图 4. DFMEC 模型结构流程图

### 3.4. DFMEC 的可解释性

模型的解释性通常可分为“事前可解释性”和“事后可解释性”[32]。事前解释是在模型训练之前设计的解释方法，旨在解释模型的结构、参数或输入特征之间的关系。这种解释方法通常依赖于模型的结构和参数，例如线性模型的系数可以用来解释特征对输出的影响程度。事后解释是在模型训练之后对模型进行解释的方法，通常通过分析模型的预测结果或内部表示来揭示模型的工作原理，例如通过特征重要性分析、梯度分析、类激活图等技术来分析整个数据集或模型的行为。

然而，现存研究一直期望能够不单单用“事后解释”来对模型进行阐述。对于神经网络这个“黑盒”模型，其“可解释性”不是根植于模型本身的设计，只是通过结果来对模型进行理解性表述。本文提出的模糊聚类网络能够借助神经网络来实现模糊聚类，是一个同时具有“事前解释”与“事后解释”的可解释学习框架。下面将详细介绍 DFMEC 模型具有可解释性的特点。

所提出的模糊聚类网络以模糊聚类的目标函数为根基，具有模型结构上的可分解性。模糊聚类层网络结构的输入、权重参数、隐含层、激活函数和输出均具有可解释性。进一步说，以单个样本  $x_i$  输入为例，模糊聚类层的输入对应数据点的特征向量，包括但不限于数据本身完整特征(原始数据空间)或通过特征学习得到的高质量特征向量。在 DFMEC 中，聚类层的输入实际上是原始数据从高维空间通过编码器网络映射到低维空间中的特征向量。模糊聚类层的权值矩阵  $\mathbf{W}$  对应的是聚类中心矩阵  $\mathbf{V}$ ，由公式(8)的定义可知，聚类中心矩阵  $\mathbf{V}$  可以通过  $\mathbf{V} = \frac{1}{2}\mathbf{W}$  进行数值恢复，因此两者在数值上存在等价关系，从这个意义上看， $\mathbf{W}_j$  即代表第  $j$  类聚类中心  $v_j$ 。隐含层代表了输入数据与聚类中心之间的相似差异，具体而言，对于隐含层的神经元  $\mathbf{W}_j^T x_i$ ，从数学的角度分析，它表示了  $x_i$  与第  $j$  个聚类中心的余弦相似度，即：

$$\cos\langle \mathbf{W}_j^T, x_i \rangle = \frac{\mathbf{W}_j^T \cdot x_i}{\|\mathbf{W}_j^T\|^2 \cdot \|x_i\|^2} = \frac{1}{2} \mathbf{W}_j^T \cdot x_i = v_j^T x_i \quad (16)$$

作为激活函数的  $\varphi_j(x_i)$  在聚类中扮演的角色是对数据点进行模糊分配，与传统模糊聚类中的隶属度不同之处在于无  $\varphi_j(x_i)$  需模糊加权指数  $m$  的参与，而是通过参数因子  $\eta$  来调节分配变量的最大熵值，进而控制聚类的模糊分配程度，使模糊聚类具有更强的鲁棒性。另外，当  $\eta$  向 0 逼近时，模糊聚类将逐渐向硬聚类进行转变。模糊聚类层的输出是数据点经过模糊分配后最终的聚类结果。该模糊聚类的可解释学习框架定义的损失函数目标是 minimized 数据点与各聚类中心的相似性差异。

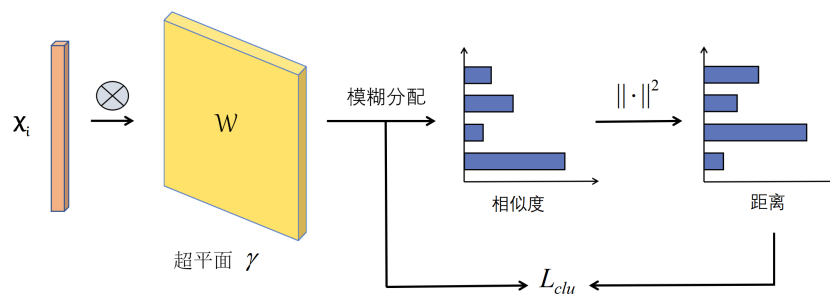


Figure 5. The flow of the fuzzy clustering layer  
图 5. 模糊聚类层流程结构

在模糊聚类网络中，如图 5 所示，权值矩阵  $\mathbf{W}$  可以看作是由所有聚类中心组成的超平面  $\gamma$ ，因此模糊聚类层的目标是学习一组尽可能满足聚类要求的超平面。从注意力机制的角度来看，超平面  $\gamma$  可以根据注意力的大小(即数据点与每个聚类中心之间的相似差异)将数据点模糊划分到相应的簇。具体地说，数



据点  $x_i$  与每个类簇之间的相似性差异将通过激活函数(在这里指的是类隶属度指标变量  $\varphi(x)$ )进行模糊划分, 再将相似度大小欧氏距离化, 依据相似度越高, 距离越小原则计算模糊聚类层的损失, 即数据点与每个聚类中心的距离之和(相似性差异之和)。另外, 在聚类时,  $\varphi(x)$ 也扮演着注意力的角色, 使得与聚类中心相似度越高的数据点受到越高的关注, 相似度低的数据点得到的关注将被抑制, 便于将数据点划分到不同的簇中。

## 4. 实验分析

### 4.1. 实验方法

为了验证 DFMEC 算法的有效性, 在 6 个具有挑战性的高维数据集上进行了实验, 分别是 MNIST、Fashion-MNIST (F-MNIST)、USPS、CIFAR-10、STL-10 和 SVHN。其中 MNIST 数据集包含 70,000 张从 0 到 9 共 10 个类别的手写数字, 每张灰度图像像素尺寸大小为  $28 \times 28$ ; F-MNIST 数据集是一个替代 MNIST 手写数字的商品图像数据集, 包含 70,000 张 10 个类别的不同商品数据; USPS 数据集由 9298 张  $16 \times 16$  像素大小的手写数字灰度图像组成; STL-10 来自 ImageNet 数据集, 包含 13,000 张 10 个类别的彩色图片, 每个类别的图像尺寸为  $96 \times 96$ ; CIFAR-10 数据集由 10 类总共 60,000 张来自 CIFAR-100 的  $32 \times 32$  大小的彩色图片构成; SVHN 街景门牌号数据集由 10 类总共 99,289 张  $32 \times 32$  彩色图片构成。数据集的具体信息可见表 1。值得一提的是, 在实验过程中, 对每个数据集均不采取任何预处理措施。另外, 针对每一个数据集, 实验将训练集和测试集合并为一个整体作为聚类任务的训练集来使用。

**Table 1.** The statistics of datasets

**表 1.** 数据集统计信息

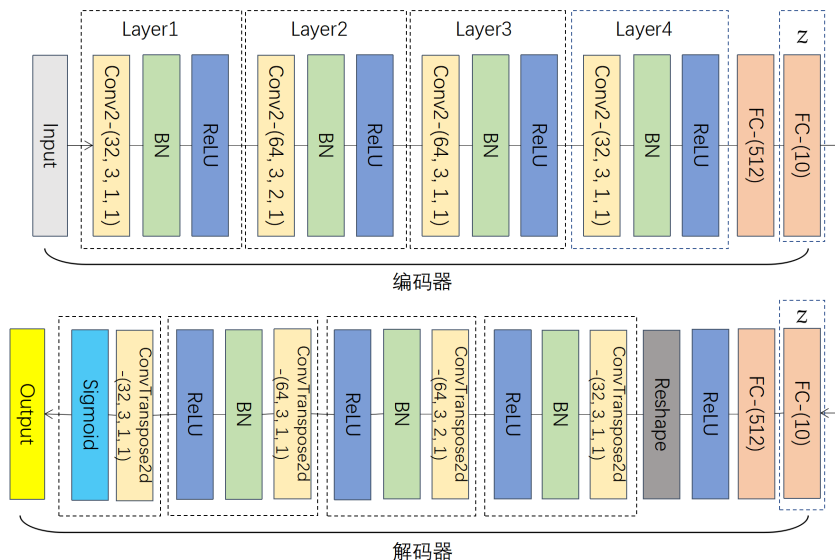
数据集	样本数量	图片大小	数据维数	类别数	通道数
MNIST	70,000	$28 \times 28$	784	10	1
F-MNIST	70,000	$28 \times 28$	784	10	1
USPS	9298	$16 \times 16$	256	10	1
CIFAR-10	60,000	$32 \times 32$	1024	10	3
STL-10	13,000	$96 \times 96$	9216	10	3
SVHN	99,289	$32 \times 32$	1024	10	3

另外, 为了在高维数据集上比较 DFMEC 算法与其他经典的深度聚类算法, 本文将重点关注 K-Means、FCM、MEC、SC 等传统的聚类算法以及 AE、VAE、GAN、DEC、DAC 等广泛应用于各种研究的先进的深度聚类算法。同时, 采用聚类精度(ACC)和归一化互信息(NMI)两个评价指标来评估聚类性能。其中, 这两个指标的值越高表示聚类性能越好。

由于在绝大多数高维空间场景中, 卷积网络比简单堆叠的全连接网络具有更强的特征学习能力。因此, 在实验中, 本文将编码器结构设置为卷积网络结构, 其中包含四个卷积层和两个全连接层, 卷积层的卷积核大小均设置为  $3 \times 3$ , 解码器结构设置为编码器网络结构的镜像。如图 6 所示, 其中 Conv2-(32, 3, 2, 1)代表的是设置卷积核通道数为 16, 卷积核大小为  $3 \times 3$ , 步长为 2 的卷积层, 填充为 1; FC-(512)表示由 256 个神经元组成的全连接层; ConvTranspose2d-(32, 3, 2, 1)代表的是反卷积层。

为了避免不同超参数对实验可行性效果测试的影响, 默认设置参数和参数, 并且在实验中采取的策略是为每个不同的聚类任务设置相同的默认参数配置。数据点和聚类中心的单位长度大小也会影响聚类结果, 这是因为图像数据的梯度值和像素值大小相差过大所致。由于图像数据的像素介于 [0,255] 之间,

如果简单地将单位长度设置为 1，那么聚类损失与重构损失的比值可能会失真。因此，在实验中，单位长度被设置为像素值的千分之一，以保持适当的损失比重。另外，梯度被归一化为单位长度的 0.2 倍，这是为了防止训练时聚类中心发生剧烈震荡。



**Figure 6.** The framework of convolutional autoencoder  
**图 6.** 卷积自动编码器结构

## 4.2. 实验结果与分析

表 2 和表 3 给出了在 6 个具有挑战性的数据集上采用不同聚类算法的 ACC 和 NMI。以粗体突出显示的数字表示最佳结果。根据表 2 和表 3 结果显示，目前基于深度特征表示的聚类方法(如 AE、VAE 和 DEC 等)的性能要优于传统的聚类方法(如 K-Means、FCM 和 SC 等)。这说明针对高维数据聚类，使用深度特征表示的方法比直接在原始数据空间中聚类更能获得显著的表现。

**Table 2.** ACC indicator results for clustering using different algorithms on six challenging datasets  
**表 2.** 在 6 个具有挑战性的数据集上使用不同的算法进行聚类的 ACC 指标结果

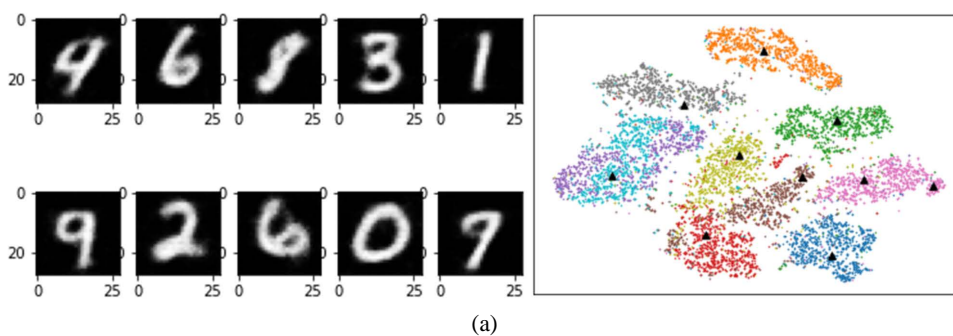
Algorithms	Dataset					
	MNIST	F-MNIST	USPS	CIFAR-10	STL-10	SVHN
K-Means [33]	0.524	0.225	0.466	0.116	0.175	-
FCM [5]	0.548	0.349	0.643	0.167	0.133	-
MEC [31]	0.426	0.203	0.420	0.102	0.087	-
SC [25]	0.479	0.359	0.659	0.138	0.158	-
AE [28]	0.812	0.494	0.716	0.161	0.253	0.109
VAE [34]	0.821	0.505	0.702	0.151	0.209	0.110
GAN [35]	0.827	0.534	0.737	0.143	0.210	0.102
DEC [20]	0.843	0.516	0.712	0.191	0.229	0.136
DFKM [21]	0.862	0.501	0.734	0.220	0.232	0.112
DAC [36]	0.946	0.544	0.795	0.237	<b>0.296</b>	0.151
<b>DFMEC</b>	<b>0.948</b>	<b>0.605</b>	<b>0.847</b>	<b>0.243</b>	0.237	<b>0.155</b>

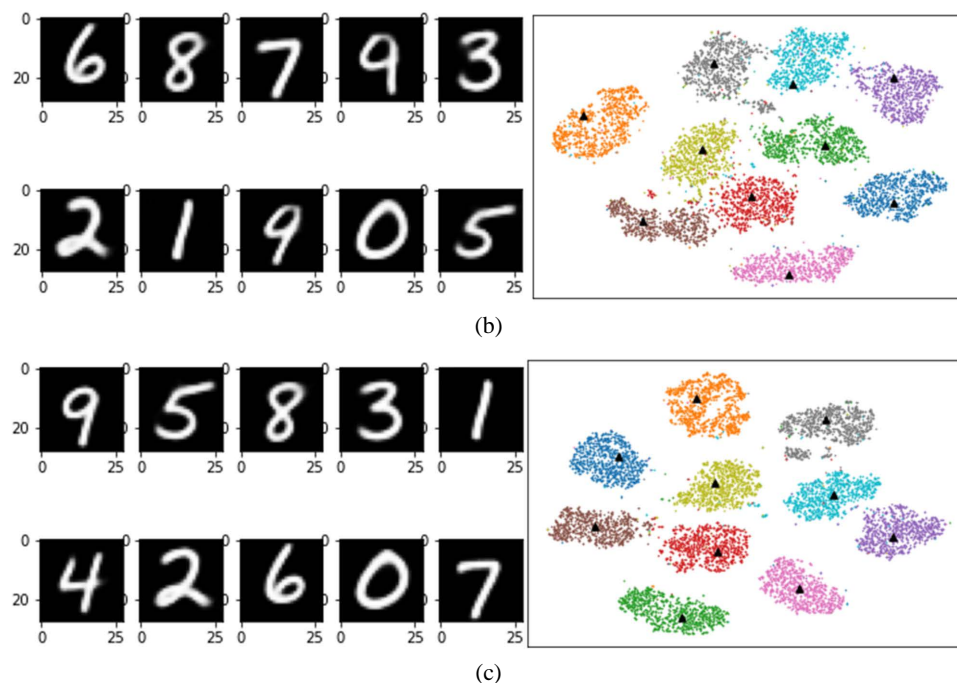
**Table 3.** NMI indicator results for clustering using different algorithms on six challenging datasets**表 3.** 在 6 个具有挑战性的数据集上使用不同的算法进行聚类的 NMI 指标结果

Algorithms	Dataset					
	MNIST	F-MNIST	USPS	CIFAR-10	STL-10	SVHN
K-Means	0.492	0.171	0.452	0.078	0.114	-
FCM	0.482	0.231	0.614	0.039	0.098	-
MEC	0.376	0.158	0.325	0.022	0.078	-
SC	0.621	0.275	0.681	0.047	0.097	-
AE	0.725	0.532	0.686	0.063	0.154	0.067
VAE	0.752	0.497	0.711	0.118	0.143	0.072
GAN	0.765	0.586	0.696	0.123	0.147	0.095
DEC	0.793	0.587	0.722	0.128	0.158	0.111
DFKM	0.780	0.572	0.718	0.086	0.113	0.091
DAC	0.854	0.513	0.753	<b>0.155</b>	<b>0.217</b>	0.113
<b>DFMEC</b>	<b>0.885</b>	<b>0.672</b>	<b>0.782</b>	0.135	0.166	<b>0.119</b>

对比所有基线算法，从 ACC 和 NMI 评价指标可看出本文提出的方法(DFMEC)在所有 6 个数据集中都取得了优越的聚类结果。

在 MNIST 数据集上，相比于其他 9 种热门的聚类算法，DFMEC 取得了最佳的 ACC 和 NMI，分别达到了 0.948 和 0.885。对比经典的深度嵌入聚类算法 DEC，DFMEC 在 ACC 上提升了约 10.5%，NMI 提升了约 9.2%。另外，DFMEC 可以不需要对自动编码器进行预训练，而是直接对整个模型进行端到端训练，这对于需要事先对自动编码器进行预训练的 DEC 算法更加方便。对于 DFKM，与本文提出的模型均属于基于模糊软分配的深度聚类方法，与其他基于硬划分以及熵分布的基线聚类算法相比，二者在两个评价指标上都有显著地提升，说明基于模糊逻辑的聚类方法具有更好的适应能力与聚类处理能力，且易于理解。与聚类性能优秀的深度自适应聚类算法 DAC 相比，DFMEC 也毫不逊色，并在 NMI 指标上提升了约 0.31%。需要指出的是，DAC 算法的 ACC 之所以高达 90% 以上是因为其在聚类过程中利用了标签信息，并且在初始化时对数据进行了滤波处理，间接提升了特征学习的能力，能够有效地提升聚类精度。而 DFMEC 在聚类时对数据不采取任何措施的前提下也依然能达到如此高的聚类准确度，说明 DFMEC 有着不弱于其他先进的深度聚类算法性能，并且在相同条件下，通过梯度下降与特征学习网络一起更新聚类中心的聚类方法相比传统的深度聚类方法(分离聚类学习和特征学习)有着更强的泛化能力和鲁棒性。

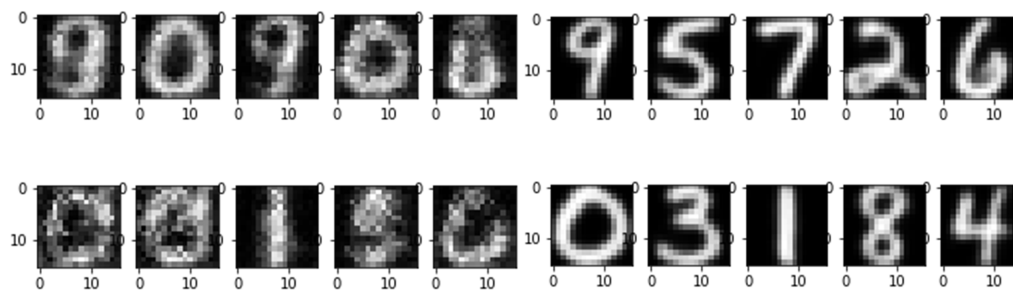




**Figure 7.** (a) The clustering center for DFMEC learning and visualization in MNIST at Epoch = 0; (b) The clustering center for DFMEC learning and visualization in MNIST at Epoch = 500; (c) The clustering center for DFMEC learning and visualization in MNIST at Epoch = 2000

**图 7.** (a) Epoch = 0 时 DFMEC 学习的 MNIST 聚类中心与可视化; (b) Epoch = 500 时 DFMEC 学习的 MNIST 聚类中心与可视化; (c) Epoch = 2000 时 DFMEC 学习的 MNIST 聚类中心与可视化

图 7 是 DFMEC 在对 MNIST 数据集进行聚类时重建的聚类中心，图 7(a)~(c)分别表示训练迭代次数为 0、500、2000 时 DFMEC 学习到的聚类中心。可观察发现，DFMEC 在训练初期也会混淆数字“4”和“9”，但随着模型训练次数增加，能够成功地将其区分，逐渐学习到了更具有代表性的聚类中心。同时，观察右图可视化结果，DFMEC 将较为离散的数据渐渐区分开来，学习到的聚类中心也能够有效地表示一个类别的中心。



**Figure 8.** The clustering center for DFMEC learning in USPS at Epoch = 0 (left) and Epoch = 2000 (right)

**图 8.** Epoch = 0 (左)和 Epoch = 2000 (右)时 DFMEC 学习的 USPS 聚类中心

在 USPS 数据集上，DFMEC 在两个评估指标上依然取得了最佳，ACC 和 NMI 分别比其他性能最好的聚类算法提升了 5.2% 和 2.9%。相比于所有对比算法在其他数据集上的聚类性能提升程度，DFMEC 在 USPS 数据集上的聚类性能提升最为显著，这是因为 USPS 数据集的数据维度大小在 6 个数据集中最小，在聚类前能够通过自动编码器学习到比较具有代表性的特征，这也侧面印证了特征学习能力越强，

DFMEC 的性能可以进一步提高。图 8 展示了当训练迭代次数为 0 时 DFMEC 学习到的 USPS 的聚类中心(左图)以及训练迭代次数为 2000 时 DFMEC 学习到的 USPS 的聚类中心(右图)。

在 CIFAR-10、STL-10 和 SVHN 三个高维 3 通道数据集上, DFMEC 相比于其他基线聚类算法有着不同程度的提升。需要指出的是, 虽然通过两个聚类评估指标发现 DAC 算法在聚类时通过借助图像标签数据在 STL-10 数据集上的聚类表现比 DFMEC 略高, 但这是 DFMEC 在默认参数设置下的结果。此外, 还需特别指出的是, 所提出的模糊聚类网络具有可解释性, 因此 DFMEC 还具有较强的可解释学习能力和全过程端到端优化的优势, 这是其他算法模型所不具备的。

## 5. 总结

本文提出了一种基于最大熵的深度模糊聚类方法(DFMEC), 通过重定义模糊聚类的目标函数, 构建了一个模糊聚类网络, 并联合了深度自动编码器实现了基于最大熵的深度模糊聚类。该方法集成了模糊逻辑、自动编码器模型和最大熵原理, 能够实现有效地高维数据聚类, 同时具有算法模型设计上的可解释性。最后, 通过在多个高维数据集上的实验验证了该方法。结果表明, DFMEC 在聚类性能方面优于其他应用广泛的传统聚类和深度聚类算法。与硬聚类方法相比, DFMEC 采用模糊聚类中软分配的思想代替离散的硬分配, 可以更容易地接近实际情况。通过神经网络中的梯度下降法来动态更新聚类中心, 使算法易于收敛, 并且具有全过程端到端优化的特点。在对比实验中, 针对 3 通道高维图片数据, 聚类的总体性能虽然比其他基线算法更优秀, 但是还仍然有很大的上升空间, 这可能是由于默认设置的特征学习网络难以获取彩色图像数据的高质量特征导致的。对于未来的工作, 我们将关注如何实现一个超参数更少以及在多个大规模高维数据集上同时具有聚类准确性和稳定性的可解释聚类模型。

## 参考文献

- [1] Cios, K.J., Pedrycz, W. and Swiniarski, R.W. (2012) *Data Mining Methods for Knowledge Discovery*. Springer Science & Business Media, New York.
- [2] Boopathi, S., Pandey, B.K. and Pandey, D. (2023) Advances in Artificial Intelligence for Image Processing: Techniques, Applications, and Optimization. In: Pandey, B.K., Pandey, D., Anand, R., Mane, D.S. and Nassa, V.K., Eds., *Handbook of Research on Thrust Technologies' Effect on Image Processing*, IGI Global, Hershey, 73-95. <https://doi.org/10.4018/978-1-6684-8618-4.ch006>
- [3] Paolanti, M. and Frontoni, E. (2020) Multidisciplinary Pattern Recognition Applications: A Review. *Computer Science Review*, **37**, Article ID: 100276. <https://doi.org/10.1016/j.cosrev.2020.100276>
- [4] Zadeh, L.A. (1965) Fuzzy Sets. *Information and Control*, **8**, 338-353. [https://doi.org/10.1016/S0019-9958\(65\)90241-X](https://doi.org/10.1016/S0019-9958(65)90241-X)
- [5] Bezdek, J.C., Ehrlich, R. and Full, W. (1984) FCM: The Fuzzy C-Means Clustering Algorithm. *Computers & Geosciences*, **10**, 191-203. [https://doi.org/10.1016/0098-3004\(84\)90020-7](https://doi.org/10.1016/0098-3004(84)90020-7)
- [6] Pimentel, B.A., Silva, R. and Costa, J. (2022) Fuzzy C-Means Clustering Algorithms with Weighted Membership and Distance. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, **30**, 567-594. <https://doi.org/10.1142/S0218488522500143>
- [7] Hashemi, S.E., Gholian-Jouybari, F. and Hajiaghahi-Keshteli, M. (2023) A Fuzzy C-Means Algorithm for Optimizing Data Clustering. *Expert Systems with Applications*, **227**, Article ID: 120377. <https://doi.org/10.1016/j.eswa.2023.120377>
- [8] Zadeh, L.A. (1971) Similarity Relations and Fuzzy Orderings. *Information Sciences*, **3**, 177-200. [https://doi.org/10.1016/S0020-0255\(71\)80005-1](https://doi.org/10.1016/S0020-0255(71)80005-1)
- [9] Kriegel, H.P., Kröger, P. and Zimek, A. (2009) Clustering High-Dimensional Data: A Survey on Subspace Clustering, Pattern-Based Clustering, and Correlation Clustering. *ACM Transactions on Knowledge Discovery from Data*, **3**, 1-58. <https://doi.org/10.1145/1497577.1497578>
- [10] Zeng, S., Wang, X., Duan, X., et al. (2020) Kernelized Mahalanobis Distance for Fuzzy Clustering. *IEEE Transactions on Fuzzy Systems*, **29**, 3103-3117. <https://doi.org/10.1109/TFUZZ.2020.3012765>
- [11] Tan, D., Zhong, W., Jiang, C., et al. (2020) High-Order Fuzzy Clustering Algorithm Based on Multikernel Mean Shift. *Neurocomputing*, **385**, 63-79. <https://doi.org/10.1016/j.neucom.2019.12.030>



- [12] Zhu, X., Pedrycz, W. and Li, Z. (2017) Fuzzy Clustering with Nonlinearly Transformed Data. *Applied Soft Computing*, **61**, 364-376. <https://doi.org/10.1016/j.asoc.2017.07.026>
- [13] Rathore, P., Bezdek, J.C., Erfani, S.M., et al. (2018) Ensemble Fuzzy Clustering Using Cumulative Aggregation on Random Projections. *IEEE Transactions on Fuzzy Systems*, **26**, 1510-1524. <https://doi.org/10.1109/TFUZZ.2017.2729501>
- [14] Long, C. and Kong, L. (2017) Fuzzy Clustering in High-Dimensional Approximated Feature Space. *International Conference on Fuzzy Theory & Its Applications*, Taichung, 9-11 November 2016, 1-6.
- [15] Rathore, P., et al. (2018) A Rapid Hybrid Clustering Algorithm for Large Volumes of High Dimensional Data. *IEEE Transactions on Knowledge and Data Engineering*, **31**, 641-654. <https://doi.org/10.1109/TKDE.2018.2842191>
- [16] Wen, Z., Hou, B., Wu, Q., et al. (2018) Discriminative Transformation Learning for Fuzzy Sparse Subspace Clustering. *IEEE Transactions on Cybernetics*, **48**, 2218-2231.
- [17] Ji, P., Zhang, T., Li, H., et al. (2017) Deep Subspace Clustering Networks. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, 4-9 December 2017, 23-32.
- [18] Peng, X., Feng, J., Zhou, J.T., et al. (2020) Deep Subspace Clustering. *IEEE Transactions on Neural Networks and Learning Systems*, **31**, 5509-5521. <https://doi.org/10.1109/TNNLS.2020.2968848>
- [19] Elhamifar, E. and Vidal, R. (2013) Sparse Subspace Clustering: Algorithm, Theory, and Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **35**, 2765-2781. <https://doi.org/10.1109/TPAMI.2013.57>
- [20] Xie, J., Girshick, R. and Farhadi, A. (2016) Unsupervised Deep Embedding for Clustering Analysis. *International Conference on Machine Learning*, **48**, 478-487.
- [21] Yang, B., Fu, X., Sidiropoulos, N.D., et al. (2017) Towards K-Means-Friendly Spaces: Simultaneous Deep Learning and Clustering. *International Conference on Machine Learning*, **70**, 3861-3870.
- [22] Guo, X., Gao, L., Liu, X., et al. (2017) Improved Deep Embedded Clustering with Local Structure Preservation. *International Conference on Artificial Intelligence*, **17**, 1753-1759. <https://doi.org/10.24963/ijcai.2017/243>
- [23] Li, F., Qiao, H. and Zhang, B. (2018) Discriminatively Boosted Image Clustering with Fully Convolutional Auto-Encoders. *Pattern Recognition*, **83**, 161-173. <https://doi.org/10.1016/j.patcog.2018.05.019>
- [24] Wang, Q., Cheng, J., Gao, Q., et al. (2020) Deep Multi-View Subspace Clustering with Unified and Discriminative Learning. *IEEE Transactions on Multimedia*, **23**, 3483-3493. <https://doi.org/10.1109/TMM.2020.3025666>
- [25] Duan, L., Ma, S., Aggarwal, C. and Sathe, S. (2021) Improving Spectral Clustering with Deep Embedding, Cluster Estimation and Metric Learning. *Knowledge and Information Systems*, **63**, 675-694. <https://doi.org/10.1007/s10115-020-01530-8>
- [26] Li, W., Wang, S., Guo, X., et al. (2023) Deep Graph Clustering with Multi-Level Subspace Fusion. *Pattern Recognition*, **134**, Article ID: 109077. <https://doi.org/10.1016/j.patcog.2022.109077>
- [27] 曾春艳, 严康, 王志锋, 等. 深度学习模型可解释性研究综述[J]. *计算机工程与应用*, 2021, 57(8): 1-9.
- [28] Hinton, G.E. and Salakhutdinov, R.R. (2006) Reducing the Dimensionality of Data with Neural Networks. *Science*, **313**, 504-507. <https://doi.org/10.1126/science.1127647>
- [29] Sagheer, A. and Kotb, M. (2019) Unsupervised Pre-Training of a Deep LSTM-Based Stacked Autoencoder for Multivariate Time Series Forecasting Problems. *Scientific Reports*, **9**, Article No. 19038. <https://doi.org/10.1038/s41598-019-55320-6>
- [30] Dang, H.V. and Ngo, D. (2019) Attentional Autoencoder for Weighted Implicit Collaborative Filtering. *Proceedings of the 2019 2nd International Conference on Computational Intelligence and Intelligent Systems*, Bangkok, 23-25 November 2019, 168-172. <https://doi.org/10.1145/3372422.3372423>
- [31] Karayiannis, N.B. (1994) MECA: Maximum Entropy Clustering Algorithm. *Proceedings of 1994 IEEE 3rd International Fuzzy Systems Conference*, Orlando, 26-29 June 1994, 630-635.
- [32] Li, X., Xiong, H., Li, X., et al. (2022) Interpretable Deep Learning: Interpretation, Interpretability, Trustworthiness, and Beyond. *Knowledge and Information Systems*, **64**, 3197-3234. <https://doi.org/10.1007/s10115-022-01756-8>
- [33] MacQueen, J. (1967) Some Methods for Classification and Analysis of Multivariate Observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, **1**, 281-297.
- [34] Lopez, R., Regier, J., Jordan, M.I., et al. (2018) Information Constraints on Auto-Encoding Variational Bayes. *32nd Conference on Neural Information Processing Systems (NeurIPS 2018)*, Montréal, 3-8 December 2018.
- [35] Radford, A., Metz, L. and Chintala, S. (2015) Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv: 1511.06434.
- [36] Yu, S., Liu, J., Han, Z., et al. (2021) Representation Learning Based on Autoencoder and Deep Adaptive Clustering for Image Clustering. *Mathematical Problems in Engineering*, **2021**, Article ID: 3742536. <https://doi.org/10.1155/2021/3742536>