DOI: 10.19650/j.cnki.cjsi.J2413072

双级联合投影包络内嵌堆栈自动编码器*

李勇明,朱立志,王 品,马 洁,周传艳
(重庆大学微电子与通信工程学院 重庆 400044)

摘 要:深度堆栈自动编码器作为一种代表性的深度网络,已被广泛应用在数据科学、模式识别等领域。现有的深度堆栈自动编码器均针对原样本个体进行深度特征变换,忽略了样本之间的关联结构信息,导致其深度特征的质量往往不尽如人意。为了解决这一问题,提出一种新的深度堆栈自动编码器网络-双级联合投影包络内嵌堆栈自动编码器。与现有的深度堆栈自动编码器本质上不同的是,双级联合投影包络内嵌堆栈自动编码器针对样本间关联信息而非样本个体本身进行深度特征变换。该模型主要包括两部分:双级联合投影包络模块和内嵌式堆栈自动编码器。在双级联合投影包络模块中,流形样本对包络子模块用于提取原样本间局部关联信息,重构生成第 1 层包络样本;保持降维式聚类子模块用于提取样本的全局关联信息,重构生成第 2 层包络样本。双级间一致性保持模块用于优化第 2 层包络样本的表征能力。然后,在这 2 层包络样本上分别训练 2 个内嵌式堆栈自动编码器,获得 2 组深度特征。组织了 4 组实验,包括消融实验、算法比较、参数影响分析以及复杂度分析。实验结果表明,双级联合投影包络内嵌堆栈自动编码器提取的深度特征具有较高且稳定的质量。

关键词: 内嵌堆栈自动编码器;包络学习;双级;包络样本;聚类;域适应
中图分类号: TP181 TH77 **文献标识码**: A **国家标准学科分类代码**: 510.4

Dual-stage joint projection envelope embedded stack autoencoder

Li Yongming, Zhu Lizhi, Wang Pin, Ma Jie, Zhou Chuanyan
(*School of Microelectronics and Communication Engineering, Chongqing University, Chongqing 400044, China*)

Abstract: The deep stacked autoencoder, as a prominent deep learning architecture, has been widely applied in fields such as data science and pattern recognition. However, existing deep stacked autoencoders focus on transforming the features of individual samples, often overlooking the inter-sample correlation, which can lead to suboptimal feature quality. To address this limitation, this paper introduces a novel deep stacked autoencoder architecture called the dual-stage joint projection envelope embedded stacked autoencoder. Unlike traditional deep stacked autoencoders, the proposed model transforms deep features based on the correlation information between samples, rather than focusing solely on the samples themselves. The model is composed of two primary components: the dual-stage joint projection envelope and the embedded stacked autoencoder. The dual-stage joint projection envelope utilizes a manifold sample-pair envelope module to extract local correlation information from the original samples and reconstruct the first layer of enveloped samples. A descending clustering module is then employed to capture global correlations and reconstruct the second layer of enveloped samples. Additionally, the dual-stage inter-consistency maintenance module enhances the representational power of the second-layer enveloped samples. Subsequently, two sets of deep features are extracted by training two embedded stacked autoencoders on these two layers of enveloped samples. The paper concludes with four sets of experiments: ablation studies, algorithm comparisons, parameter sensitivity analysis, and complexity analysis. Experimental results demonstrate that the deep features extracted by the proposed dual-stage joint projection envelope embedded stacked autoencoder exhibit both high quality and stability.

Keywords: embedded stack autoencoder; envelope learning; dual-stage; envelope samples; clustering; domain adaptation

收稿日期:2024-07-18 Received Date: 2024-07-18
*基金项目:国家自然科学基金项目(61771080)、国家自然科学基金重点项目(U21A20448)资助

0 引　　言

特征学习在传统机器学习和深度学习领域具有重要意义[1]。相比传统特征学习方法,深度学习模型其强大的特征学习能力使得其能够自动学习和发现数据中的复杂非线性关系,对于高维、非线性数据表现更为出色,在各种复杂任务中有显著的表现,已成为许多领域的关键技术。其中,深度堆栈自动编码器(deep stacked auto-encoder, SAE)作为一种有代表性的深度特征学习方法,能够通过多层非线性转换学习数据的高阶表示,从而捕获数据的复杂结构和规律,易于理解和实现,已被广泛应用在计算机视觉、自然语言处理、推荐系统等领域[2-6]。

根据模式识别相关研究表明,样本间存在一定的关联性(关联信息)且常常有助于分类识别。例如,人们识别草地和人脸,往往基于像素之间的关联性。基于同一个受试者的各项检测样本之间存在明显关联性,被联合用于疾病诊断等。然而,现有的 SAE 模型的基本原理是利用重构的深度特征最小化每个输入样本与输出样本之间的误差,忽略了样本之间的关联信息。假设第 i 个样本和第 j 个样本之间存在关联性(关联信息)且有助于分类,而现有 SAE 只会逐个或逐批输入训练样本,从而导致 SAE 训练过程未考虑某些样本间的关联信息,降低了训练质量,从而影响特征的分类能力。虽然样本间关联信息隐藏在原样本中,但不可能将所有样本一次性送入 SAE 用于特征学习。即便能将所用样本一次性送入 SAE 进行学习,但海量的样本组合与有限的样本数之间存在巨大矛盾。例如,输入样本特征的个数为 d,特征的组合为 $C1(d)$,样本的个数为 n,样本的组合记录为 $C2(n)$,则两者的组合个数为 $C1(d) \times C2(n)$。可见,在维度一定的情况下,输入样本数较大时,待搜索的组合数出现爆炸性增长,将会显著加大过拟合风险,降低深度特征的质量。上述这一问题是目前 SAE 存在的共性问题,至今还未得以解决。因此,有必要研究如何挖掘样本间关联信息并用于 SAE 训练的预处理方法,从而改进现有 SAE 的特征学习效率。

样本间关联信息可分为局部和全局关联信息。基于此,拟研究流形近邻样本对来挖掘样本间局部关联信息,研究均值聚类来挖掘样本间全局关联信息,将两种信息转化为包络样本用于后续 SAE 学习,从而构建一种新的 SAE 模型-双级联合投影包络内嵌堆栈自动编码器(dual-stage joint projection envelope embedded stack autoencoder,DSJPE-ESAE)。该模型包括两大部分:双级联合投影包络模块(dual-stage joint projection envelope, DSJPE)和内嵌式堆栈自动编码器(embeded stacked autoencoder, ESAE)。首先设计流形样本对包络模块(manifold sample-pair envelope module,MSPE),挖掘样本间局部关联结构信息,重构生成第 1 层包络样本。其次设计保持降维式聚类模块(maintaining descending clustering module,MDC),通过聚类降维联合优化,从而确保在适宜维度下最优挖掘到样本间全局关联结构信息,重构生成第 2 层包络样本。之后,设计了双级间一致性保持模块(dual-stage inter consistency maintenance module,DSICM),优化第 2 层包络样本。最后设计内嵌式堆栈自动编码器,并分别在 2 层包络样本集上进行训练,可输出 2 组深度特征。由后续的 MSPE 介绍可知,当样本间不存在局部关联性时,包络样本则退化为原样本,DSJPE-ESAE 则退化为 ESAE。因此,传统 SAE 可以被看成本文模型的特例,本文模型适用范围更广。

本研究主要创新点和贡献为:

1)现有的 SAE 是针对原样本个体提取深度特征,没有考虑样本间的关联信息。与此不同,DSJPE-ESAE 是考虑了样本间关联信息,并在此基础上提取了深度特征,从而实现样本与特征的协同(并行)深度变换,实现了新的 SAE 特征学习范式。

2)现有的 SAE 是基于原样本个体的单级深度结构,是一种单输入-单输出的堆栈自动编码器,而 DSJPE-ESAE 是针对样本个体及其关联信息的双级深度结构,得到 2 种样本并联接入 ESAE,从而构建了一种双输入-双输出的双级联合投影包络内嵌堆栈自动编码器,可学习到 2 种深度特征。

3)与现有的 SAE 不同,DSJPE-ESAE 针对样本间局部关联信息挖掘,设计了流形样本对包络模块 MSPE,有效挖掘样本间的局部关联结构信息,重构生成第 1 层包络样本,从而实现对样本间局部关联信息的深度特征变换。

4)与现有的 SAE 不同,DSJPE-ESAE 针对样本间全局关联信息挖掘,设计了降维式聚类模块,将聚类与流形降维同时进行,降低陷入局部极值风险,提高了第 2 层包络样本的生成质量,从而实现了对样本间全局关联信息的深度特征变换。

5)与现有的 SAE 不同,DSJPE-ESAE 设计了双级间一致性保持机制,保障了变化后包络样本对原样本的表征能力,进一步提升第 2 层包络样本的生成质量,并通过将双级间一致性保持问题转化为域适应问题加以解决。

SAE 是一种由多层自动编码器(autoencoder, AE)组成的反馈式神经网络模型[7],以非监督的训练方式实现数据降维与分类,是目前深度学习领域最重要的研究热点之一[8]。AE 是一种无监督的单隐层神经网络,其输出层设置为等于输入层,目标是在输出层尽可能精准重构原始输入[9]。AE 就是要最小化隐层对输入数据的重

构误差,这样便可使得隐层数据成为输入数据的一种特征表示[10]。自编码器是一种深度学习模型,它的目标是学习一个 能够有效编码和解码输入数据的表示[11]。典型的 SAE 有堆栈去噪自动编码器(stacked denoising autoencoder,SDAE)、堆栈稀疏自动编码器(stacked sparse autoencoder,SSAE)、堆栈变分自动编码器(stacked variational autoencoder,SVAE)和堆栈卷积自动编码器(stack convolution autoencoder,SCAE)等。Görgel 等[12]在研究中采用了堆栈去噪稀疏自动编码器(stacked denoising sparse autoencoder,SDSAE)来进行人脸识别;Zhu 等[13]提出了一种新的堆栈剪枝稀疏自动编码器(stacked pruned sparse autoencoder,SPSAE)模型,包含一个全连接的自动编码器,利用每一层提取的优势特征参与到后续的层中,从而减少了样本信息的丢失;Raulf 等[14]研究了从堆栈收缩自动编码器中学习到的特征的鲁棒性。

目前的 SAE 改进研究主要在网络结构、正则化方法、权重、激活函数和损失函数等方面。Shao 等[15]提出了一种基于自适应 Morlet 小波作为激活函数的 SAE 改进方法。与传统的激活函数相比,自适应 Morlet 小波激活函数能更好地提取信号特征;Nouri 等[16]在 SAE 的训练过程中生成吸引子,提出了一种新的损失函数,通过减少特征值的绝对实数来保持重构误差;Yang 等[17]采用了一种改进的带有稀疏项和权值惩罚项的损失函数作为 SSAE 模型的损失函数,有效提高了模型的预测精度和泛化能力;Yang 等[18]提出双层堆栈自动编码器(double stacked autoencoder,DSAE)模型,该模型包括 2 个堆栈的自动编码器层,用于特征提取、降维和转换,为判别特征进行分类;Sun 等[19]提出门控堆栈目标相关自编码器(gated stacked target autoencoder,GSTAE),在 SAE 预训练时将目标预测损失项纳入其损失函数;Ou 等[20]提出了一种新的质量驱动正则化堆栈自动编码器(quality-driven regularization stacked autoencoder,QRSAE),该模型使用质量相关约束来调节不同强度的权重矩阵。El-Fiqi 等[21]提出加权门自编码器(weighted gate layer autoencoder,WGLAE),它可以同时学习变量之间的函数映射,在缺失变量的数据重建方面优于经典 SAE,在缺失数据恢复方面具有潜在的应用前景;Xiao 等[22]提出堆栈图自动编码器(stacked graph autoencoder,SGAE)使用消息传递机制从邻近收集信息,并在集群应用中邻接矩阵不可用时构建信息表示;Dai 等[23]利用 SDAE 重构缺失数据,在自编码器损失函数中引入了权重平衡、方差限制、缺失数据惩罚等多种约束条件,解决了缺失数据重构问题,并将移动边缘计算技术应用于模型训练和测试。

基于以上文献分析可以看出,现有 SAE 的改进研究取得了显著的正面效果,但主要局限在训练、网络结构和参数、损失函数等方面,没有考虑对输入样本的预处理,没有挖掘样本间的关联结构信息,特征的重建误差计算仍然只针对原样本个体。

1 双级联合投影包络内嵌堆栈自动编码器

本研究提出的 DSJPE-ESAE 的结构如图 1 所示。其主要包括了两部分:双级联合投影包络模块 DSJPE 和内嵌式堆栈自动编码器 ESAE。

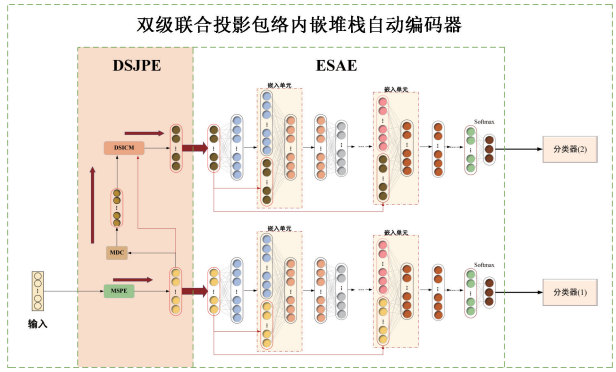


图 1 DSJPE-ESAE 原理

Fig. 1 Schematic of DSJPE-ESAE

在 DSJPE 中,MSPE 子模块用于提取原样本间局部关联信息,重构生成第 1 层包络样本;MDC 子模块用于提取样本的全局关联信息,重构生成第 2 层包络样本。DSICM 用于优化第 2 层包络样本的表征能力。然后,分别在这 2 层包络样本上训练 2 个 ESAE,得到 2 个输出(输出 2 组深度特征)。

1.1 流形样本对包络模块

简单的欧式距离无法反应复杂近邻样本间的真实距离,因此这里采用测地距离(geodesic distance,GD)来度量非线性流形上样本之间的相似性。测地距离最小的样本被认为是最近邻样本。MSPE 过程如图 2 所示,将原始样本与其各自的 v 个流形最近邻样本拼接,生成流形最近样本对。

给定一个多维样本向量 $\boldsymbol{X} \in \boldsymbol{R}^{n\times d}$,其中包含 n 个样本 $\{x_1,\cdots,x_i,\cdots,x_n\}$,MSPE 的流程为:

首先,根据式(1)计算原始样本 x_i 与所有其他样本 x_j 之间的最短流形距离。

$$S_g(x_i,x_j)=\min_{\ell\in\rho_{i,j}}\sum_{k=1}^{l-1}\Lambda(\ell_k,\ell_{k+1}) \tag{1}$$

其中,$S_g(x_i,x_j)$ 表示 x_i 和 x_j 之间的流形距离。将数据点作为图 $g=(\Delta,I)$ 的顶点,设 $\ell=\{\ell_1,\ell_2,\cdots,\ell_l\}\in\Delta^i$ 表示图上连接点 ℓ_i 和 ℓ_j 的路径,其中边 $\{\ell_k,\ell_{k+1}\}\in I$,$1\leqslant k<l-1$。$\ell_{i,j}$ 表示连接数据 x_i 和 x_j 的所有路径的集

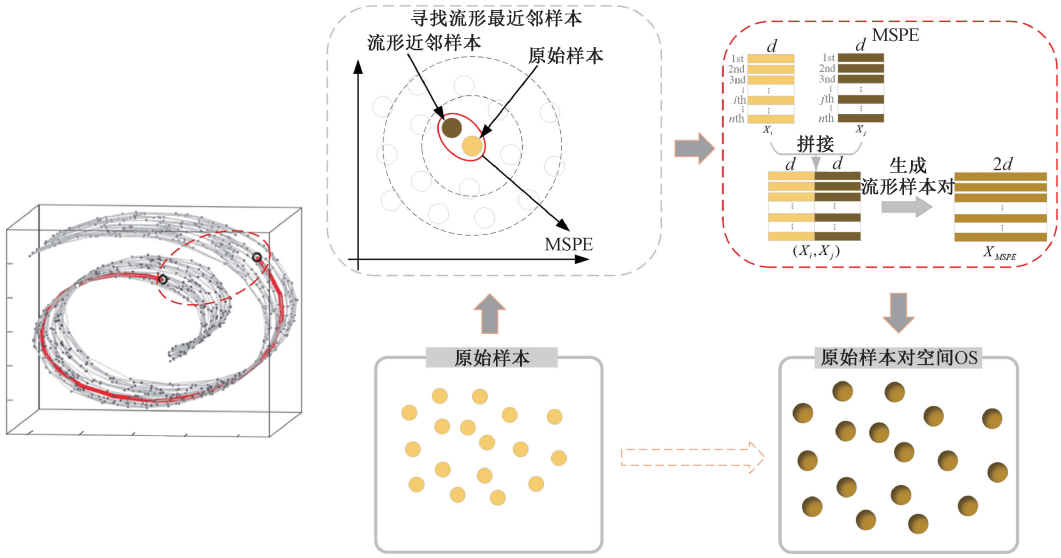


图 2　MSPE 示意图

Fig. 2　Schematic diagram of MSPE

合。$\Lambda(\ell_k,\ell_{k+1})$ 是流形上的线段的长度是两点之间在空间 ℓ_k 和,其中 $\Lambda(\ell_k,\ell_{k+1}) = \rho^{dist(\ell_k,\ell_{k+1})} - 1$。$dist(\ell_k,\ell_{k+1})$ 是两点之间的欧氏距离,$\rho(\rho > 1)$ 是拉伸因素。

其次得到流形距离矩阵 $\boldsymbol{S} = [S_g(x_i,x_j)]_{n\times n}$,在矩阵 $\boldsymbol{S}$ 的每一行中选择距离值最小的 v 个样本作为流形最近邻样本 $x_{nearest} = \{x^1_{nearest}, x^2_{nearest}, \cdots, x^v_{nearest}\}$。

最后,将原始样本 $x_{original}$ 与最近的流形样本 $x_{nearest}$ 拼接成流形样本对 $\boldsymbol{X}_{MSPE} = [x_{original}, x_{nearest}]$。MSPE 伪代码见算法 1。

算法 1:MSPE 伪代码算法

输入:原始样本集 $\boldsymbol{X} \in \boldsymbol{R}^{n\times d}$,流形近邻样本拼接数 v

输出:流形近邻样本对集 $\boldsymbol{X}_{MSPE} \in \boldsymbol{R}^{n\times 2d}$

1:选择原始样本 $x_{original}$

2:计算最短的流形距离 $S_g(x_i,x_j)$

3:获得流形距离矩阵 $\boldsymbol{S}$

4:根据拼接数 v 获得流形近邻样本 $x_{nearest}$

5:拼接 $\boldsymbol{X}_{MSPE} = [x_{original}, x_{nearest}]$

1.2　保持降维式聚类模块

均值聚类挖掘样本间的全局关联信息,聚类中心作为生成的包络样本。为了避免高维稀疏空间中聚类容易陷入局部极值的问题,本节设计了 MDC 模块,将聚类和降维联合进行,从而将以往处理高维数据的聚类研究的 2 个独立阶段整合到一个统一的模型中,避免了因降维不当导致陷入局部极值的情况,在处理高维稀疏数据集时优势尤为明显。MDC 流程如图 3 所示。

1) 构建 MDC 的目标函数

构建 MDC 的目标函数如式(2)所示。

$$J_{MDC}(\boldsymbol{M},\boldsymbol{P}) = \min_{\boldsymbol{M},\boldsymbol{P}} \sum_{k=1}^{c}\sum_{i=1}^{n} \| \boldsymbol{X}^i_{MSPE}\boldsymbol{P} - \boldsymbol{M}_k\boldsymbol{P} \|_2 + \sigma \sum_{i,j} \| \boldsymbol{X}^i_{MSPE}\boldsymbol{P} - \boldsymbol{X}^j_{MSPE}\boldsymbol{P} \|_2 \boldsymbol{W}_{ij} + \mathfrak{s} \| \boldsymbol{P} \|_{2,1}$$
$$\text{s.t.}\quad \boldsymbol{P}^{\mathrm{T}}\boldsymbol{P} = \boldsymbol{I} \tag{2}$$

其中,c 表示样本类别数,$\boldsymbol{M}$ 表示聚类中心,$\boldsymbol{P}$ 表示投影矩阵,$\boldsymbol{X}^i_{MSPE}\boldsymbol{P}$ 将样本从一个维度映射到另一个维度,其中 $q \ll 2d$,投影空间具有较低的维数,可以去除原空间中的冗余特征。$\| \|_{2,1}$ 表示 $L2,1$ 范数,用于约束投影矩阵 $\boldsymbol{P}$ 的稀疏性。参数 σ 是通过表征损失量化贡献的平衡因子,$\mathfrak{s}$ 表示投影矩阵 $\boldsymbol{P}$ 的权衡系数。

目标函数式(2)的第 1 项表示聚类误差,它由 $\boldsymbol{M}$、$\boldsymbol{X}_{MSPE}$ 和 $\boldsymbol{P}$ 组成,形成于比原始空间更低维的空间,投影矩阵 $\boldsymbol{P}$ 将原空间变换为低维的新空间。参数 $\mathfrak{s}$ 是衡量表征损失贡献的平衡因子,当参数值较大时,目标函数式(2)的第 3 个部分的作用更大。目标函数(2)的第 2 项约束第 1 项使用的投影矩阵 $\boldsymbol{P}$。第 3 项是正则化应用,控制第 2 部分涉及的投影矩阵 $\boldsymbol{P}$ 的结构稀疏性。在此使用 L2,1 范数和 $\boldsymbol{P}$ 具有行稀疏(联合稀疏)的性质,增强了所选特征在投影空间中的可解释性。

局部线性嵌入(locally linear embedding,LLE)技术用于构建亲和矩阵。如式(3)所示,这种方法能够有效地保留数据样本在高维空间中的局部结构特性,从而更好地挖掘数据之间的关系和相似性。

$$\boldsymbol{W}_{ij} = \begin{cases} 1, & \boldsymbol{X}^i_{MSPE} \in N_z(\boldsymbol{X}^j_{MSPE}) \ \| \ \boldsymbol{X}^j_{MSPE} \in N_z(\boldsymbol{X}^i_{MSPE}) \\ 0, & \text{其他} \end{cases} \tag{3}$$

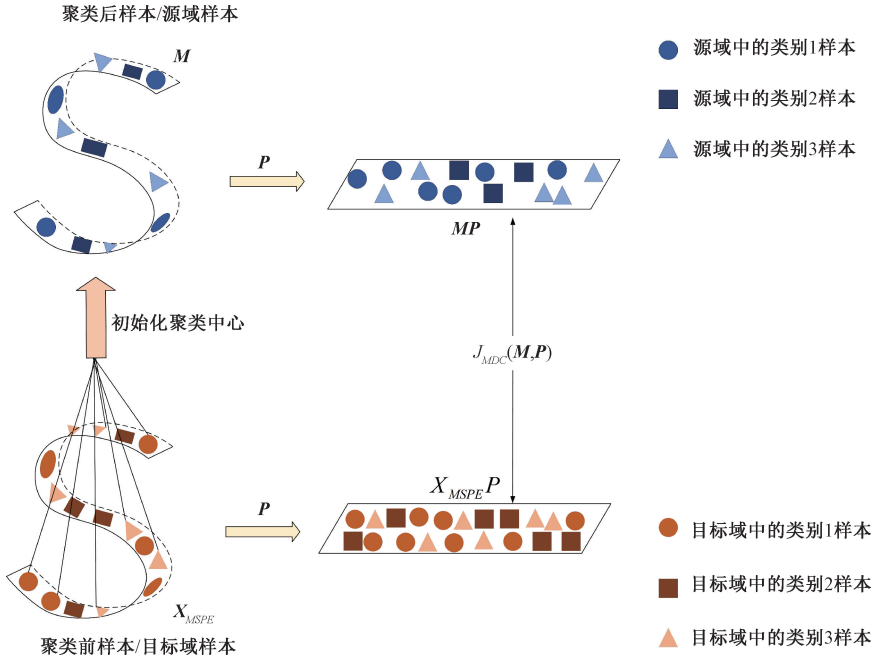


图 3 MDC 原理

Fig. 3 Schematic diagram of MDC

其中，$N_z(\boldsymbol{X})$ 代表样本 $\boldsymbol{X}$ 的第 z 个最近邻样本。

2）优化 MDC 的目标函数

MDC 目标函数中有 2 个变量 $\boldsymbol{M}$ 和 $\boldsymbol{P}$ 需要求解，需要首先初始化聚类中心 $\boldsymbol{M}$。选择 K-means 作为聚类中心初始化方法。为降低特征的冗余度，对已有特征进行 K 均值聚类，得到最佳类别数也称为网络本质特征个数[24]。交替定向乘法（alternating direction method of multipliers，ADMM）以及增强拉格朗日法（augmented Lagrangian method，ALM）可以有效求解，即优化其中一个变量时，固定剩下一个变量。（优化参见链接：https://github.com/DSJPE/DSJPE-ESAE。）

（1）优化 $\boldsymbol{M}$ 时，固定 $\boldsymbol{P}$，目标函数变为：

$$\min_{\boldsymbol{M}} \sum_{k=1}^{c} \sum_{i=1}^{n} \| \boldsymbol{X}_{MSPE}^{i}\boldsymbol{P} - \boldsymbol{M}_k\boldsymbol{P} \|_2 \tag{4}$$

（2）优化 $\boldsymbol{P}$ 时，固定 $\boldsymbol{M}$，目标函数变为：

$$\min_{\boldsymbol{P}} \sum_{k=1}^{c} \sum_{i=1}^{n} \| \boldsymbol{X}_{MSPE}^{i}\boldsymbol{P} - \boldsymbol{M}_k\boldsymbol{P} \|_2 + \sigma \sum_{i,j} \| \boldsymbol{X}_{MSPE}^{i}\boldsymbol{P} - \boldsymbol{X}_{MSPE}^{j}\boldsymbol{P} \| \boldsymbol{W}_{ij} + \varsigma \| \boldsymbol{P} \|_{2,1}$$

$$\text{s. t.} \quad \boldsymbol{P}^{\mathrm{T}}\boldsymbol{P} = \boldsymbol{I} \tag{5}$$

MDC 算法的伪代码如算法 2 所示。

对式（5）的 3 部分进行逐步求解然后再整合见式（6），显然，$\boldsymbol{B}$ 是一个实对称矩阵。因此，可以根据 $\boldsymbol{B}$ 的特征值分解结果求解 $\boldsymbol{P}$，且所得到的 $\boldsymbol{P}$ 必然是正交的。$\boldsymbol{P} \in R^{2d\times q}$ 由特征向量组成，其对应的特征值是 q 的非零最小值。q 代表投影空间中变量的个数。当获得最终优化后的变量 $\boldsymbol{M}$ 和 $\boldsymbol{P}$ 后，由保持降维式聚类模块 MDC 生成的样本可表示为 $\boldsymbol{X}_{MDC} = \boldsymbol{MP}$。（优化过程见链接：https://github.com/DSJPE/DSJPE-ESAE。）

算法 2：MDC 伪代码算法

输入：数据集 $\boldsymbol{X}_{MSPE} \in \boldsymbol{R}^{n\times 2d}$

输出： $\boldsymbol{X}_{MDC} \in \boldsymbol{R}^{u\times q}$

1：设置参数：u，c，σ，ς，q，$\boldsymbol{M}$；计算亲和矩阵 $\boldsymbol{W}$

2：While not converge do

3：根据式（4）优化 $\boldsymbol{M}$，固定 $\boldsymbol{P}$

4：根据式（5）和（6）优化 $\boldsymbol{P}$，固定 $\boldsymbol{M}$

5：End while

6：计算 $\boldsymbol{X}_{MDC} = \boldsymbol{MP}$

$$\min_{\boldsymbol{P}} \sum_{k=1}^{c} \sum_{i=1}^{n} \| \boldsymbol{X}_{MSPE}\boldsymbol{P} - \boldsymbol{M}_k\boldsymbol{P} \|_2 + \zeta \| \boldsymbol{P} \|_{2,1} + \sigma \sum_{i,j} \| \boldsymbol{X}_{MSPE}^{i}\boldsymbol{P} - \boldsymbol{X}_{MSPE}^{j}\boldsymbol{P} \| \boldsymbol{W}_{ij} \Leftrightarrow \min_{\boldsymbol{P}} Tr(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{X}_{MSPE}^{\mathrm{T}}\boldsymbol{X}_{MSPE}\boldsymbol{P}) + Tr(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{M}^{\mathrm{T}}\boldsymbol{MP}) - Tr(2\boldsymbol{P}^{\mathrm{T}}\boldsymbol{X}_{MSPE}^{MSPE}\boldsymbol{MP}) + 2\sigma Tr(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{X}_{MSPE}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_{MSPE}\mathrm{P}) + 2\zeta Tr(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{AP}) \Leftrightarrow \min_{\boldsymbol{P}} Tr(\boldsymbol{P}^{\mathrm{T}}((\boldsymbol{X}_{MSPE}^{\mathrm{T}}\boldsymbol{X}_{MSPE} - (\boldsymbol{X}_{MSPE}^{\mathrm{T}}\boldsymbol{M} + \boldsymbol{M}^{\mathrm{T}}\boldsymbol{X}_{MSPE}) + \boldsymbol{M}^{\mathrm{T}}\boldsymbol{M}) + 2\sigma\boldsymbol{X}_{MSPE}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_{MSPE} + 2\zeta A))\boldsymbol{P}$$

$$\boldsymbol{B} = (\boldsymbol{X}_{MSPE}^{\mathrm{T}}\boldsymbol{X}_{MSPE} - (\boldsymbol{X}_{MSPE}^{\mathrm{T}}\boldsymbol{M} + \boldsymbol{M}^{\mathrm{T}}\boldsymbol{X}_{MSPE}) + \boldsymbol{M}^{\mathrm{T}}\boldsymbol{M}) + 2\sigma\boldsymbol{X}_{MSPE}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_{MSPE} + 2\zeta A) \tag{6}$$

1.3　双级间一致性保持模块

通过 MSPE 和 MDC 这 2 个模块,得到 2 层包络样本,记为 $\boldsymbol{X}_{MSPE}$ 和 $\boldsymbol{X}_{MDC}$。这 2 层样本之间存在分布差异,影响了聚类后样本对聚类前样本的表征能力。为了消除这种分布差异,需要设计层间一致性保持模块 DSICM,从而提高 $\boldsymbol{X}_{MDC}$ 的表征能力。首先令 $\boldsymbol{X}'_{MSPE}=\boldsymbol{X}_{MSPE}\boldsymbol{P}$,使得 MDC 和 MSPE 2 个空间中的样本维度一致,然后对每个样本空间中的矩阵进行转置(即 $\boldsymbol{X}_{MDC}\in\boldsymbol{R}^{q\times u}$,$\boldsymbol{X}'_{MSPE}\in\boldsymbol{R}^{q\times n}$)。MDC 可以通过生成传递矩阵 $\boldsymbol{Q}(\boldsymbol{Q}\in\boldsymbol{R}^{u\times n})$ 生成一个与 MSPE 分布相似的中间样本空间(intermediate sample space,ISS)。DSICM 流程如图 4 所示,经 DSICM 处理后,目标域样本和源域样本达到全局和局部分布的一致性。

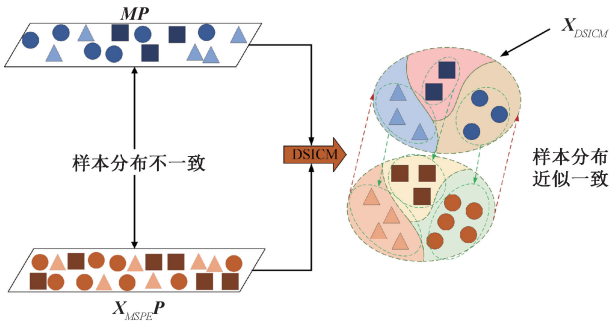


图 4　DSICM 示意图

Fig. 4　Schematic diagram of DSICM

1)构建 DSICM 目标函数

首先引入隐式但通用的变换 φ 来表示 MSPE、MDC 和 ISS 的训练集,并分别定义为 $\varphi(\boldsymbol{X}_{MSPE'})$、$\varphi(\boldsymbol{X}_{MDC})$ 和 $\varphi(\boldsymbol{X}_{ISS})$,为方便起见,将 $\varphi(\boldsymbol{X}^i_{ISS})$ 定义为 ISS 中的第 i 个样本,将 $\varphi(\boldsymbol{X}^j_{MSPE'})$ 定义为 MSPE 中的第 j 个样本,考虑到 $\varphi(\boldsymbol{X}_{ISS})=\varphi(\boldsymbol{X}_{MDC})\boldsymbol{Q}$。因此,DSICM 的目标函数表达式为式(7)。

$$J_{DSICM}(\boldsymbol{\Psi},\boldsymbol{\Phi},\boldsymbol{Q})=J_{LDD}(\boldsymbol{\Phi},\boldsymbol{Q})+\tau J_{GDD}(\boldsymbol{\Psi},\boldsymbol{Q})=\sum_{i,j}^{n}\boldsymbol{W}_{ij}\parallel\varphi(\boldsymbol{X}^i_{ISS})-\varphi(\boldsymbol{X}^j_{MSPE'})\parallel_2^2+\tau\frac{1}{n}\sum_{j=1}^{n}\parallel\varphi(\boldsymbol{X}^j_{ISS})-\varphi(\boldsymbol{X}^j_{MSPE'})\parallel_2^2+\gamma\parallel\boldsymbol{Q}\parallel_* \tag{7}$$

总体而言,DSICM 由 3 个部分组成:第 1 部分是局部分布差异(local distribution difference,LDD)损失 $J_{LDD}(\boldsymbol{\Phi},\boldsymbol{Q})$,即利用 MSPE 的局域性来度量与 ISS 的局部区域差异;第 2 部分是全局分布差异(global distribution difference,GDD)损失 $J_{GDD}(\boldsymbol{\Psi},\boldsymbol{Q})$,最小化了 ISS 和 MSPE 之间边缘分布的全局差异;第 3 部分是低秩约束(low rank constraint,LRC)正则化[25],为了保证生成传递矩阵 $\boldsymbol{Q}$ 的泛化性,$\parallel\boldsymbol{Q}\parallel_*$ 是矩阵 $\boldsymbol{Q}$ 的低秩约束,可以有效显示 MSPE 和 MDC 中样本的全局结构。DSICM 各部分描述为:

(1)全局分布差异 GDD:GDD 以最小化 $\boldsymbol{X}_{MSPE'}$ 和 $\boldsymbol{X}_{ISS}$ 之间边际分布的全局差异。使用线性变换的投影矩阵 $\boldsymbol{\Psi}$ 来找到 $\varphi(\boldsymbol{X}^i_{ISS})$ 和 $\varphi(\boldsymbol{X}^j_{MSPE'})$ 的潜在公共子空间。在此引入了生成传递矩阵 $\boldsymbol{Q}$,并将其代入目标函数式(7)中,则投影后的 GDD 损失公式可以重写为:

$$\min_{\boldsymbol{\Psi},\boldsymbol{Q}}J_{GDD}(\boldsymbol{\Psi},\boldsymbol{Q})=\min_{\boldsymbol{\Psi},\boldsymbol{Q}}\frac{1}{n}\parallel\boldsymbol{\Psi}^{\mathrm{T}}\varphi(\boldsymbol{X}_{MDC})\boldsymbol{Q}-\boldsymbol{\Psi}^{\mathrm{T}}\varphi(\boldsymbol{X}_{MSPE'})\boldsymbol{1}\parallel_2^2 \tag{8}$$

其中,$\boldsymbol{\Psi}$ 可以表示成一些线性组合,即 $\boldsymbol{\Psi}^{\mathrm{T}}=\boldsymbol{\Phi}^{\mathrm{T}}\varphi(\boldsymbol{X}_{OP})^{\mathrm{T}}$,$\boldsymbol{\Phi}$ 表示线性组合系数矩阵,$\boldsymbol{X}_{OP}=[\boldsymbol{X}_{MDC},\boldsymbol{X}_{MSPE'}]$,$\varphi(\boldsymbol{X}_{OP})\in\boldsymbol{R}^{q\times(u+n)}$。则投影后的 MDC 可以表示为 $\boldsymbol{\Phi}^{\mathrm{T}}\varphi(\boldsymbol{X}_{OP})^{\mathrm{T}}\varphi(\boldsymbol{X}_{MDC})$,投影后的 MSPE 可以表示为 $\boldsymbol{\Phi}^{\mathrm{T}}\varphi(\boldsymbol{X}_{OP})^{\mathrm{T}}\varphi(\boldsymbol{X}_{MSPE'})$,在此之后,令 $\boldsymbol{K}_{MSPE'}=\varphi(\boldsymbol{X}_{OP})^{\mathrm{T}}\varphi(\boldsymbol{X}_{MSPE'})$,MDC 和 MSPE 分别可以简单表示为 $\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}$ 和 $\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'}$。因此,GDD 损失公式式(8)可以重写为:

$$\min_{\boldsymbol{\Phi},\boldsymbol{Q}}\frac{1}{n}\parallel\boldsymbol{\Psi}^{\mathrm{T}}\varphi(\boldsymbol{X}_{MDC})\boldsymbol{Q}-\boldsymbol{\Psi}^{\mathrm{T}}\varphi(\boldsymbol{X}_{MSPE'})\boldsymbol{1}\parallel_2^2=\frac{1}{n}\parallel\boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{K}_{PS}\boldsymbol{Q}-\boldsymbol{K}_{MSPE'})\boldsymbol{1}\parallel_2^2 \tag{9}$$

(2)局部分布差异 LDD

$$\min_{\boldsymbol{\Phi},\boldsymbol{Q}}J_{LDD}(\boldsymbol{\Phi},\boldsymbol{Q})=\min_{\boldsymbol{\Phi},\boldsymbol{Q}}\sum_{i,j}^{n}\boldsymbol{W}_{ij}\parallel\varphi(\boldsymbol{X}^i_{ISS})-\varphi(\boldsymbol{X}^j_{MSPE'})\parallel_2^2=Tr(\varphi(\boldsymbol{X}_{ISS})\boldsymbol{D}(\varphi(\boldsymbol{X}_{ISS})^{\mathrm{T}})+Tr(\varphi(\boldsymbol{X}_{MSPE'})\boldsymbol{D}(\varphi(\boldsymbol{X}_{MSPE'})^{\mathrm{T}})-2Tr(\varphi(\boldsymbol{X}_{ISS})\boldsymbol{W}(\varphi(\boldsymbol{X}_{MSPE'})^{\mathrm{T}}) \tag{10}$$

LDD 利用 MSPE 的局部性来度量局部差异,有效捕捉到数据中的局部模式和结构,从而间接增强了 ISS 与 MSPE 之间分布的一致性。目标函数式(7)中的 LDD 可以重写为式(10)。

其中矩阵 $\boldsymbol{D}\in\boldsymbol{R}^{n\times n}$ 是一个对角线矩阵,矩阵中元素 $\boldsymbol{D}_{ii}=\sum_j\boldsymbol{W}_{ij},i=1,\cdots,n$,$Tr(\cdot)$ 是矩阵的迹。$\boldsymbol{W}_{ij}\in\boldsymbol{R}^{n\times n}$ 代表亲和矩阵。将 $\varphi(\boldsymbol{X}_{ISS})=\varphi(\boldsymbol{X}_{MDC})\boldsymbol{Q}$ 代入,利用线性变换后的投影矩阵 $\boldsymbol{\Psi}$ 求出 $\varphi(\boldsymbol{X}^i_{ISS})$ 和 $\varphi(\boldsymbol{X}^j_{MSPE'})$ 的潜在子空间,因此 LDD 损失公式式(10)可重写为:

$$\min_{\boldsymbol{\Phi},\boldsymbol{Q}}\frac{1}{n^2}[Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{D}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q})^{\mathrm{T}})-2Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{W}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'})^{\mathrm{T}})+Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'}\boldsymbol{D}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'})^{\mathrm{T}})] \tag{11}$$

(3)基于 GDD 和 LDD 结合的 DSICM 联合优化:本节提出的 DSICM 旨在保证多层样本空间之间样本分布的一致性,最后,结合以上部分,并引入权衡参数 G 和 γ,DSICM 的目标函数(7)可以重写为:

$$\min J_{DSICM}(\boldsymbol{\Psi},\boldsymbol{\Phi},\boldsymbol{Q})=\min_{\boldsymbol{\Phi},\boldsymbol{Q}}\mathscr{G}\frac{1}{n}\parallel\boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{K}_{MDC}\boldsymbol{Q}-\boldsymbol{K}_{MPSE'})\boldsymbol{1}\parallel_2^2+$$

$$\frac{1}{n^2}[\,Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{D}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q})^{\mathrm{T}}) -$$
$$2Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{W}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'})^{\mathrm{T}})\,] +$$
$$Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'}\boldsymbol{D}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'})^{\mathrm{T}}) + \gamma\,\|\boldsymbol{Q}\|_{*}$$
$$\text{s.t.}\quad \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{\Phi} = \boldsymbol{I} \tag{12}$$

2)优化 DSICM 目标函数

显然,在 DSICM 中,有 2 个变量 $\boldsymbol{\Phi}$、$\boldsymbol{Q}$ 需要求解,可以通过 ADMM 有效求解。通过引入辅助变量 $\boldsymbol{\Theta}$,再利用 ALM,DSICM 的目标函数可以重写为式(13),其中 $\boldsymbol{E}$ 和 $\rho 1$ 分别是增广拉格朗日乘子和惩罚参数,$\boldsymbol{1}$ 表示一个满 1 的矩阵。

$$\min_{\boldsymbol{\Phi},\boldsymbol{Q},\boldsymbol{\Theta}} G\,\frac{1}{n^2}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}(\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{1}(\boldsymbol{K}_{MDC}\boldsymbol{Q})^{\mathrm{T}} -$$
$$\boldsymbol{K}_{MSPE'}\boldsymbol{1}\boldsymbol{Q}^{\mathrm{T}}(\boldsymbol{K}_{MDC})^{\mathrm{T}}) + \boldsymbol{K}_{MSPE'}\boldsymbol{1}(\boldsymbol{K}_{MSPE'})^{\mathrm{T}}) +$$
$$\frac{1}{n^2}(\,Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{D}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q})^{\mathrm{T}}) -$$
$$2Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{W}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'})^{\mathrm{T}})) +$$
$$Tr(\mathrm{E}^{\mathrm{T}}(\boldsymbol{Q}-\boldsymbol{\Theta})) + \frac{\rho 1}{2}(\,\|\boldsymbol{Q}-\boldsymbol{\Theta}\|_F^2) +$$
$$Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'}\boldsymbol{D}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'})^{\mathrm{T}}) -$$
$$\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{1}(\boldsymbol{K}_{MDC})^{\mathrm{T}} + \gamma\,\|\boldsymbol{\Theta}\|_{*} \tag{13}$$

变量 $\boldsymbol{\Phi}$,$\boldsymbol{Q}$ 和 $\boldsymbol{\Theta}$ 的优化过程可采用变量交替策略求解。

(1)优化 $\boldsymbol{\Phi}$ 时,固定 $\boldsymbol{Q}$ 和 $\boldsymbol{\Theta}$,将式(13)中的目标函数转换为:

$$\min_{\boldsymbol{\Phi}} \frac{\boldsymbol{G}}{n^2}\boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{1}(\boldsymbol{K}_{MDC}\boldsymbol{Q})^{\mathrm{T}} +$$
$$\frac{1}{n^2}[\,Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{D}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q})^{\mathrm{T}}) -$$
$$2Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{W}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'})^{\mathrm{T}}) +$$
$$Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'}\boldsymbol{D}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'})^{\mathrm{T}}) - \boldsymbol{K}_{MSPE'}\boldsymbol{1}\boldsymbol{Q}^{\mathrm{T}}(\boldsymbol{K}_{MDC})^{\mathrm{T}}) +$$
$$\boldsymbol{K}_{MSPE'}\boldsymbol{1}(\boldsymbol{K}_{MSPE'})^{\mathrm{T}})\,] - \boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{1}(\boldsymbol{K}_{MDC})^{\mathrm{T}}]$$
$$\text{s.t.}\quad \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{\Phi} = \boldsymbol{I} \tag{14}$$

(2):优化 $\boldsymbol{Q}$ 时,固定 $\boldsymbol{\Phi}$ 和 $\boldsymbol{\Theta}$,将式(13)中的目标函数转换为:

$$\min_{\boldsymbol{Q}} \frac{\mathscr{G}}{n^2}\boldsymbol{\Phi}^{\mathrm{T}}(\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{1}(\boldsymbol{K}_{MDC}\boldsymbol{Q})^{\mathrm{T}} - \boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{1}(\boldsymbol{K}_{MDC})^{\mathrm{T}} +$$
$$\frac{1}{n^2}(\,Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{D}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q})^{\mathrm{T}}) -$$
$$2Tr(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MDC}\boldsymbol{Q}\boldsymbol{W}(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}_{MSPE'})^{\mathrm{T}})) +$$
$$Tr(\boldsymbol{E}^{\mathrm{T}}(\boldsymbol{Q}-\boldsymbol{\Theta})) + \frac{\rho 1}{2}(\,\|\boldsymbol{Q}-\boldsymbol{\Theta}\|_F^2) -$$
$$\boldsymbol{K}_{MSPE'}\boldsymbol{1}\boldsymbol{Q}^{\mathrm{T}}(\boldsymbol{K}_{MDC})^{\mathrm{T}}))\boldsymbol{\Phi} \tag{15}$$

(3)优化 $\boldsymbol{\Theta}$ 时,固定 $\boldsymbol{\Phi}$ 和 $\boldsymbol{Q}$,将式(13)中的目标函数转换为:

$$\min_{\boldsymbol{\Theta}} \gamma\,\|\boldsymbol{\Theta}\|_{*} + Tr(\boldsymbol{E}^{\mathrm{T}}(\boldsymbol{Q}-\boldsymbol{\Theta})) + \frac{\alpha}{2}(\,\|\boldsymbol{Q}-\boldsymbol{\Theta}\|_F^2) \tag{16}$$

获得最优 $\boldsymbol{\Phi}$ 时,$\boldsymbol{X}'_{MDC} = [\boldsymbol{\Phi}^{\mathrm{T}}\varphi(\boldsymbol{X}_{OP})^{\mathrm{T}}\varphi(\boldsymbol{X}_{MDC})]^{\mathrm{T}}$ 得到与 MSPE 保持样本数据结构信息的局部和全局一致性后的一次聚类包络样本集 $\boldsymbol{X}'_{MDC} \in \boldsymbol{R}^{u\times d'}$,其中 d' 表示 DSICM 后的样本维数。

DSICM 算法的伪代码如算法 3 所示。

算法 3:DSICM 伪代码算法

输入:样本对矩阵 $\boldsymbol{X}'_{MSPE} \in \boldsymbol{R}^{n\times q}$,$\boldsymbol{X}_{MDC} \in \boldsymbol{R}^{u\times q}$

输出:$\boldsymbol{X}'_{MDC} \in \boldsymbol{R}^{u\times d'}$

1:矩阵转置 $\boldsymbol{X}'_{\mathrm{MSPE}} \in \boldsymbol{R}^{q\times n}$,$\boldsymbol{X}_{MDC} \in \mathbf{R}^{q\times u}$

2:依次进行如下计算:

$\boldsymbol{X}_{OP} = [\boldsymbol{X}_{MDC}, \boldsymbol{X}_{MSPE'}]$,

$\boldsymbol{K}_{MSPE'} = \varphi(\boldsymbol{X}_{OP})^{\mathrm{T}}\varphi(\boldsymbol{X}_{MSPE'})$,

$\boldsymbol{K}_{MDC} = \varphi(\boldsymbol{X}_{OP})^{\mathrm{T}}\varphi(\boldsymbol{X}_{MDC})$,$\boldsymbol{K} = \varphi(\boldsymbol{X}_{OP})^{\mathrm{T}}\varphi(\boldsymbol{X}_{OP})$,

$\boldsymbol{Q} = \boldsymbol{\Theta} = 0$

3:While not converge do

4: 根据式(14)~(16)优化 $\boldsymbol{\Phi}$、$\boldsymbol{\Theta}$ 和 $\boldsymbol{Q}$

5: 更新乘数 $\boldsymbol{E}$:$\boldsymbol{E} \leftarrow \boldsymbol{E} + \rho 1(\boldsymbol{Q}-\boldsymbol{\Theta})$

6: 更新参数 $\rho 1$:$\rho 1 \leftarrow \min(\rho 1 \times 1.01, \max_{\rho 1})$

7:End while

8:计算 $\boldsymbol{X}'_{MDC} = [\boldsymbol{\Phi}^{\mathrm{T}}\varphi(\boldsymbol{X}_{OP})^{\mathrm{T}}\varphi(\boldsymbol{X}_{MDC})]^{\mathrm{T}}$

1.4 内嵌式堆栈自动编码器

SAE 利用多层自编码器提取数据的复杂非线性特征[26]。图 5 显示本文作者前期设计的内嵌式堆栈自动编码器-ESAE。与传统 SAE 不同,ESAE 在相邻隐含层中间引入了自动编码器的输入层特征(即嵌入单元),提高了深度特征与原特征的互补性。

假设 ESAE 网络的输入样本矩阵为 $\boldsymbol{X}_{OS}$,其中 $\boldsymbol{H}^k = [\boldsymbol{h}_1^k, \boldsymbol{h}_2^k, \cdots, \boldsymbol{h}_n^k] \in \boldsymbol{R}^{n\times d^k}(1 < k < K)$ 表示为第 k 个编码器中隐含层的输出矩阵。d^k 表示第个 k 编码器中的隐含层神经元个数。ESAE 第 k 编码器的优化目标函数写成:

$$\min_{\theta} \frac{1}{n}\sum_{i=1}^{n}\|\boldsymbol{L}(\boldsymbol{V}^{k-1}) - \boldsymbol{L}'(\boldsymbol{V}^{k-1})\|^2 +$$
$$\lambda(\,\|\boldsymbol{W}_{k1}\|_2 + \|\boldsymbol{W}_{k2}\|_2) + \beta(\sum_{j=1}^{d^k}\mathrm{KL}(\rho\,\|\,\hat{\rho}_j)) \tag{17}$$

其中,λ 和 β 分别是 ESAE 正则化项和稀疏准则项的惩罚参数。$\boldsymbol{L}(\boldsymbol{V}^{k-1})$ 为嵌入单元输入,$\boldsymbol{L}'(\boldsymbol{V}^{k-1})$ 为重构嵌入单元输出得到的数据。

经过 2 个阶段的自编码器网络预训练和微调之后,对于网络中的第 i 个输入向量矩阵可以表示为 $\boldsymbol{X}_{iOS} = [\phi(\boldsymbol{x}_{i1}), \phi(\boldsymbol{x}_{i2}), \cdots, \phi(\boldsymbol{x}_{i2d})]$,网络中的每个隐含层输

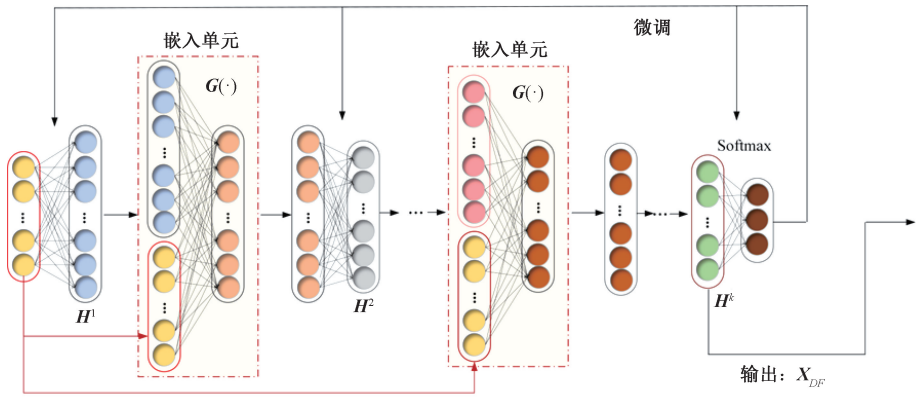


图 5　ESAE 示意图

Fig. 5　Schematic diagram of ESAE

出一个新的特征向量,都代表着不同特征层次的信息。用 q 代表最后一个隐含层的神经元数量,将最后一个隐含层的输出作为自编码器网络学习到的深度特征,则可以表示为:

$$\boldsymbol{X}_{DF} = [\boldsymbol{x}'_1, \boldsymbol{x}'_2, \cdots, \boldsymbol{x}'_d] \in \boldsymbol{R}^{n \times q} \tag{18}$$

1.5　计算复杂度分析

DSJPE-ESAE 的计算复杂度主要包括 4 个任务的总和:MSPE 的复杂度、MDC 的复杂度、DSICM 的复杂度和 ESAE 的复杂度。可以表示为 $T_{DSJPE-ESAE} = T_{MSPE} + T_{MDC} + T_{DSICM} + T_{ESAE}$。

1)MSPE 的复杂度为 $O(n)$,其中 n 为样本个数。

2)MDC 的计算复杂度包括 4 个基本步骤:优化 $\boldsymbol{M}$、优化 $\boldsymbol{P}$、优化 $\boldsymbol{H}$ 和优化 $\boldsymbol{L}$。计算 $\boldsymbol{M}$ 的复杂度为 $O(cDn)$,计算 $\boldsymbol{P}$ 的复杂度为 $O(D^3)$,计算 $\boldsymbol{H}$ 的复杂度为 $O(cDnq)$,计算 $\boldsymbol{L}$ 的复杂度为 $O(Dn^2q)$,假设迭代次数为 T,则 MDC 的总计算复杂度可表示为 $O(TcDn + TD^3 + TcDnq + TDn^2q)$。MDC 的计算复杂度可以简化为 $O(TD^3 + TcDnq + TDn^2q)$。

3)DSICM 的计算复杂度包括 3 个基本步骤:优化 $\boldsymbol{Q}$、优化 $\boldsymbol{\Theta}$ 和优化 $\boldsymbol{\Phi}$。计算 $\boldsymbol{\Theta}$ 和 $\boldsymbol{Q}$ 的复杂度为 $O(n^2)$。$\boldsymbol{\Phi}$ 的计算涉及到特征分解和矩阵乘法,复杂度为 $O(n^3)$。假设迭代次数为 T,DSICM 的总计算复杂度可表示为 $O(Tn^2 + Tn^3)$,可简化为 $O(Tn^3)$。值得注意的是,这里没有考虑 ***Gram*** 矩阵的计算复杂度,因为它可以提前计算,而不必在 DSICM 中计算。

4)ESAE 的计算复杂度与隐含层神经元的数量有关,它们分别为 h_1, h_2, h_3,因此 ESAE 的计算复杂度为 $O(h_1 + h_2 + h_3)$。同时还有反向传播部分,其计算复杂度为 $O(Tnl)$,其中 T 为迭代次数,n 为样本个数,l 为计算单个样本梯度的计算复杂度;因此,ESAE 的总计算复杂度可以表示为 $O(h_1 + h_2 + h_3) + O(Tnl)$。

综上所述,所有给定算法的复杂度之和为 $O(n + TD^3 + TcDnq + TDn^2q + Tn^3 + h_1 + h_2 + h_3 + Tnl)$,简化为 $O(TD^3 + TcDnq + TDn^2q + Tn^3 + h_1 + h_2 + h_3 + Tnl)$。

2　实基于 DSJPE-ESAE 的双层空间融合机制

DSJPE-ESAE 可以获得 2 组深度特征。本节设计的双层空间融合机制(dual layer spatial integration mechanism,DLSIM)针对每组深度特征各训练一个基分类器,然后加权融合 2 个分类结果(weighted fusion,WF),与现有基于 SAE 的分类算法是不同的。DLSIM 融合了 2 层 ESAE 的分类结果,实现了样本和特征间的协同转换。假设在第 1 层包络样本上的预测结果为 y_0,在第 2 层包络样本上的预测结果为 y_1,则 WF 的实现过程为:

$$\begin{aligned} &\underset{\boldsymbol{w}}{\operatorname{argmax}} = \sum_{i=0}^{\Lambda_v} \wp(\mathbf{round}(\boldsymbol{w}_i^{\mathrm{T}} r_i), y_i) \\ &\text{s.t.}\quad \boldsymbol{w}^{\mathrm{T}} \boldsymbol{I} = 1, \boldsymbol{I} = (1, 1, \cdots, 1) \end{aligned} \tag{19}$$

假设每个编码器在验证集上的预测结果为 $r = (r_0, r_1)$,则最优权向量为 $\boldsymbol{w} = (\boldsymbol{w}_0, \boldsymbol{w}_1)$,最优权向量可由式(19)得到。其中 Λ_v 表示为验证集的个数,$\wp(a,b) = \begin{cases} 1, a = b \\ 0, a \neq b \end{cases}$,假设 $y = (y_0, y_1)$。最终测试样本的预测结果 Y 可表示为:$Y = round(\boldsymbol{w}^{\mathrm{T}} y)$。

3　实验结果与分析

DSJPE-ESAE 主要包括 MSPE、MDC、DSICM 这几个创新点。为了验证这些创新点的有效性,组织了 4 组实验。第 1 组基于消融实验验证 MSPE、MDC、DSICM 和 ESAE 的有效性。第 2 组实验将 DSJPE-ESAE 与代表性 SAE 模型进行比较。第 3 组实验分析了关键参数对算法性能的影响。第 4 组实验进一步分析了算法的性能。

(实验所用的数据集和完整的结果参见:https://github.com/DSJPE/DSJPE-ESAE。)

3.1 实验环境

本研究的实验平台是 64 位 windows 10 操作系统,硬件部分是带有 Inter(R)CORE(TM)CPU i5-8400 的计算机 CPU@2.8 GHZ 处理器和 8 GB RAM,以及开发工具为 Matlab R2018(b),实验中用到的工具箱为 libsvm 3.20 版本。实验采用了 15 个具有代表性的公共数据集。所用的数据集的详细信息可在链接(https://archive.ics.uci.edu/ml/index.php)中找到。这些数据集有代表性,覆盖了不同样本数(由 90~20 000 不等),不同类别数(由 2~26 分类不等),不同特征数(由 8~561 不等)。数据集的主要信息见表 1。

表 1 数据集的基本信息

Table 1 Basic information about the dataset

数据集	样本数	特征数	类别数
AD(Alzheimer's disease)	90	32	3
LSVT(LSVT voice rehabilitation dataset)	126	310	2
PD(Parkinson speech dataset)	1 040	26	2
Pendigits(pen-based recognition of handwritten digit)	10 992	16	10
Statlog(Statlog landsat_satellite)	6 435	36	6
Vehicle(Statlog vehicle silhouettes)	846	18	4
Heart(Statlog heart dataset)	270	13	2
Maxlettle(Maxlettle Parkinson dataset)	195	22	2
Urban(urban land cover)	675	147	9
WDBC(breast cancer Wisconsin diagnostic)	569	30	2
Wisconsin(breast cancer Wisconsin original)	683	9	2
PID(Pima Indians diabetes dataset)	768	8	2
LR(letter recognition)	20 000	16	26
GSAD(gas sensor array drift)	13 910	128	6
HAR(human activity recognition)	10 299	561	6

为了公平比较,采用了结构化数据集来验证各个 SAE 的性能。此外,数据集的样本量不是太大,因此 SAE 的层数设置为 3~6 层就可以满足充分训练要求了。实验中,根据数据集的样本数量和特征维数确定每个自动编码器的隐含层神经元数量,并通过网格搜索方法确定最优结构。

为了公平比较,对比的各个 SAE 的参数均最优设置。实验采用 hold-out 交叉验证方法,训练集:验证集:测试集的比例为 1:1:1。每个数据集重复 5 次实验取平均值和标准偏差作为本章实验的分类结果,用于评估算法的性能。实验中使用分类精度 ACC、MACRO-F1(F1)、AUC 和平均精确度(average precision,AP)来评价不同模型的模型性能。表格实验中的最佳结果用粗体展示。如未说明,使用的分类器均是支持向量机(support vector machines,SVM)。

3.2 消融实验

通过对比实验来验证所提出的 DSJPE-ESAE 算法,本研究算法的主要创新点在于 MSPE、MDC、DSICM、ESAE。因此采用消融实验来对比创新前后的效果。

1)MSPE、MDC 和 DSICM 的有效性验证

这里比较了原样本(original features, OF)、流形样本对拼接 MSPE 学习的样本(MSPE)、使用保持降维式聚类模块来生成下一级样本空间(MDC)、MDC 学习并经双级间一致性保持模块 DSICM 处理的样本(DSICM)和最终得到的双级包络样本(DSJPE)的质量。实验结果如表 2 和 3 所示。

表 2 本研究算法各阶段的样本的质量(ACC)

Table 2 Quality of samples at each stage of the algorithm in this paper (ACC) (%)

数据集	OF	MSPE	MDC	DSICM	DSJPE
AD	54.00	64.67	66.67	72.67	**75.58**
LSVT	80.48	94.29	95.24	95.80	**96.54**
Pendigits	98.13	98.62	99.09	98.78	**99.27**
Statlog	86.13	**88.65**	85.86	86.02	88.36
heart	80.89	85.56	90.91	91.20	**92.44**
WDBC	95.66	97.57	98.59	98.99	**99.18**
LR	85.84	89.65	87.92	89.06	**89.79**
GSAD	99.20	**99.45**	96.67	97.45	96.05
HAR	98.25	98.72	98.36	98.35	**99.10**

表 3 本研究算法各阶段的样本的质量(F1)

Table 3 Quality of samples at each stage of the algorithm in this paper (F1) (%)

数据集	OF	MSPE	MDC	DSICM	DSJPE
AD	50.24	64.78	72.09	73.85	**74.12**
LSVT	76.57	93.61	94.75	95.24	**95.77**
Pendigits	97.74	**99.04**	97.95	98.03	**99.04**
Statlog	80.98	**85.60**	75.24	77.26	85.59
heart	79.95	86.14	90.43	90.86	**91.09**
WDBC	95.23	97.63	97.31	98.80	**98.85**
LR	85.83	89.61	87.87	89.01	**90.18**
GSAD	99.16	**99.43**	96.55	97.37	96.17
HAR	98.35	98.79	98.46	98.42	**99.20**

从表 2 中的结果可见,MSPE 在所有数据集上都显著优于 OF,这说明 MSPE 是有效的,可能的原因是 MSPE 挖掘了样本间局部关联信息。MSPE 和 MDC 之间的比较表明,MDC 结合 MSPE 的组合更加有效,可能的原因是 MDC 挖掘了样本间全局关联信息。DSICM 优于 MDC,这说明双级间一致性保持方法是有效的,也就是 MSPE、MDC 与 DSICM 三者的结合更有效。此外,在各个评价指标下,DSJPE 在大多数数据集上都取得了最优异的性能,这表明基于双层包络样本 SAE 建模范式是最有效的。

2)MDC 和 DSICM 的有效性验证

为进一步验证 MDC 的有效性,本研究设计了单纯使用 K-menas 聚类来生成新样本(KM)与使用 K-menas 聚类学习后经 DSICM 处理的新样本(KM&DSICM)的比较,以及 MDC 和 DSICM 生成的新样本的比较。基于各种样本采用 ESAE 建模的结果比较如表 4 和 5 所示。

表 4　MDC 与 KM 在 ACC 上的比较结果

Table 4　Comparative results of MDC and KM on ACC

(%)

数据集	KM	MDC	KM&DSIM	DSICM	DSICM (ESAE)	DSICM (ESAE)
AD	75.33	66.67	66.00	72.67	67.33	**76.00**
LSVT	92.86	95.24	94.76	95.80	96.07	**97.41**
Pendigits	98.64	99.09	98.83	98.78	98.84	**99.50**
Statlog	85.59	85.86	85.13	86.02	85.92	**88.57**
heart	90.89	90.91	88.44	91.20	90.96	**92.67**
WDBC	97.99	98.59	98.73	98.99	98.10	**99.15**
LR	87.73	87.92	89.83	89.06	88.09	**90.07**
GSAD	96.60	96.67	97.58	**97.45**	97.94	97.72
HAR	98.35	98.36	98.58	98.35	98.81	**99.39**

表 5　MDC 与 KM 在 F1 上的比较结果

Table 5　Comparative results of MDC and KM on F1

(%)

数据集	KM	MDC	KM&DSIM	DSICM	DSICM (ESAE)	DSICM (ESAE)
AD	53.70	72.09	70.01	73.85	**75.00**	74.59
LSVT	92.50	94.75	94.19	**95.24**	92.18	94.73
Pendigits	98.62	97.95	96.21	98.03	98.99	**99.46**
Statlog	81.49	75.24	79.46	77.26	85.95	**86.81**
heart	79.77	90.43	88.54	90.86	91.80	**93.29**
WDBC	95.94	97.31	98.65	98.80	**99.32**	96.69
LR	87.67	87.87	**89.91**	89.01	84.32	89.77
GSAD	96.42	96.55	97.49	97.37	**98.42**	95.24
HAR	98.46	98.46	98.65	98.42	94.25	**98.98**

KM 和 MDC 之间的比较表明,MDC 在大多数数据集上都超过了 KM,这表明 MDC 是更有效的,能构造质量更高的样本。可能的原因是 MDC 降低在高维稀疏空间里聚类陷入局部极值的风险。此外,基于各种评价指标,同样的聚类方法采用 DSICM 后都取得了明显的效果改善,这再次说明了层间一致性是必要和有效的,也就是 MDC 与 DSICM 的结合是更有效的。

3.3　算法比较

为了进一步验证本研究算法 DSJPE-ESAE 的有效性,这里选择了一些代表性的 SAE,包括堆栈式剪枝稀疏自编码器(SPSAE)[27]、内嵌式 L1 正则化和流形约简的堆栈式组稀疏自编码器集成(ESGSAE_FF)[28] 和门控式堆栈目标相关自编码器(GSTAE)[29] 等。实验结果见表 6 和 7。

表 6　不同 SAE 算法的比较(ACC)

Table 6　Comparison of different SAE algorithms (ACC)

(%)

数据集	SPSAE	ESGSAE-FF	GSTAE	SGAE	DSJPE-ESAE
AD	57.78	67.33	71.11	56.11	**76.67**
LSVT	84.33	92.76	84.66	71.59	**97.62**
Pendigits	91.60	98.00	93.53	90.33	**99.54**
Statlog	85.87	87.28	85.42	74.13	**89.42**
heart	88.90	84.67	82.56	69.67	**94.67**
WDBC	93.03	**99.81**	99.34	90.65	98.08
LR	94.88	**95.55**	92.10	84.30	94.38
GSAD	98.89	**99.07**	97.42	87.86	96.71
HAR	98.13	97.81	98.22	97.03	**99.51**

表 7　不同 SAE 算法的比较(F1)

Table 7　Comparison of different SAE algorithms (F1)

(%)

数据集	SPSAE	ESGSAE-FF	GSTAE	SGAE	DSJPE-ESAE
AD	67.02	69.32	67.19	66.67	**71.86**
LSVT	81.87	88.92	63.25	78.53	**95.79**
Pendigits	91.51	99.36	91.68	91.51	**99.55**
Statlog	75.75	82.76	81.71	75.55	**87.81**
heart	81.24	84.53	82.54	57.16	**93.61**
WDBC	94.36	94.44	94.30	92.83	**97.15**
LR	96.28	**96.34**	83.27	87.41	90.18
GSAD	**99.39**	99.37	95.35	89.50	96.73
HAR	98.72	99.08	97.39	96.60	**99.78**

结果表明,DSJPE-ESAE 在大部分情况下均优于其他对比算法。进一步观察发现,DSJPE-ESAE 在大部分数据集上获得了更低的方差。这意味着,相比其他方法,DSJPE-ESAE 具有更强的鲁棒性和稳定性。除此之外,其余 3 项评价指标也呈现类似结果。可能的原因是:DSJPE-ESAE 训练过程中能够捕获样本之间的关联性;保持降维式聚类技术能够在保留数据特征的同时减少特征空间的维度,提高了模型的计算效率和降低了计算复杂度;构建双层包络样本可以帮助模型更好地理解数据样本间的关联结构信息,从而提高模型对数据的抽象表示能力和泛化能力。

为了进一步验证 DSJPE-ESAE 的优势,本研究比较各种 SAE 基于 Statlog 数据集上学习到的特征的表征能力和可分度(二维图形),结果如图 6 所示,图中不同的形状对应不同的类别。

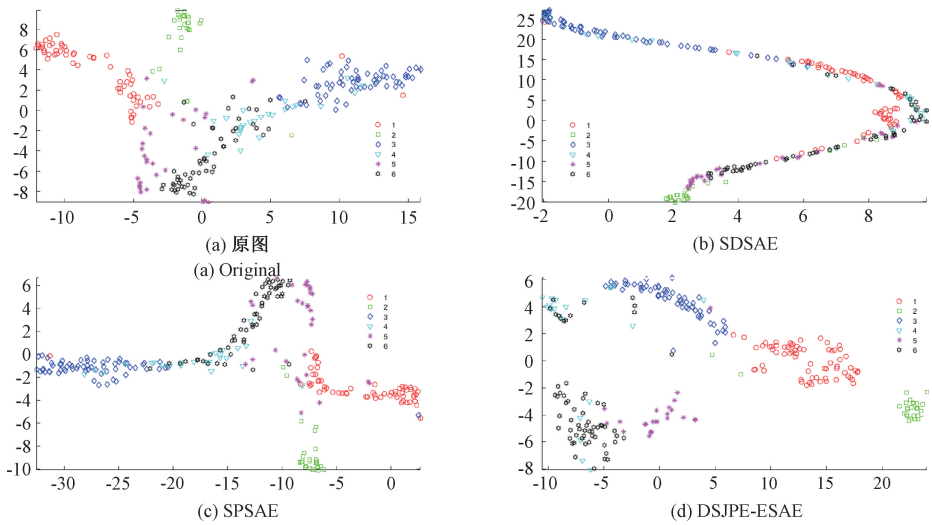


图 6 基于 Statlog 数据集提取的深度特征的可视化比较

Fig. 6 Visual comparison of extracted depth features based on the Statlog dataset.

图 6 结果表明,DSJPE-ESAE 提取的深度特征比其他 SAE 模型具有更好的可分度,即类间样本相距较远,类内样本更为紧凑。

3.4 参数分析

1)MSPE 的拼接数 υ 对算法的影响

流形最近邻样本拼接数 υ 是 MSPE 的关键参数,其影响分析结果如表 8 所示。

表 8 不同 MSPE 拼接数 υ 对算法性能影响

Table 8 Effect of different MSPE splicing number υ on algorithm performance

数据集	υ = 0	υ = 1	υ = 2	υ = 3
AD	54.00±9.55	64.67±4.47	**73.33±6.24**	71.33±3.80
LSVT	82.38±9.00	**96.19±3.61**	95.71±5.16	95.71±3.10
Pendigits	98.13±0.05	98.62±0.07	99.20±0.24	**99.39±0.13**
Statlog	86.13±0.53	88.65±0.75	89.13±0.62	**89.73±0.81**
heart	80.89±4.26	85.56±2.83	84.67±3.46	**89.11±2.53**
WDBC	95.66±1.52	97.57±1.38	99.58±0.24	**100.0±0.00**
LR	85.84±0.16	89.65±0.21	90.41±0.31	**92.26±0.17**
GSAD	99.20±0.15	99.45±0.09	99.56±0.11	**99.59±0.08**
HAR	98.25±0.32	98.72±0.07	99.38±0.15	**99.68±0.03**

结果表明,在大多数数据集上,增加流形最近邻样本的数量可以显著提高样本的分类精度。然而,数量增大将增加特征维度,从而导致模型训练时间代价显著增加。因此,在实际应用中,需要平衡分类精度和训练时间成本,来选择合适的 MSPE 拼接数 υ。

2)MDC 的的参数 μ、σ、γ 的影响

MDC 有 3 个关键参数,分别是 MDC 聚类中心初始化比例 μ,量化代表性损失贡献的平衡因子 σ 和参数 γ 控制投影矩阵 $\boldsymbol{P}$ 的稀疏性。图 7 中每个点的颜色表示 ACC 和 F1 对应于参数值集 (μ,σ,γ) 的值。

结果表明,较大的 μ 和 σ 值可以获得较好的性能。基于此,假设 μ 约为 0.9,σ 约为 1 000,它们可以被认为是运行 MDC 方法的潜在选择。更大的 μ 可以确保新空间中更多的样本来自前一个空间,从而增加训练样本的数量,提高新层的性能。但是考虑到较大的 μ 值可能会降低不同层之间的样本多样性,在中间阶段取 μ 值可以达到最好的效果,这样既可以保证新空间中有足够的训练样本,又可以保证不同空间之间的样本多样性,从而提高后续整体模型的融合性能。较大的 σ 值可以使 MDC 能更有效地保留数据之间的局部邻域结构。此外,可以发现,参数 γ 对模型影响不太显著。

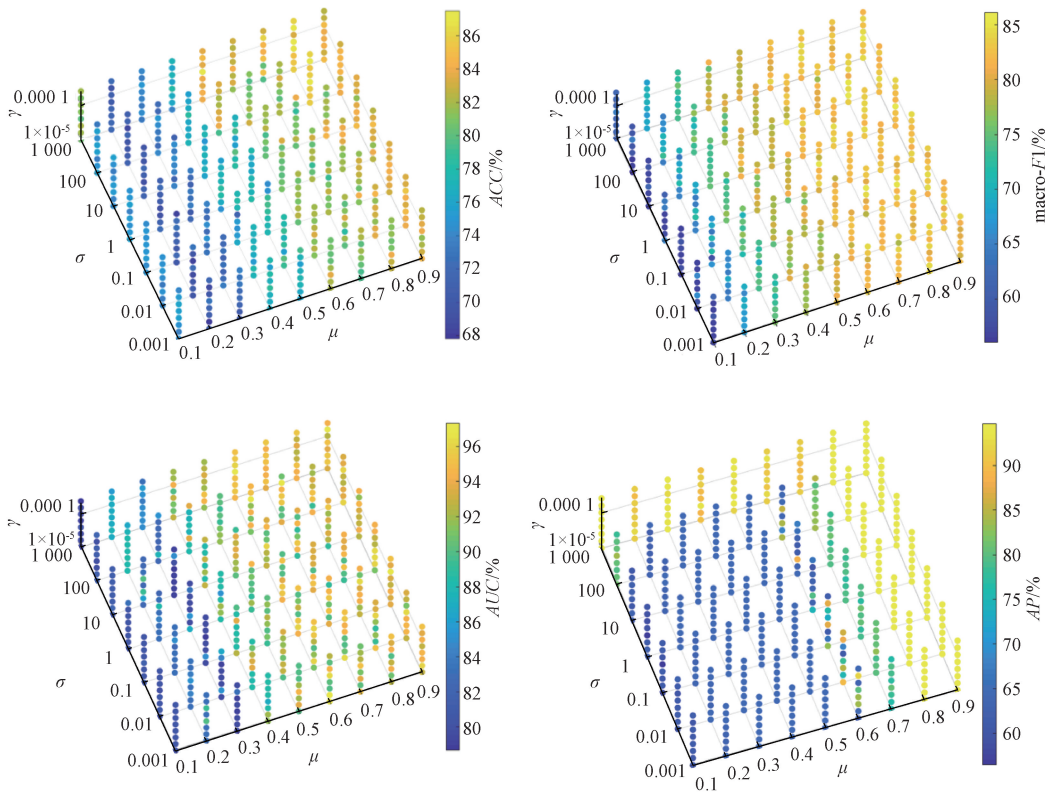
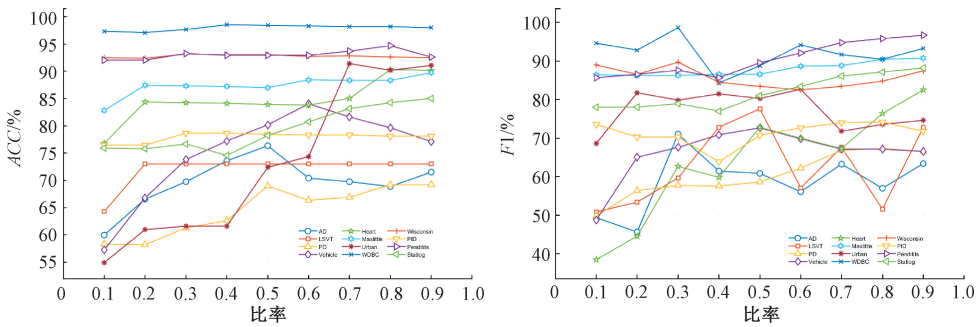


图 7　不同 MDC 参数值集 μ、σ 和 γ 对算法性能的影响

Fig. 7　Effect of different sets of MDC parameter values, and on the performance of the algorithm

3) MDC 降维参数 δ 对算法性能的影响

降维参数 δ 为降维后的维度数与原维度数的比例,直接影响第 2 层包络样本质量。为了验证其对模型的敏感性,对 12 个数据集进行了研究。实验结果如图 8 所示。



(a) ACC　　(b) $F1$

图 8　不同 MDC 降维参数 δ 对算法的影响

Fig. 8　Effect of different MDC downscaling parameters on the algorithms

随着 δ 的增大,在不同维度大小下,各指标的性能呈现先上升后下降的趋势。可能的原因是:随着维度增加,模型可以更好地捕捉数据中的复杂关系和结构,从而提高分类准确性。然而,当维度过高时,模型可能会过度拟合观察到的边缘数据点,导致对新数据边缘的泛化能力降低,进而导致整体性能下降。总之,每个数据集的性能在相似的范围内达到最佳值。

3.5　算法性能分析

1)混淆矩阵

这里选择 3 个有代表性的数据集计算混淆矩阵,它们是六分类任务(Statlog)、九分类任务(Urban)和十分类任务(Pendigits)。结果如图 9 所示。针对 Pendigits 数据集,该模型对所有 10 个类别的平均分类精度达到 99.0%。类别 5 的识别准确率为 100%,而其他类别只有

少量误分类样本,类别 1~4 和 6~10 的分类精度分别为 98.4%、99.2%、96.3%、99.5%、99.7%、99.5%、99.4%、99.4%和 98.2%。实验结果表明,DSJPE-ESAE 在各分类任务中表现出色。

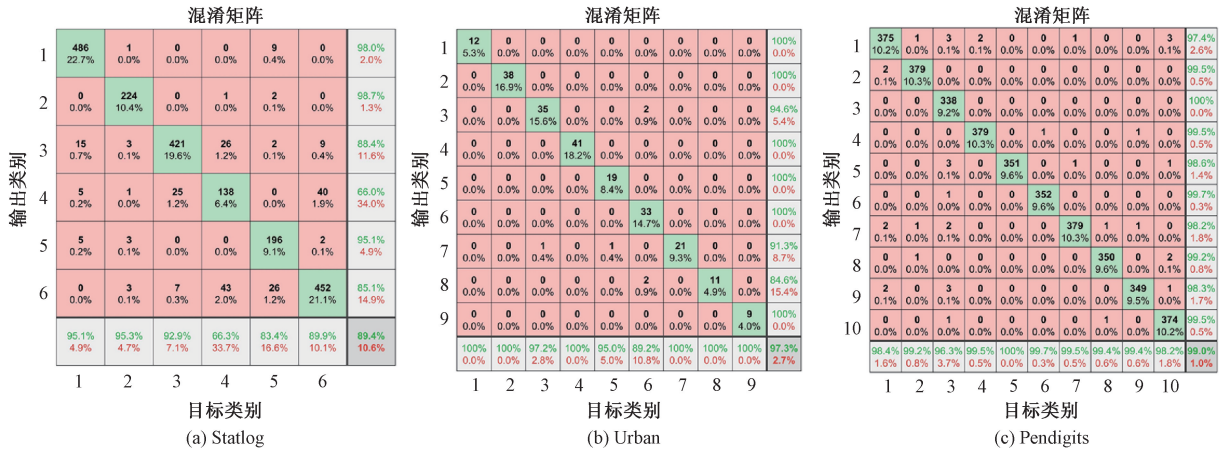


图 9 本文算法在 3 个数据集上的混淆矩阵结果

Fig. 9 Confusion matrix of this paper's algorithm on 3 datasets

2)时间代价分析

这里进一步比较 DSJPE-ESAE 和其他 SAE 的训练时间代价。本节记录了各 SAE 模型在不同数据集上的运行时间。为了公平比较,所有网络都有相同数量的隐含层;比较算法的所有参数都设置为其调谐范围的中值。针对 LSVT,Statlog,GSAD 和 HAR 这 4 个数据集的计算时间比较结果如图 10 所示。可以看出,DSJPE_ESAE 模型的平均准确率最优,但时间代价居中。也就是说,尽管 DSJPE_ESAE 模型的训练时间不是最短的,但是分类精度的提升明显,综合性能较优。

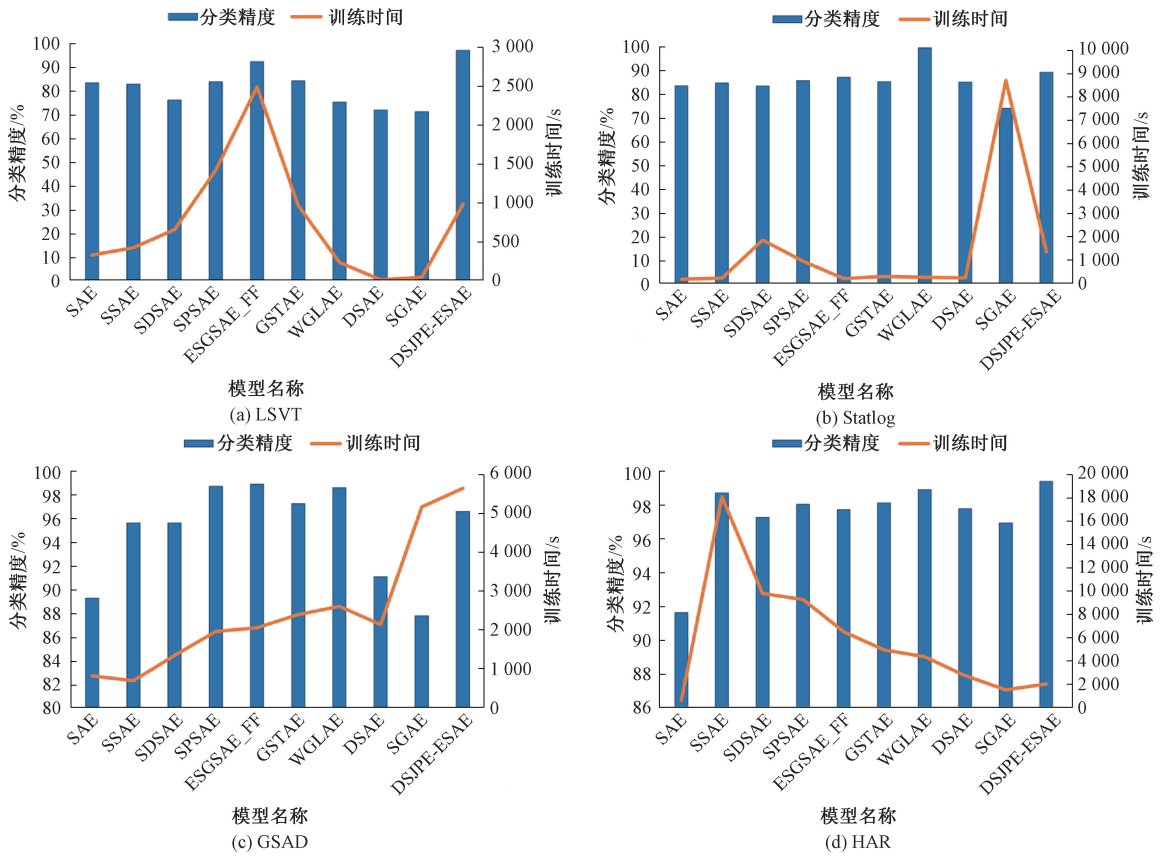


图 10 训练时间比较

Fig. 10 Comparison of training times

3.6　实际场景的应用效果与分析

本节将 DSJPE-ESAE 应用在 2 个实际应用场景里用于评估其效果。这 2 个实际场景分别是帕金森语音诊断(PD)以及糖尿病文本诊断(PID)。2 个场景的实采数据均来源于重庆西南医院的神经内科和内科。分别采用现有的几个自动编码器和 DSJPE-ESAE 基于实采数据构建临床诊断模型,并进行试验,试验结果如表 9 所示。

表 9　PD 及 PID 在不同 SAE 算法上结果

Table 9　PD and PID results on different SAE algorithms

(%)

数据集	SPSAE	ESGSAE-FF	GSTAE	SGAE	DSJPE-ESAE
PD	64.22	66.72	73.89	63.88	75.98
PID	78.76	72.27	77.81	73.29	84.06

从表 9 中可以看到,针对 PD 语音诊断数据集,DSJPE-ESAE 算法达到了最高的准确率(75.98%),相较于传统 SAE 模型(64.22%),提高了 11.76 个百分点。对于 PID 数据集,DSJPE-ESAE 模型的表现最为突出,准确率达到 84.06%,较其他算法均有显著优势。这些结果充分证明了本研究提出的自动编码器模型在实际医疗诊断中是高效的。相较于传统自动编码器方法,效果显著提升,尤其在准确性和诊断能力上表现得更为突出。

4　结　　论

SAE 作为一种代表性的深度网络,具有诸多优点,被广泛应用在工业设备的故障检测、图像去噪、高维基因数据分析、音频数据的降维等实际场景。DSJPE-ESAE 是一种改进的 SAE,也可以应用于这些场景,并且在大多数情况下已被验证优于 SAE。由于 DSJPE-ESAE 是建立在样本集关联信息上的双级深度结构,因此具有更高的准确性,并且采用并联方式可以大大降低时间代价,它适用于对准确率要求高、数据样本分布复杂、样本间关联性强的实际场景中。以本研究实验为例,DSJPE-ESAE 在 LSVT、Statlog 及 heart 等数据集上比现有的 SAE 及其改进的算法的优势更为明显,可能的的原因是这些数据集的统计分布更为复杂以及其样本间的关联信息更为明显。实验结果表明,DSJPE-ESAE 统计上具有最高的准确率,合适的时间代价,较好的稳定性,最优的综合性能。以 Pendigits 数据集为例,在 ACC、F1、AUC 和 AP 值方面,DSJPE-ESAE 比性能最好的 SAE 分别高出 0.69%、0.19%、0.46%和 0.35%,比性能差的 SAE 算法分别高出 24.37%、26.35%、31.11%和 17.28%。

总之,本研究的主要贡献和创新是提出了一种与现有 SAE 本质不同的新结构新范式,这体现在:1)机制不同。现有 SAE 是针对样本个体进行深度特征提取,而 DSJPE-ESAE 是针对样本间关联信息进行深度特征提取。2)结构不同。现有 SAE 是单输入(原样本)单输出(深度特征),而 DSJPE-ESAE 是双输入(两种包络样本)双输出(两种深度特征)。3)功能不同。现有 SAE 仅实现了深度特征变换,而 DSJPE-ESAE 实现了样本与特征的协同深度变换。4)性能不同。比较结果表明,DSJPE-ESAE 具有最好的综合性能。

未来工作主要包括以下两个方面:一是考虑其他的样本变换方法或者其他的聚类方法来重构样本,从而进一步改进本文模型;二是考虑尝试更多的数据集来验证本文模型的有效性。

参考文献

[1] BENGIO Y, COURVILLE A, VINCENT P, et al. Representation learning: A review and new perspectives[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(8): 1798-1828.

[2] FERLES C, PAPANIKOLAOU Y, NAIDOO K J. Denoising autoencoder self-organizing map (DASOM)[J]. Neural Networks, 2018, 105: 112-131.

[3] LEI Y, YUAN W, WANG H P, et al. A skin segmentation algorithm based on stacked autoencoders[J]. IEEE Transactions on Multimedia, 2017, 19(4): 740-749.

[4] POTAPOV A, POTAPOVA V, PETERSON M. A feasibility study of an autoencoder meta-model for improving generalization capabilities on training sets of small sizes[J]. Pattern Recognition Letters, 2016, 80: 24-29.

[5] WANG L, ZHANG Z J, CHEN J Q. Short-term electricity price forecasting with stacked denoising autoencoders[J]. IEEE Transactions on Power Systems, 2017, 32(4): 2673-2681.

[6] DIZAJI K G, HERANDI A, DENG C, et al. Deep c-lustering via joint convolutional autoencoder embeddi-ng and relative entropy minimization [C]. 2017 IEEE International Conference on Computer Vision (ICCV), 2017: 5747-5756.

[7] 陈瑞娟,戚昊峰,李炳南, 等. 基于栈式自编码器的磁探测电阻抗成像算法研究[J]. 仪器仪表学报,2019, 40(1):257-264.
CHEN R J, QI H F, LI B N, et al. Study on magnetic detection electrical impedance tomography algorithm

based on stacked auto-encoder[J]. Chinese Journal of Scientific Instrument, 2019,40(1):257-264.

[8] 李晓彬,牛玉广,葛维春, 等. 基于改进堆叠自编码网络的电站辅机故障预警[J]. 仪器仪表学报,2019,40(6):39-47.
LI X B, NIU Y G, GE W CH, et al. Early fault warning of power plant auxiliary engine based on improved stacked autoencoder network[J]. Chinese Journal of Scientific Instrument, 2019, 40(6):39-47.

[9] YUAN X F, HUANG B, WANG Y L, et al. Deep learning-based feature representation and its application for soft sensor modeling with variable-wise weighted SAE[C]. IEEE Transactions on Industrial Informatics, 2018,14(7): 3235-3243.

[10] 李荣雨,徐宏宇. 平行堆栈式自编码器及其在过程建模中的应用[J]. 电子测量与仪器学报,2017,31(2): 264-271.
LI R Y, XU H Y. Parallel stacked autoencoder and its application in process modeling[J]. Journal of Electronic Measurement and Instrumentation, 2017, 31(2): 264-271.

[11] 童子滔,张治中,张涛, 等. 基于零样本学习和自编码器的调制信号识别研究[J]. 电子测量技术,2024,47(14):1-9.
TONG Z T, ZHANG ZH ZH, ZHANG T, et al. Zero shot learning and autoencoder based modulation signal recognition[J]. Electronic Measurement Technology, 2024,47(14):1-9.

[12] GÖRGEL P, SIMSEK A. Face recognition via deep stacked denoising sparse autoencoders (DSDSA)[J]. Applied Mathematics and Computation, 2019, 355: 325-342.

[13] ZHU H P, CHENG J X, ZHANG C, et al. Stacked pruning sparse denoising autoencoder based intelligent fault diagnosis of rolling bearings[J]. Applied Soft Computing, 2020, 88: 106060.

[14] RAULF A P, BUTKE J, KÜPPER C, et al. Deep representation learning for domain adaptable classifica-tion of infrared spectral imaging data[J]. Bioinformatics, 2020, 36(1): 287-294.

[15] SHAO H D, XIA M, WAN J F, et al. Modified stacked autoencoder using adaptive Morlet wavelet for intelligent fault diagnosis of rotating machinery[J]. IEEE/ASME Transactions on Mechatronics, 2022, 27(1): 24-33.

[16] NOURI A, SEYYEDSALEHI S A. Eigen value based loss function for training attractors in iterated autoencoders[J]. Neural Networks, 2023, 161: 575-588.

[17] YANG Y, ZHAO X F, ZHAO L. Research on asymmetrical edge tool wear prediction in milling TC4 titanium alloy using deep learning[J]. Measurement, 2022, 203: 111814.

[18] YANG D SH, QIN J, PANG Y H, et al. A novel double-stacked autoencoder for power transformers DGA signals with an imbalanced data structure[J]. IEEE Transactions on Industrial Electronics, 2022, 69(2): 1977-1987.

[19] SUN Q Q, GE ZH Q. Gated stacked target-related autoencoder: A novel deep feature extraction and layerwise ensemble method for industrial soft sensor application[J]. IEEE Transactions on Cybernetics, 2022, 52(5): 3457-3468.

[20] OU CH, ZHU H Q, SHARDT Y A W, et al. Quality-driven regularization for deep learning networks and its application to industrial soft sensors[J]. IEEE Transactions on Neural Networks and Learning Systems, 2022: 1-11.

[21] EL-FIQI H, WANG M, KASMARIK K, et al. Weighted gate layer autoencoders[J]. IEEE Transactions on Cybernetics, 2022, 52(8): 7242-7253.

[22] XIAO SH X, WANG SH P, GUO W ZH. SGAE: Stacked graph autoencoder for deep clustering[J]. IEEE Transactions on Big Data, 2023, 9(1): 254-266.

[23] DAI P L, LUO J T, ZHAO K L, et al. Stacked denoising autoencoder for missing traffic data reconstruction via mobile edge computing[J]. Neural Computing and Applications, 2023, 35(19): 14259-14274.

[24] 付晓,沈远彤,付丽华, 等. 基于特征聚类的稀疏自编码快速算法[J]. 电子学报,2018,46(5):1041-1046.
FU X, SEHNG Y T, FU L H, et al. An optimized sparse auto-encoder network based on feature clustering[J]. Acta Electronica Sinica, 2018,46(5):1041-1046.

[25] LIN Y X, CHEN S C. A centroid auto-fused hierarchical fuzzy C-means clustering[J]. IEEE Transactions on Fuzzy Systems, 2021, 29(7): 2006-2017.

[26] 王雪松,张翰林,程玉虎. 基于自编码器和超图的半监督宽度学习系统[J]. 电子学报,2022,50(3):533-539.
WANG X S, ZHANG H L, CHEN Y H, et al. Autoencoder and hypergraph-based semi-supervised broad learning system [J]. Acta Electronica Sinica, 2022, 50(3):533-539.

[27] ZHU H P, CHENG J X, ZHANG C, et al. Stacked pruning sparse denoising autoencoder based intelligent fault diagnosis of rolling bearings [J]. Applied Soft Computing, 2020, 88: 106060.

[28] LI Y M, LEI Y, WANG P, et al. Embedded stacked group sparse autoencoder ensemble with L1 regularization and manifold reduction [J]. Applied Soft Computing, 2021, 101:107003.

[29] SUN Q Q, GE ZH Q. Gated stacked target-related autoencoder: A novel deep feature extraction and layerwise ensemble method for industrial soft sensor application [J]. IEEE Transactions on Cybernetics, 2022, 52(5):3457-3468.

作者简介

李勇明(通信作者),1999 年于电子科技大学获得学士学位,2003 年于重庆大学获得硕士学位,2007 年于重庆大学获得博士学位,现为重庆大学教授,主要研究方向为智能信息处理。

E-mail:yongmingli@cqu.edu.cn

Li Yongming (Corresponding author) received his B. Sc. degree from University of Electronic Science and Technology of China in 1999, received his M. Sc. degree from Chongqing University in 2003, received his Ph. D. degree from Chongqing University in 2007. Now he is a professor at Chongqing University. His main research interest includes intelligent information processing.