Multi-source unsupervised soft sensor based on joint distribution alignment and mapping structure preservation

Zhang, Zheming; Yan, Gaowei; Qiao, Tiezhu; Fang, Yaling; Pang, Yusong

DOI
10.1016/j.jprocont.2021.11.009

Publication date
2022

Document Version
Accepted author manuscript

Published in
Journal of Process Control

Multi-source unsupervised soft sensor based on joint distribution alignment and mapping structure preservation

Zheming Zhang[a], Gaowei Yan[a,*], Tiezhu Qiao[b], Yaling Fang[a], Yusong Pang[c]

[a] *College of Electrical and Power Engineering, Taiyuan University of Technology, Taiyuan 030024, China*
[b] *Key Laboratory of Advanced Transducers and Intelligent Control System, Ministry of Education, Taiyuan University of Technology, Taiyuan 030024, China*
[c] *Faculty of Mechanical, Maritime and Materials Engineering, Delft University of Technology, Delft 2628CD, Netherlands*

Abstract

Aiming at the problem of mismatch between real-time data distribution and modeling data distribution caused by the change of working conditions in industrial process, which leads to the performance deterioration of the soft sensor model, a multi-source unsupervised soft sensor method based on joint distribution alignment and mapping structure preservation is proposed. Firstly, the method uses the hypergraph to establish the complex structure of feature and label, and clusters the hypergraph matrix in multiple views to completely construct the class pseudo label; then dynamic distribution alignment is used to adapt marginal distribution and conditional distribution between the data of historical working conditions and the current working conditions, and the hypergraph Laplacian operator is introduced for manifold regularization to prevent the mapping relationship between feature and label from being destroyed; finally, similar working conditions are introduced to further enhance the robustness of the model. The experimental results show that compared with the traditional unsupervised soft sensor methods, the method used in this paper can effectively improve the prediction accuracy of the model.

Keywords: Soft sensor, Hypergraph, Multi-view clustering, Dynamic distribution alignment

1. Introduction

With the increasing requirements for control, monitoring and operational reliability in industrial processes, real-time monitoring of the key variables has become particularly important. However, factors such as process mechanism, physical environment and characteristics of instrument hardware often make it difficult to directly measure process parameters with sensors, which will affect process monitoring and automatic control. Soft sensor [1–3] has become an effective solution to the above-mentioned problems. It adopts the idea of indirect measurement and establishes a model to estimate the main variables through auxiliary process information. At present, soft sensor methods can be divided into two classes:

*Corresponding author
Email address: yangaowei@tyut.edu.cn (Gaowei Yan)

modeling methods based on process mechanism analysis and data-driven. The process mechanism model is easily affected by many factors such as changes in application environment. At the same time, there are many disturbance factors in the actual industrial process, such as nonlinearity, time-varying and large hysteresis. There are a large number of differential processes in the mechanism model constructed, which lead to problems such as complex solutions and difficulty in obtaining measured values in real time. The data-driven modeling methods rely on the internal connection of data in the process, so there is no need to deeply understand the research object. This method solves the measurement of key parameters in practical engineering problems, and is suitable for modeling applications in the process industry.

At present, data-driven soft sensor methods mainly include multivariate statistical methods represented by partial least squares and principal component analysis, and machine learning methods represented by support vector machines and neural networks. However, the premise of these methods is that the modeling data and real-time data must satisfy the same probability distribution. In the actual production process, due to some situations in the production process such as equipment reorganization, material or environmental changes, production conditions will change significantly. The production system presents the characteristics of multiple working conditions and multiple modes [4, 5], resulting in a distribution mismatch between real-time data and modeling data, causing the original soft sensor model to be inaccurate. And because of the lack of actual sensor data, it is impossible to form an effective mark value of modeling, so it is difficult to establish an accurate soft sensor model after the working conditions change.

Transfer learning solves the problem which is difficult to establish a machine learning model in the target domain due to changes in data distribution and lack of labeled data by transferring the model or parameters of the source domain. It provides new ideas and methods for soft sensor modeling under multiple working conditions. It uses known modal data as the source domain and unknown modal data as the target domain for transfer prediction. Gretton et al. [6] used the maximum mean discrepancy (MMD) to measure the data distribution difference between source domain and target domain, and then reduce the distribution distance between them to achieve the purpose of domain adaptation. However, MMD is mainly used for marginal distribution adaptation, and cannot perform joint adaptation of conditional distribution, thus losing the relationship between feature and label. Therefore, Long et al. [7] used the joint distribution adaptation (JDA) algorithm to match marginal distribution and conditional distribution of the source and target domain data during the transfer process, thereby reducing the overall distribution difference. However, JDA assumes that marginal distribution and conditional distribution are equally important, and this assumption may not be applicable in actual situations. Wang et al. [8] proposed a manifold embedded distribution alignment (MEDA), by introducing a balance factor to weigh the importance of marginal distribution and conditional distribution

in domain adaptation. According to the actual situation, the marginal distribution and conditional distribution are assigned different weights to improve the performance of joint distribution adaptation. But this method is mainly used to solve the classification problem. In the regression problem studied in this paper, the continuous characteristic of data will not cause the MMD matrix to change during the process of adjusting data distribution, and thus it is impossible to directly use MEDA to perform joint distribution adaptation.

Therefore, a classification framework is needed to solve the problem of joint distribution adaptation in soft sensor modeling. However, the compactness criterion of the data contained in the classification problem can make the original class more distinguishable in new space through multiple iterations. But the regression data does not have this characteristic, and only discretizing the data to obtain label may not suitable for regression problems. At the same time, it should be noted that, unlike the classification problem, the regression problem focuses on the internal connection between the feature and label. Therefore, this paper intends to adopt the method of multi-view joint clustering, which uses the feature and label as the two views of the working condition data. In the process of mapping to the low dimensional space, the information between the two views is combined to preserve mapping relationship with feature and label.

At the same time, the hypergraph can more completely express the complex relationships between research objects and capture the deep connections between features and labels than simple graphs. It can better describe its internal overall structure and enhance the effect of multi-view clustering. Hypergraph combined with manifold regularization can reduce the damage to the data structure due to the compactness criterion to a certain extent.

In conclusion, a multi-source unsupervised soft sensor method based on joint distribution alignment and mapping structure preservation (DASP) is proposed, to solve the problem that the distribution of modeling data and real-time data is not consistent due to the multi-mode of the system. This work makes the following contributions: (1) DASP constructs pseudo label to dynamically distribute adapt the continuous regression data and keeps the internal structure of the data in the new projection space. (2) A multi-view classification pseudo label construction method based on hypergraph is proposed, the new label can retain the mapping relationship between the original data feature and label. (3) The experimental results of the ball mill load parameters and Tennessee Eastman (TE) process verify the effectiveness of the method.

The rest of this article is organized as follows. Section 2 introduces related work. Section 3 introduces the soft sensor model based on our method. Section 4 takes TE and wet ball mill experiments as example to verify the effectiveness of our method. Section 5 draws a conclusion.

2. Related work

There are two main difficulties in the soft sensor of multi-conditions processes. The first difficulty is how to deal with the differences between different condition. Because traditional multivariate statistical methods are difficult to deal with the differences between different modes in a model, some researchers use mixed models for modeling, integrating pattern recognition and regression into one model, so as to avoid switching predictive models when data patterns change. Ge et al. [9] extended the principal component regression model to form a mixed probabilistic regression model for soft sensor modeling, used the expectation maximization algorithm to solve the parameters of the mixed probability model, and calculated each type of new data sample for the posterior probability in the operating mode, the combined model gave the estimated result. However, the principal component regression modeling process assumed that the modeling variables were subject to Gaussian distribution, while this was not the case in actual industrial processes. Mei et al. [10] used Gaussian mixture model for regression, set up several Gaussian models to fit the distribution, and directly fused the predicted output of the Gaussian model as the final output. However, the problem of determining the number of Gaussian models that fit the distribution needed to be optimized. Tan et al. [11] used local nearest neighbor standardization to Gaussian processing of the original data and established a partial least square (PLS) model for fault detection in a multi-condition process. However, the above method also requires the labeled data in each condition when modeling, which is unrealistic in the actual process. And when the working conditions are frequently changed, the global model cannot effectively track the changes in the dynamic characteristics of the industrial process, resulting in a decrease in predictive ability. The method based on multi-model matching once the model is mismatched, it will have a greater impact on system monitoring [12].

Another difficulty in soft sensor modeling for multi-modal processes is the update of the system model to cope with the conceptual drift in the process, so that the model has the ability to adapt to unknown operating conditions. Recursive Modeling/Moving Window (MW) and Just-in-time Learning (JITL) are commonly used adaptive (online) learning tools to deal with concept drift in the industrial process [13]. The recursive iteration method/moving window uses the sample closest to the query point time in the historical data segment for modeling. However, in the case of large process drift, the established model is difficult to track the process dynamics that occur in the new data. Just-in-time learning selects the sample set most relevant to the current sample from the marked historical data according to the similarity metric to establish a real-time regression model. However, the established model is susceptible to the influence of different similarity measurement criteria, and when a new working condition appears, historical data that matches it cannot be found. A process may experience various types and frequencies of operating condition changes during its operation. Therefore, it is unreasonable to expect a single MW or JITL model to be effective for a long time. The most popular method to

solve this problem is ensemble learning (EL), in which models built for different working conditions are adaptively combined to predict query points [14]. Ensemble learning builds multiple sub-models of historical data, evaluates the prediction results of each sub-model, and weights and fuses the multiple sub-models according to the confidence level of the model output, and finally obtains the ensemble regression model. The ensemble learning strategy balances the diversity of process data by establishing multiple local models, which is essential to offset the changes in operating conditions of different types and rates in industrial processes. However, due to the need to build multiple local models, the amount of calculation in the training process will increase exponentially in the case of large data sets.

None of the above methods substantially eliminates the impact of data distribution differences on modeling under multiple working conditions. At the same time, the above-mentioned adaptive real-time modeling methods all assume that the real label can be obtained under a certain delay, which is unrealistic in some actual processes. Transfer learning aims to reduce the distribution difference between the source domain and the target domain, so that the knowledge obtained from the source domain can be used to help improve the learning of the prediction function in the target domain. Regarding the historical working condition in the multi-condition problem as the source domain and the current working condition as the target domain, constructing a transfer learning model that migrates from the historical working condition to the current working condition provides a solution to the soft sensing problem under multiple working conditions [15]. It is important to note that unsupervised transfer learning provides a distribution alignment framework that does not require target working condition label values, Zheng et al. [16] designed a multisource-Refined transfer network based on unsupervised transfer learning for unsupervised cross-domain fault diagnosis. So the use of transfer learning based multi-modal soft sensor methods to solve this problem has become a hot spot in current researches. It is assumed that there is a shared latent feature space between the source domain and target domain to reduce the existing distribution differences between domains. The strategy to find such a shared latent feature space is to adopt a dimensionality reduction method and minimize some predefined distance measurements to reduce the marginal distribution or conditional distribution mismatch between the source domain and target domain. In order to match the marginal distribution, Chen et al. [17] introduced two subspace distribution adaptation frameworks. Both frameworks use the subspace distribution adaptation function to make source distribution similar to target distribution, and at the same time learn the adaptive classifier through the principle of structural risk minimization. Kumagai et al. [18] transformed the source feature representation through a linear matrix function, so that the source distribution and target distribution are similar under the MMD distance. Pan et al. [19] proposed the transfer component analysis (TCA) algorithm, which projected the source and target domain data into a high dimensional Hilbert space, minimized the distance between source domain and target domain instead of only modifying the source distribution, while reducing the difference between source domain

and target domain distribution, retained their respective internal attributes to the greatest extent. Kan et al. [20] proposed the target source domain (TSD) algorithm. While preserved the data structure, constructed the projection matrix of the conversion to reduce the distribution difference between source domain and target domain in the subspace. However, the marginal distribution adaptation would lose the role of the label in the distribution adaptation. Therefore in [21–23], the author explored matching the marginal distribution and the class conditional distribution at the same time to enhance the effect of the label. Roughly speaking, most of these works are based on the joint distributed adaptation method. The above methods are very likely to cause damage to the data structure in the process of adjusting distribution and adaptation. Du et al. [24] introduced the idea of manifold regularization to reduce the distribution difference between source domain and target domain, while preserved the local feature information, thereby reducing the structural drift of the two in the process of projecting into the subspace. However, the above methods are all used for the classification problem. Applied in the field of soft sensor, the continuous distribution of the regression data itself is different from the compactness structure of the classification, which will cause greater damage to the data structure. Therefore, this paper uses the method of multi-view clustering to combine the feature and label data to construct the process pseudo label so that it retains the mapping relationship between feature and label.

Since different features can be extracted to describe a sample, multi-view learning can capture the internal associations between multiple views, thereby improving learning performance [25]. The success of multi-view learning lies in its principles of consistency and complementarity, which can well characterize the relationship between multiple views. In recent years, multi-view clustering mostly combines data from different views into a single view representation before data clustering. Guo et al. [26] described multi-view subspace learning as a joint optimization problem, which has a common subspace representation matrix and group sparsity inducing norm. White et al. [27] learned a common expression based on multiple views in a targeted manner, and solved a joint optimization problem through a common subspace representation matrix. Lu et al. [28] tried to find low dimensional embedding of the data by calculating the eigenvectors of the standardized Laplacian matrix, so as to use lower dimensional representation methods to solve the problem that is difficult to calculate in the high dimensional space. Brbicet al. [29] proposed a multi-view low-rank sparse subspace clustering method. This method learns joint subspace representations by constructing an association matrix shared between views, and then used spectral clustering to process multi-view data. This method combines the feature information of multiple different views and divides similar samples into the same group in an attempt to obtain a more accurate cluster assignment.

At the same time, when the research object has a paired relationship, it can be represented by a graph. However, in many practical problems, the relationship between objects is much more complicated than the pair-wise relationship. Simply compressing complex relationships into pairwise relationships

will inevitably lead to the loss of information. Therefore, consider using hypergraph [30] to completely represent the complex relationship between the research objects. Agarwal et al. [31] proposed the use of hypergraph to construct Laplacian matrix, and developed a general framework for classification and clustering of complex relational data. Wang et al. [32] proposed hypergraph canonical correlation analysis. This method is based on canonical correlation analysis and considers high-level label structure information through hypergraph regularization.

It is also noted that the utilization of multiple source domains is an opportunity to further improve the model performance by extracting more useful information. Liu et al. [33] used a novel framework of an adversarial transfer learning (ATL)-based soft sensing method which was designed for the quality inferring of multigrade processes. Treating each grade as a domain, the concept of ATL was adopted to learn a suitable feature transformation between different domains, which reduces the data distribution discrepancy. As a supervised soft sensing method, the labeled target domain data is often difficult to obtain. Therefore, this method uses the MMD similarity measure to select the two domains closest to the target domain from multiple source domains to build models and carry out weighted integration, so as to improve the prediction accuracy and enhance the robustness of the model.

To sum up, in order to solve the problem that the regression data cannot be adapted to the joint distribution of the multi-condition soft sensor modeling, this paper uses multi-view clustering to establish class pseudo label with the known working conditions, and hypergraph can be used to describe feature of deep structure of data to construct the view matrix, which made the multi-view clustering result more reliable. In addition, the Laplacian similarity matrix is constructed through the hypergraph, and the manifold regularization constraint is performed to keep the data structure during the projection process. The algorithm diagram is shown in Figure 1.

3. Related theories and algorithms

Throughout this paper, matrices are represented with bold capital symbols and vectors with bold lower-case symbols. For matrix $\mathbf{X} = (x_{ij})$, the row i is denoted as $\mathbf{x}_i$, and the column j is denoted as $\mathbf{x}_j$. Given the feature $\mathbf{X}_s = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n]^{\mathrm{T}} \in R^{n\times r}$ and label $\mathbf{Y}_s \in R^{n\times 1}$ of the historical working condition (source domain) $\mathcal{D}_s$ and the feature $\mathbf{X}_t = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m]^{\mathrm{T}} \in R^{m\times r}$ of the current working condition (target domain) $\mathcal{D}_t$, where n, m is the number of data samples and r is the dimension of the sample feature vector. $\mathbf{X}_s^c = [\mathbf{x}_1^c, \mathbf{x}_2^c, \cdots, \mathbf{x}_n^c]^{\mathrm{T}}$, $\mathbf{X}_t^c = [\mathbf{x}_1^c, \mathbf{x}_2^c, \cdots, \mathbf{x}_m^c]^{\mathrm{T}}$ and $\mathbf{Y}_s^c$ represent the feature and label after clustering, respectively.

3.1. Dynamic distribution alignment

In the soft sensor model, due to process differences, the distribution of real-time data and modeling data among multiple working conditions will be inconsistent, which does not satisfy the assumption of

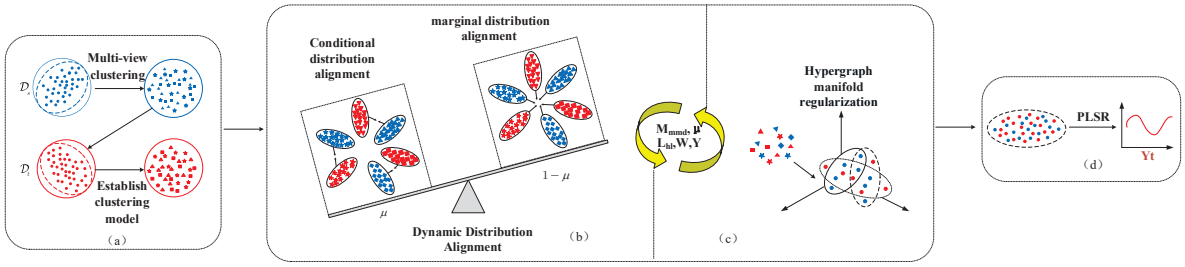


Figure 1: Schematic diagram of the algorithm. (a) Perform multi-view clustering through labeled source domain data to obtain class pseudo label and establish a clustering model, such as KNN, to predict feature of unlabeled target domain; (b) Condition and edge of two domain data distribute adaptation and dynamically assign weights; (c) In the process of domain adaptation, the internal complex structure of the data is constrained by hypergraph Laplacian regularization. In the figure, blue represents the source domain data, and red represents the target domain data. Use triangles, squares, and five-pointed stars to represent different class of data, that is, the process of class pseudo labeling, and circles represent regression data. In the process of (b-c), iteratively update and optimize the MMD matrix, balance factor, hypergraph Laplacian matrix $\mathbf{L}$, projection matrix $\mathbf{W}$ and label matrix $\mathbf{Y}$; (d) The source domain and target domain data distribution after feature transformation is pulled in, finally, PLSR is used to obtain the final predicted label.

the same data distribution. And there are differences in the distribution of feature and label at the same time, resulting in a mismatch between the marginal distribution and conditional distribution. In addition, the degree of difference between two distributions may be different, so an adaptive factor needs to be introduced to weigh the importance of marginal and conditional distribution and adjust them dynamically. The dynamic distribution alignment is defined as [8]:

$$\bar{D}_f(\mathcal{D}_s, \mathcal{D}_t) = (1-\mu) D_f(P_s, P_t) + \mu \sum_{c=1}^{k} D_f^{(c)}(Q_s, Q_t) \tag{1}$$

where $\mu \in [0,1]$ is the balance factor, $c \in [1, \cdots, k]$ is the class indicator. $D_f(P_s, P_t)$ denotes the marginal distribution alignment, and $D_f^{(c)}(Q_s, Q_t)$ denotes the conditional distribution alignment for class c.

The balance factor μ is calculated according to the global and local structure of the domain, that is, a linear classifier is established using a metric to distinguish the error of two domains (i.e. a binary classification), such as $\mathcal{A}$-distance [34]. Therefore, the marginal distribution distance φ_m and conditional distribution distance φ_c can be measured by this method. The estimated value of μ is [8]:

$$\hat{\mu} \approx 1 - \frac{\varphi_m}{\varphi_m + \sum_{c=1}^{k} \varphi_c} \tag{2}$$

This paper uses the MMD to calculate the difference between the two probability distributions, and continuously adjusts the MMD matrix by minimizing the overall difference between the two, so that the constructed projection matrix adaptively distributes the source and target domains data conduct guidance. The dynamic distribution alignment item can be expressed in the form of a matrix as:

$$\bar{D}_f(\mathcal{D}_s, \mathcal{D}_t) = (1-\mu) \left\| \frac{1}{n} \sum_{i=1}^{n} \phi(\mathbf{x}_i) - \frac{1}{m} \sum_{j=1}^{m} \phi(\mathbf{x}_j) \right\|_{\mathcal{H}_K}^2 + \mu \sum_{c=1}^{k} \left\| \frac{1}{n} \sum_{i=1}^{n} \phi(\mathbf{x}_i^c) - \frac{1}{m} \sum_{j=1}^{m} \phi(\mathbf{x}_j^c) \right\|_{\mathcal{H}_K}^2 \tag{3}$$

where $\phi(\cdot)$ represents the transformation of the sample in the reproducing kernel Hilbert space $\mathcal{H}_K$.

The marginal distribution alignment item is:

$$\begin{aligned} D_f(P_s, P_t) &= \left\| \frac{1}{n} \left(\mathbf{W}^{\mathrm{T}} \mathbf{K}_1 + \ldots + \mathbf{W}^{\mathrm{T}} \mathbf{K}_n \right) - \frac{1}{m} \left(\mathbf{W}^{\mathrm{T}} \mathbf{K}_{n+1} + \ldots + \mathbf{W}^{\mathbf{T}} \mathbf{K}_{n+m} \right) \right\|_{\mathcal{H}_K}^2 \\ &= \mathrm{tr} \left(\mathbf{W}^{\mathrm{T}} \left(\frac{1}{n} \mathbf{K}_s \mathbf{1}_{n \times 1} - \frac{1}{m} \mathbf{K}_t \mathbf{1}_{m \times 1} \right) \left(\mathbf{W}^{\mathrm{T}} \left(\frac{1}{n} \mathbf{K}_s \mathbf{1}_{n \times 1} - \frac{1}{m} \mathbf{K}_t \mathbf{1}_{m \times 1} \right) \right)^{\mathrm{T}} \right) \\ &= \mathrm{tr}(\mathbf{W}^{\mathrm{T}} \mathbf{K} \mathbf{M}_0 \mathbf{K} \mathbf{W}) \end{aligned} \tag{4}$$

where $\mathbf{K}_s = (\mathbf{K}_1, \ldots, \mathbf{K}_n)$ and $\mathbf{K}_t = (\mathbf{K}_{n+1}, \ldots, \mathbf{K}_{n+m})$ are the kernel matrices of source and target domains, respectively, $\mathbf{K} = [\mathbf{K}_s, \mathbf{K}_t] \in R^{(n+m)\times(n+m)}$.The projection matrix is $\mathbf{W} \in R^{(n+m)\times k}$, $\mathrm{tr}(\cdot)$ represents the trace of the matrix. In the same way, the conditional distribution alignment item is:

$$\begin{aligned} D_f^{(c)}(Q_s, Q_t) &= \sum_{c=1}^{k} \mathrm{tr} \left(\left(\mathbf{W}^{\mathrm{T}} \left(\frac{1}{n^{(c)}} \mathbf{K}_s^{(c)} \mathbf{1}_{n^{(c)} \times 1} - \frac{1}{m^{(c)}} \mathbf{K}_t^{(c)} \mathbf{1}_{m^{(c)} \times 1} \right) \right) \right. \\ &\quad \left. \times \left(\mathbf{W}^{\mathrm{T}} \left(\frac{1}{n^{(c)}} \mathbf{K}_s^{(c)} \mathbf{1}_{n^{(c)} \times 1} - \frac{1}{m^{(c)}} \mathbf{K}_t^{(c)} \mathbf{1}_{m^{(c)} \times 1} \right) \right)^{\mathrm{T}} \right) \\ &= \sum_{c=1}^{k} \left(\mathrm{tr}(\mathbf{W}^{\mathrm{T}} \mathbf{K} \mathbf{M}_c \mathbf{K} \mathbf{W}) \right) \end{aligned} \tag{5}$$

Therefore, the dynamic distribution alignment item can be expressed as:

$$\mathrm{tr}(\mathbf{W}^{\mathrm{T}} \mathbf{K} \mathbf{M} \mathbf{K} \mathbf{W}) \tag{6}$$

where $\mathbf{M}$ is the MMD matrix, expressed as:

$$\mathbf{M} = (1-\mu)\mathbf{M}_0 + \mu \sum\nolimits_{c=1}^{k} \mathbf{M}_c \tag{7}$$

where $\mathbf{M}_0$ represents the marginal distribution matrix, $\mathbf{M}_c$ represents the conditional distribution matrix, $\mathbf{M}_0$ and $\mathbf{M}_c$ are constructed as follows:

$$(\mathbf{M}_0)_{ij} = \begin{cases} \frac{1}{n^2}, & \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_s \\ \frac{1}{m^2}, & \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_t \\ -\frac{1}{mn}, & otherwise \end{cases} \tag{8}$$

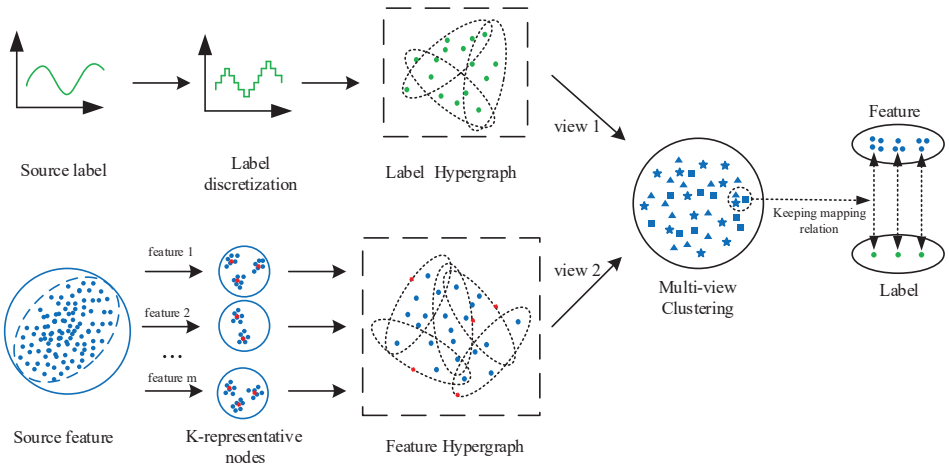


Figure 2: Schematic diagram of pseudo label structure

$$(\mathbf{M}_c)_{ij} = \begin{cases} \frac{1}{n_c^2}, & \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_s^{(c)} \\ \frac{1}{m_c^2}, & \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_t^{(c)} \\ -\frac{1}{m_c n_c}, & \begin{cases} \mathbf{x}_i \in \mathcal{D}_s^{(c)}, \mathbf{x}_j \in \mathcal{D}_t^{(c)} \\ \mathbf{x}_i \in \mathcal{D}_t^{(c)}, \mathbf{x}_j \in \mathcal{D}_s^{(c)} \end{cases} \\ 0, & otherwise \end{cases} \tag{9}$$

where $\mathcal{D}_s^{(c)}$ and $\mathcal{D}_s^{(c)}$ denote samples from class c in $\mathcal{D}_s$ and $\mathcal{D}_t$, respectively, and $n_c = \left|\mathcal{D}_s^{(c)}\right|, m_c = \left|\mathcal{D}_t^{(c)}\right|$.

3.2. Acquisition of category pseudo labels in regression problems

Dynamic distribution alignment is mainly proposed for classification problems. When facing soft sensor regression problems, conditional distribution adaptation cannot be performed directly, and class pseudo label need to be obtained firstly. However, the traditional clustering method cannot fully express the information association between the original data: it only considers the feature or label of data, and ignores the internal connection between feature and label. In other words, if only using feature, it will lose the guiding role of the label; if only use the label, it will lose the relationship between feature and label. In response to such problem, this paper uses the hypergraph based multi-view method to construct source domain pseudo label: firstly cluster the data to obtain the internal structure relationship of data, and construct its hypergraph matrix; then, use the hypergraph matrix as its view matrix. Using the method of multi-view clustering, label and feature are used as views representing two opposite directions of the data structure, which act on the whole clustering process. Applying the hypergraph matrix to multi-view clustering can more effectively express the internal structure of each view and promote data association between multiple views. The schematic diagram of pseudo label structure is shown in Figure 2 .

3.2.1. Hypergraph construction

Generally, in a simple graph, the connecting edges between nodes can only reflect a certain relationship that exists between these two nodes. However, the hyperedge in a hypergraph can include any number of nodes, which can reflect the relationship between multiple nodes, so the hypergraph can represent the complex relationship between objects. For soft sensor, the data collected in the industrial process is used to represent the information transmitted by multiple sensors over time, such as liquid level, pressure, temperature, etc. which are the physical meaning of the feature. With the development of industrial process, the change of feature information often does not proceed simultaneously. Therefore, at a certain process point, the ways that different features affect are different. In the process of feature change, due to the setting of the threshold, the value will fluctuate within this range. When the threshold is exceeded, the feature is considered to have entered a new working state. Therefore, at the same time, different features may be in different working states, which lead to the clustering of continuous data, and the same sample will be divided into different class. Therefore, in the hypergraph constructed by clustering the regression data, each sample represents a vertex, and a working state is a hyperedge. The hypergraph can be used to obtain a variety of state information contained in the different feature of the working condition, so the knowledge structure between data collected under different conditions can be better expressed and the robustness of model can be enhanced.

If the finite set of vertices $\mathbf{V}$ and the set of edges $\mathbf{E}$ satisfy $U_{\mathbf{e}\in\mathbf{E}} = \mathbf{V}$, then a hypergraph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ can be constructed. If each hyperedge $\mathbf{e}$ is associated with a positive weight $\psi(\mathbf{e})$, it is called a weighted hypergraph $\mathbf{G}$. For a hyperedge $\mathbf{e} \in \mathbf{E}$, the number of vertices is its degree, namely $\delta(\mathbf{e}) = |\mathbf{e}|$. For a vertex $\mathbf{v} = \mathbf{V}$, its degree is defined as [32]:

$$d(\mathbf{v}) = \sum_{\mathbf{v}\in\mathbf{e},\mathbf{e}\in\mathbf{E}} \psi(\mathbf{e}) \tag{10}$$

The hypergraph $\mathbf{G}$ can be represented by the incidence matrix of vertices and edges as:

$$\mathbf{H}(\mathbf{v},\mathbf{e}) = \begin{cases} h(\mathbf{v},\mathbf{e}) = 1; & \mathbf{v} \in \mathbf{e} \\ h(\mathbf{v},\mathbf{e}) = 0; & otherwise \end{cases} \tag{11}$$

The essence of graph-based or hypergraph-based methods is to discover the underlying structure of the data set. Therefore, it is necessary to reduce the number of hyperedge while preserving the original structure. For this reason, this paper uses the clustering method to generate the centroid as the most representative data point in the data set, and iteratively makes this point have a strong representation ability and can fully cover the data set. If the same number of hyperedges is used to represent the hypergraph, using a centroid to generate the hyperedges is better than other methods. This method can keep the integrity of the data set structure to the maximum. In order to achieve this goal, this paper uses a general clustering method, such as the k-means method.

Considering the interrelationship between the high dimensional feature samples of the data set, all samples of each dimension feature are clustered to construct a feature hypergraph matrix. Each hyperedge is composed of a sample and all other samples that belong to the same centroid. Since each sample will belong to multiple class at the same time (it is assumed that the importance of each class is the same, which is, the weight of the hyperedge is 1), the structural relationship between the data can be established through the hypergraph. Then the feature hypergraph matrix can be expressed as:

$$\begin{aligned} \mathbf{H}_i^f &= \begin{cases} 1; & x_{ij}^c \in c \\ 0; & otherwise \end{cases} \quad j = 1, \cdots, n \\ \mathbf{H}^f &= \left[\mathbf{H}_1^f, \mathbf{H}_2^f, \cdots \mathbf{H}_m^f\right] \end{aligned} \tag{12}$$

where m is the feature dimension, n is the number of samples, $\mathbf{H}_i^f$ is the hypergraph matrix of the i-th dimensional feature, and x_{ij}^c is the class of the j-th sample after the i-th dimensional feature clustering.

At the same time, from the point of view of data structure, the main difference between feature and label is that feature is a set of data composed of multi-dimensional while label can be regarded as single dimensional feature and have a guiding role for feature. With the continuous nature of the label itself, discretizing it into segments can perform clustering more efficiently and obtain the label hypergraph matrix $\mathbf{H}^l$.

3.2.2. Multi-view subspace clustering

Given the feature hypergraph matrix $\mathbf{H}_s^f$ and label hypergraph matrix $\mathbf{H}^l$ constructed by the source domain feature $\mathbf{X}_s^c$ and label $\mathbf{Y}_s^c$ after clustering, they are regarded as the respective view matrix $\mathbf{H} = \left\{\mathbf{H}^f, \mathbf{H}^l\right\}$. Therefore, for a hypergraph matrix with two views, this paper uses a low-rank sparse subspace multi-view clustering (MC) method to map the data from the high dimensional space to the low dimensional subspace, using the linear combination of few bases represents the essential feature of the data, and a joint representation matrix $\mathbf{C}$ is found to weigh the consistency between different views. Need to solve the following problems [29]:

$$\begin{aligned} &\min_{\mathbf{C}} \tfrac{1}{2} \left\|\mathbf{H} - \mathbf{H}\mathbf{C}\right\|_{\mathrm{F}}^2 + \theta_1 \|\mathbf{C}\|_* + \theta_2 \|\mathbf{C}\|_1 \\ &s.t. \quad diag(\mathbf{C}) = 0. \end{aligned} \tag{13}$$

where the kernel norm $\|\cdot\|_*$ is used to approximate the rank of $\mathbf{C}$. Matrix sparsity requires that each simple is represented by a small number of data points in its own subspace. The ℓ_1 norm is used as the tightest convex relaxation of the ℓ_0 quasi-norm that counts the number of nonzero elements of the solution. Constraint $diag(\mathbf{C}) = 0$ is used to avoid trivial solution of representing a data point as a linear combination of itself.

In order to solve the problem in equation (13), introducing auxiliary variables $\mathbf{C}_1^{(v)}, \mathbf{C}_2^{(v)}$ and $\mathbf{F}^{(v)}$.

Without considering the influence of noise, the objective function can be expressed as:

$$\begin{aligned}
&\min_{\mathbf{C}_1^{(v)},\mathbf{C}_2^{(v)},\mathbf{F}^{(v)}} \theta_1\left\|\mathbf{C}_1^{(v)}\right\|_* + \theta_2\left\|\mathbf{C}_2^{(v)}\right\|_1 \\
&s.t. \quad \mathbf{H}^{(v)} = \mathbf{H}^{(v)}\mathbf{F}^{(v)}, \mathbf{F}^{(v)} = \mathbf{C}_2^{(v)} - diag(\mathbf{C}_2^{(v)}), \\
&\mathbf{F}^{(v)} = \mathbf{C}_1^{(v)}, v = 1, 2.
\end{aligned} \tag{14}$$

where $\mathbf{C}^{(v)}$ is the representation matrix of the view v. Parameters θ_1, θ_2 are the trade-off coefficients of low-rank and sparsity constraints.

Augmented Lagrangian is:

$$\begin{aligned}
&\mathcal{L}(\left\{\mathbf{C}_i^{(v)}\right\}_{i=1}^2, \mathbf{F}^{(v)}, \left\{\mathbf{\Lambda}_i^{(v)}\right\}_{i=1}^3) = \theta_1\left\|\mathbf{C}_1^{(v)}\right\|_* + \theta_2\left\|\mathbf{C}_2^{(v)}\right\|_1 + \tfrac{\psi_1}{2}\left\|\mathbf{H}^{(v)} - \mathbf{H}^{(v)}\mathbf{F}^{(v)}\right\|_{\mathrm{F}}^2 \\
&+\tfrac{\psi_2}{2}\left\|\mathbf{F}^{(v)} - \mathbf{C}_2^{(v)} + diag(\mathbf{C}_2^{(v)})\right\|_{\mathrm{F}}^2 + \tfrac{\psi_3}{2}\left\|\mathbf{F}^{(v)} - \mathbf{C}_1^{(v)}\right\|_{\mathrm{F}}^2 \\
&+\mathrm{tr}\left[\mathbf{\Lambda}_1^{(v)T}\left(\mathbf{H}^{(v)} - \mathbf{H}^{(v)}\mathbf{F}^{(v)}\right)\right] + \mathrm{tr}\left[\mathbf{\Lambda}_2^{(v)T}\left(\mathbf{F}^{(v)} - \mathbf{C}_2^{(v)} + diag(\mathbf{C}_2^{(v)})\right)\right] \\
&+\mathrm{tr}\left[\mathbf{\Lambda}_3^{(v)T}\left(\mathbf{F}^{(v)} - \mathbf{C}_1^{(v)}\right)\right]
\end{aligned} \tag{15}$$

where $\{\psi_i > 0\}_{i=1}^3$ is the penalty coefficient and $\left\{\mathbf{\Lambda}_i^{(v)}\right\}_{i=1}^3$ is the Lagrangian dual variable. In order to solve the convex optimization problem in the above formula, the Alternating Direction Method of Multipliers (ADMM) [35] can be used to obtain the update formula of each iteration process:

$$\begin{aligned}
\mathbf{F}^{(v)} &= \left[\psi_1 \mathbf{H}^{(v)\mathrm{T}}\mathbf{H}^{(v)} + (\psi_2 + \psi_3)\mathbf{I}\right]^{-1} \\
&\times \left(\psi_1 \mathbf{H}^{(v)\mathrm{T}}\mathbf{H}^{(v)} + \psi_2\mathbf{C}_2^{(v)} + \psi_3\mathbf{C}_1^{(v)} + \mathbf{H}^{(v)\mathrm{T}}\mathbf{\Lambda}_1^{(v)} - \mathbf{\Lambda}_2^{(v)} - \mathbf{\Lambda}_3^{(v)}\right) \\
\mathbf{C}_1^{(v)} &= \Pi_{\frac{\theta_1}{\psi_3}}\left(\mathbf{F}^{(v)} + \frac{\mathbf{\Lambda}_3^{(v)}}{\psi_3}\right) \\
\mathbf{C}_2^{(v)} &= \pi_{\frac{\theta_2}{\psi_2}}\left(\mathbf{F}^{(v)} + \frac{\mathbf{\Lambda}_2^{(v)}}{\psi_2}\right) \\
\mathbf{\Lambda}_1^{(v)} &= \mathbf{\Lambda}_1^{(v)} + \psi_1\left(\mathbf{H}^{(v)} - \mathbf{H}^{(v)}\mathbf{F}^{(v)}\right) \\
\mathbf{\Lambda}_2^{(v)} &= \mathbf{\Lambda}_2^{(v)} + \psi_2\left(\mathbf{F}^{(v)} - \mathbf{C}_2^{(v)}\right) \\
\mathbf{\Lambda}_3^{(v)} &= \mathbf{\Lambda}_3^{(v)} + \psi_3\left(\mathbf{F}^{(v)} - \mathbf{C}_1^{(v)}\right)
\end{aligned} \tag{16}$$

where $\Pi_\theta(\mathbf{\Delta}) = \mathbf{U}\pi_\theta(\mathbf{\Sigma})\mathbf{V}^{\mathrm{T}}$ represents the soft threshold operation on the singular values of $\mathbf{\Delta}$ and $\pi_\theta(\mathbf{\Sigma})$ represents the defined soft threshold operator.

By averaging the elements $\left\{\mathbf{C}^{(1)}, \mathbf{C}^{(2)}\right\}$ and obtaining the matrix $\mathbf{C}$, the adjacency matrix $\mathbf{B}$ can be obtained as:

$$\mathbf{B} = |\mathbf{C}| + |\mathbf{C}|^{\mathrm{T}} \tag{17}$$

Because spectral clustering [36] only needs the similarity matrix between data, it is very effective for processing sparse data clustering. Therefore, the main steps of using spectral clustering to obtain pseudo label $\mathbf{Y}_s^c$ of source domain class are as follows:

1) Obtain the Laplacian matrix $\mathbf{L}_y$ through the adjacency matrix $\mathbf{B}$;

2) Perform eigenvalue decomposition on $\mathbf{L}_y$ and take the eigenvector corresponding to the k smallest eigenvalue;

3) Take the solved eigenvectors (and normalize them respectively) to form a new spectral clustering characteristic matrix $\mathbf{X}^{sc} = [\mathbf{x}_1^{sc}, \mathbf{x}_2^{sc}, \cdots, \mathbf{x}_n^{sc}]^{\mathrm{T}} \in R^{n \times k}$, and k-means clustering of matrix $\mathbf{X}^{sc}$ to obtain pseudo label $\mathbf{Y}_s^c \in {}^{n \times 1}$.

3.3. Hypergraph Manifold Regularization

The pseudo label constructed by the multi-view method can preserve the mapping relationship between the original feature and label, but this relationship between the data may be destroyed in the process of dynamic distribution adaptation. In order to solve this problem, this paper introduces hypergraph manifold regularization (HMR) to constrain the projection matrix, and uses the hypergraph Laplacian to construct data associations between feature and label, so that the data can preserve the deep geometric structure of the original data in the new projected space.

The hypergraph regular item is defined as [30]:

$$R_h(\ell) = \frac{1}{2} \sum_{\mathbf{e} \in \mathbf{E}} \sum_{\mathbf{u}, \mathbf{v} \in \mathbf{V}} \frac{\psi(\mathbf{e})\, h(\mathbf{v}, \mathbf{e})}{\delta(\mathbf{e})} \left(\frac{\ell(\mathbf{u})}{d(\mathbf{u})} - \frac{\ell(\mathbf{v})}{d(\mathbf{v})} \right)^2 \tag{18}$$

Taking the diagonal matrix $\mathbf{D}_v$, $\mathbf{D}_e$ as the degree matrix of the vertices and the hyperedge in the hypergraph, respectively, $\mathbf{Z}_e$ as the weight matrix of the hyperedge, since the weight is 1, this matrix is equivalent to the identity matrix. The Laplacian of the hypergraph is $\mathbf{L} = \mathbf{I} - \mathbf{S}$, where $\mathbf{I}$ is the identity matrix, and the similarity matrix $\mathbf{S}$ can be expressed as [32]:

$$\mathbf{S} = \mathbf{D}_v^{-1/2} \mathbf{H}_j \mathbf{Z}_e \mathbf{D}_e^{-1} \mathbf{H}_j^{\mathrm{T}} \mathbf{D}_v^{-1/2} \tag{19}$$

where $\mathbf{H}_j$ is the joint hypergraph matrix, which can be expressed as:

$$\mathbf{H}_j = \begin{bmatrix} \mathbf{H}_s \\ \mathbf{H}_t \end{bmatrix}, \mathbf{H}_s = \begin{bmatrix} \mathbf{H}_{\mathrm{s}}^f & \mathbf{H}_{\mathrm{s}}^l \end{bmatrix}, \mathbf{H}_t = \begin{bmatrix} \mathbf{H}_t^f & \mathbf{H}_t^l \end{bmatrix} \tag{20}$$

where $\mathbf{H}_s$ and $\mathbf{H}_t$ are the hypergraph matrices of source domain and target domain respectively, and they are obtained by the hypergraph matrices of their respective feature and label. So the regularization expression of manifold based on the hypergraph Laplacian is:

$$\mathrm{tr}\left(\mathbf{W}^{\mathrm{T}} \mathbf{K} \mathbf{L} \mathbf{K} \mathbf{W}\right) \tag{21}$$

4. Algorithm model and optimization solution

DASP is mainly divided into two parts: (1) Obtain class pseudo label through the hypergraph based multi-view clustering method; (2) Use the pseudo label obtained in the first part to perform

dynamic distribution alignment and manifold regularization constraints. At the same time, both parts use iterative methods to optimize and update the requested parameters.

The first part to obtain class pseudo label $\mathbf{Y}_s^c$. The second part of the model needs to integrate the above parts. Due to the large number of parameters, it is easy to cause large model complexity. And the empirical risk optimization strategy believes that the model with the least empirical risk is the optimal model, but using this model may cause over-fitting problems. Therefore, this paper uses the structural risk function to prevent overfitting. Its structural risk is defined as [8]:

$$R_{srm} = \frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(y_i, \hat{y}_i) + \eta J(f_m) \tag{22}$$

where $J(f_m)$ is the model complexity. η is a coefficient, used to weigh empirical risk and model complexity. $\mathcal{L}(y_i, \hat{y}_i)$ is the loss function, y_i is the true value, and $\hat{y}_i$ is the predicted value. This paper adopts the square loss function, which is expressed as:

$$\arg\min_{f_m \in \mathcal{H}_K} \sum_{i=1}^{n} \left(y_i - f_m(x_i)\right)^2 + \eta \left\| f_m \right\|_K^2 \tag{23}$$

where $\mathcal{H}_K$ represents the reproducing kernel Hilbert space. Using the representation theorem [37], it can be extended to:

$$\begin{aligned} f_m(\cdot) &= \sum_{i=1}^{n+m} w_i k(x_i, \cdot) \\ &= (w_1, \ldots, w_{n+m}) \begin{pmatrix} k\left(x_1, \cdot\right) \\ \ldots \\ k\left(x_{n+m}, \cdot\right) \end{pmatrix} \\ &= \mathbf{W}^{\mathrm{T}}\mathbf{K} \end{aligned} \tag{24}$$

Therefore, the structural risk function can be written as:

$$\begin{aligned} \sum_{i=1}^{n+m} \left(y_i - f_m(x_i)\right)^2 + \eta \left\| f_m \right\|_K^2 &= \sum_{i=1}^{n+m} \mathbf{A}_{ii}(y_i - \mathbf{w}^{\mathrm{T}}\mathbf{k}_i))^2 + \eta tr\left(f_m {f_m}^{\mathrm{T}}\right) \\ &= \left\|(\mathbf{Y} - \mathbf{W}^{\mathrm{T}}\mathbf{K})\mathbf{A}\right\|_{\mathrm{F}}^2 + \eta \mathrm{tr}(\mathbf{W}^{\mathrm{T}}\mathbf{K}\mathbf{W}) \end{aligned} \tag{25}$$

where $\|\cdot\|_{\mathrm{F}}$ represents the F norm. $\mathbf{K}_{ij} = \mathbf{K}(\mathbf{x}_i, \mathbf{x}_j)$ is the kernel matrix, $\mathbf{A} \in R^{(n+m)\times(n+m)}$ represents the diagonal matrix used to identify the domain. If $i \in \mathcal{D}_s$, $\mathbf{A}_{ii} = 1$, otherwise $\mathbf{A}_{ii} = 0$. $\mathbf{Y} = [y_1, y_2, \cdots, y_{n+m}]^{\mathrm{T}}$ indicates the label of source and target domains.

In summary, each part of the algorithm is optimized under the framework of structural risk minimization, and combined with the above parts, DASP can be expressed as:

$$\min\{Empirical\ risk\} + \eta\{Modelcomplexity\} + \lambda\{Distributionshift\} + \rho\{Manifoldregularization\} \tag{26}$$

where η, λ and ρ are the regular coefficients of each item.

According to equations (6), (21) and (25), the objective function can be written as:

$$f_o = \arg\min_{f_o \in \mathcal{H}_K} \left\|(\mathbf{Y} - \mathbf{W}^{\mathrm{T}}\mathbf{K})\mathbf{A}\right\|_{\mathrm{F}}^2 + \eta \mathrm{tr}(\mathbf{W}^{\mathrm{T}}\mathbf{K}\mathbf{W}) + \mathrm{tr}(\mathbf{W}^{\mathrm{T}}\mathbf{K}(\lambda\mathbf{M} + \rho\mathbf{L})\mathbf{K}\mathbf{W}) \tag{27}$$

Let $\partial f_o / \partial \mathbf{W} = 0$, it can get:

$$\mathbf{W}^* = ((\mathbf{A} + \lambda\mathbf{M} + \rho\mathbf{L})\mathbf{K} + \eta\mathbf{I})^{-1}\mathbf{A}\mathbf{Y}^{\mathrm{T}} \tag{28}$$

Using similar working condition selecting (SDS) to understand the data distribution of the current working condition, by selecting working conditions with similar data distribution, the data distribution between different working conditions can be processed to a certain extent the problem of poor transfer effect caused by differences enhances the robustness of the model. The similar working conditions are measured by MMD, and the smaller the calculated value was, the more similar the two working conditions were. So we can proceed as follows: first use MMD to measure the data distribution distance between each working condition, select p working conditions ($q > p$) that are similar to the current working conditions among q historical working conditions, then reconstruct the data through dynamic distribution alignment for this p historical working condition, and reconstruct each group establish a regression model $f_i^r(\cdot), i \in [1, p]$ based on the historical working condition data $\mathbf{X}_{S_i}$, and use MMD to measure the similar weight of each historical working condition. The formula is as follows:

$$\alpha_i = \frac{1}{MMD(\mathbf{X}_{S_i}, \mathbf{X}_T)} \tag{29}$$

Algorithm 1 Pseudo-code of DASP algorithm

Input: Data: q historical working condition data $\mathbf{X}_1 \cdots \mathbf{X}_q$ and its label $\mathbf{Y}_1 \cdots \mathbf{Y}_q$; current working condition data $\mathbf{X}_t$. The regular coefficients η , λ , ρ and the number of iterations t of each item.

Output: current working condition label $\mathbf{Y}_t$.

1: Select p working conditions similar to the current working conditions among q historical working conditions.
2: Use multi-view clustering to construct initial pseudo label for historical conditions, establish a clustering model, and use current working conditions to predict its pseudo label $\hat{y}_t$;
3: Construct the kernel matrix $\mathbf{K}$ and the hypergraph Laplacian matrix $\mathbf{L}$;
4: **for** each $i \in [1, t]$ **do**
5: Calculate the balance factor μ, and calculate the marginal distribution matrix $\mathbf{M}_0$ and conditional distribution matrix $\mathbf{M}_c$ by formulas (8) and (9);
6: Calculate the projection matrix $\mathbf{W}^*$ in the objective function by formula (28), and obtain the reconstructed historical working condition and current working condition data.
7: Update the pseudo label $\hat{y}_t$ and the hypergraph Laplacian matrix L of the target domain;
8: **end for**
9: Use the reconstructed historical working condition data to train the regression model, and test the reconstructed current working condition data to obtain the required prediction label;
10: Calculate the final current working condition label from the predicted label obtained from each similar working condition, and obtain the root mean square error with the real label.

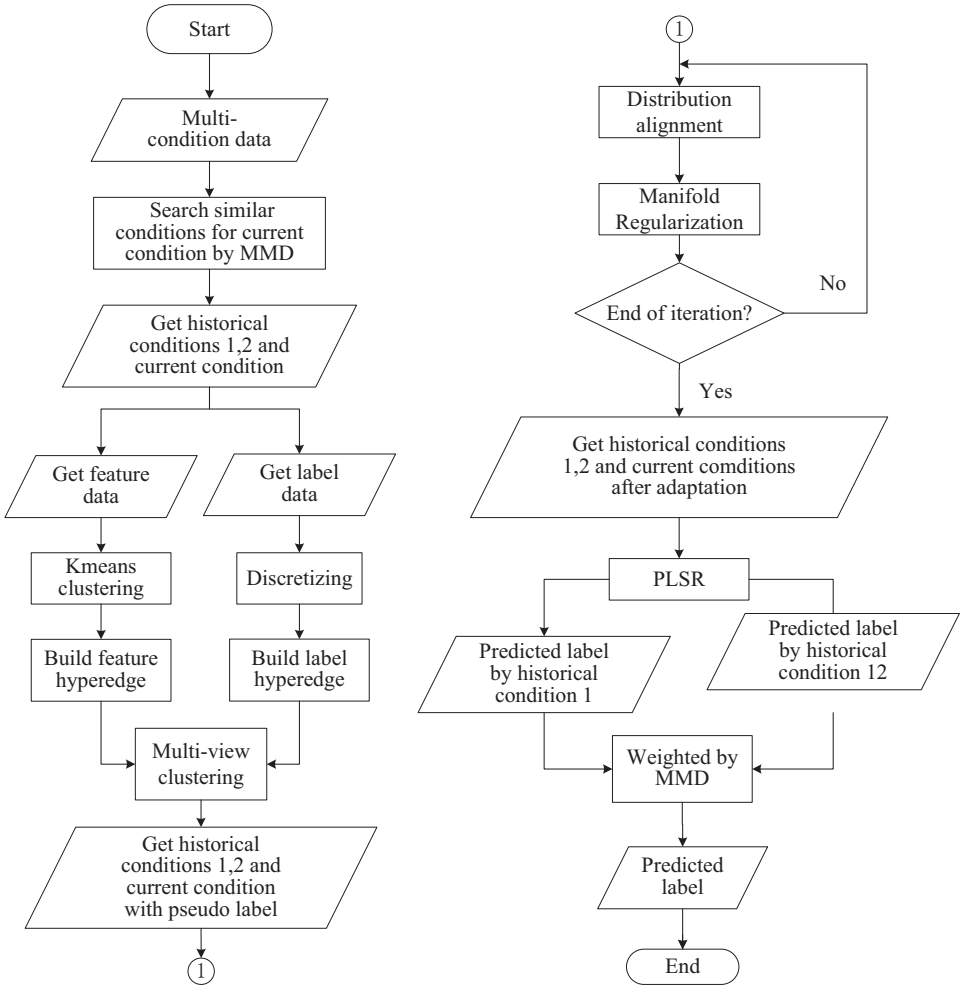


Figure 3: Flow chart of DASP

$$\beta_i = \frac{\alpha_i}{\sum_{i=1}^{p} \alpha_i} \tag{30}$$

where α_i is the reciprocal of the MMD between the i-th reconstructed historical working condition data and the current working condition data. β_i is the weight of the i-th regressor. The integrated regression model $f^r(\cdot)$ can be expressed as:

$$f^r = \beta_1 f_1^r + \beta_2 f_2^r + \cdots + \beta_p f_p^r \tag{31}$$

The regression model is established through the above formula for prediction, and each regression machine is used to predict the label of the current working condition.The pseudo code of DASP algorithm is shown in Algorithm 1. The DASP flow chart is shown in Figure 3.

5. Experiment

In this section, several experiments are conducted to evaluate the performance of the proposed DASP method in multiple data sets.

5.1. Data set

TE dataset: The Tennessee Eastman process[38] was created by Eastman chemical company and can simulate the chemical production process. It is a typical multi-modal process, and its operating point can be adjusted according to production requirements, so that the data can produce multi-modal and multi-condition characteristics. The whole process consists of five main operating units: reactor, stripper, condenser, gas-liquid separator and circulating compressor. There are 8 kinds of material components in the whole process, including the reacting gases A, C, D, E and the inert and insoluble B, the liquid products G and H, and the by-product F. The working process is shown in Figure 4. In addition, the entire TE process involves 41 measured variables and 12 control variables, of which 41 monitored variables are divided into 22 process variables and 19 component variables.

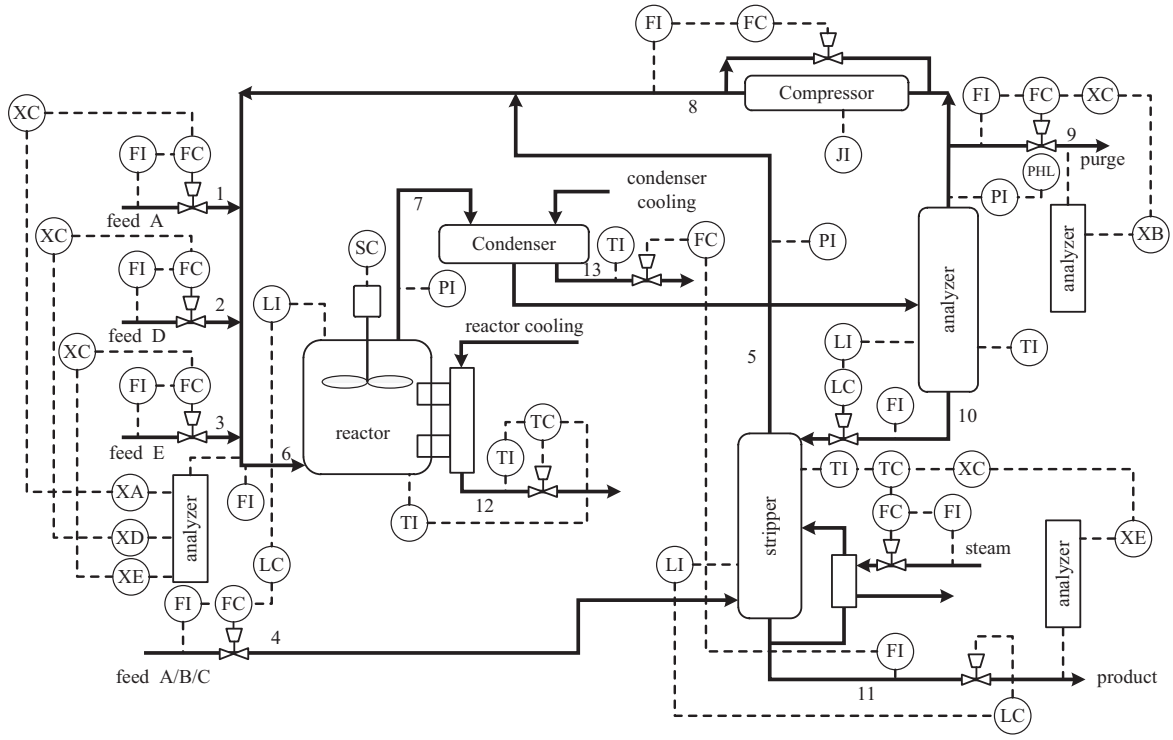


Figure 4: Schematic diagram of TE process

In view of the fact that the reactor pressure and the reactor liquid level have the most important influence on the product, experiments in this paper changes the reactor pressure setting value and the reactor liquid level to make the system produce multi-model characteristics. In order to simulate the

continuous production scenario in the industrial process, the entire TE process is based on the setting value of working condition 1 as the initial state. After the simulation runs for 50 hours, it is switched to working condition 2, and the setting value of working condition is switched according to the same running time, until the operation reaches the end of the set value under working condition 18. Working condition setting of TE process is shown in Table 1. The data sampling interval of all working conditions is 3 minutes, that is, 1000 samples are collected under each working condition. Since the stirring rate among the 12 control variables belongs to the mechanical field and will not have a great impact on the final product, 22 process variables and 11 control variables are selected as input for each sample under all working conditions in this article.

Table 1: Working condition setting of TE process

System Settings	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6	Mode 7	Mode 8	Mode 9
Reactor pressure	2800	2750	2700	2650	2600	2550	2500	2450	2400
Reactor liquid level	65	65	65	65	65	65	65	65	65
System Settings	Mode 10	Mode 11	Mode 12	Mode 13	Mode 14	Mode 15	Mode 16	Mode 17	Mode 18
Reactor pressure	2350	2300	2300	2350	2400	2450	2500	2550	2600
Reactor liquid level	65	65	75	75	75	75	75	75	75

Ball mill dataset: Ball mill is a typical energy consuming equipment, widely used in electric power, chemical industry and other process industries. The accurate detection of the load parameters of the ball mill is of great significance to the optimization control of the grinding process, energy saving and consumption reduction, and safe operation. The comprehensive and complex characteristics of the grinding process and the characteristics of the operation of the ball mill make it difficult to directly detect the key internal parameters. Therefore, the use of an effective soft sensor strategy to predict the load parameters of the ball mill is a problem worthy of study in the multi-modal soft sensor. This experiment uses a small-scale wet ball mill in the laboratory as shown in Figure 5 to perform soft sensor

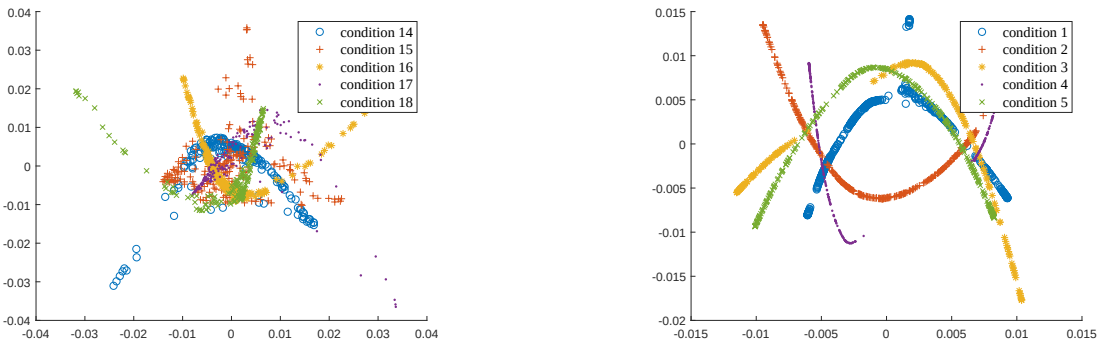
Figure 5: Ball mill equipment used in the experiments

modeling and prediction of load parameters. By changing the ball volume ratio to simulate the sudden change of working conditions, using a multi-channel data acquisition device, five groups of vibration signals were collected on the ball mill. In order to ensure the high resolution of the load parameters, each group has carried out sufficient experiments and synchronously collected vibration signals on site. For each group of experiments, the charge volume ratio (CVR), the pulp density (PD) and the material to ball volume ratio (MBVR) were changed by changing the amount of material. The experimental setup is shown in Table 2. Each group of working condition data and vibration signal is divided into 20 samples on average. The coverage length of each sample is longer than the rotation time of the wet ball mill. Then the fast Fourier transform is used to transform the time-domain signal, which is difficult to model, into the frequency-domain signal.

Table 2: Working condition setting of ball mill

working condition	steel ball/kg	water /kg	starting material/kg	ending material/kg	material change times
1	292	35	25.5	174	139
2	340.69	40	29.7	170.1	103
3	389.36	40	34.2	157.5	88
4	483.02	35	23.4	151.2	95
5	486.7	40	15.3	144.9	102

Figure 6 shows that the two data sets are processed respectively, and the data of five working conditions are randomly selected to be reduced to 2 dimensions for plane visualization. As can be seen from the figure, the data distribution modes under different working conditions have certain similarity, but there are obvious distribution differences. Different working conditions all belong to the same process, so there is a strong similarity between different working conditions. However, when the composition content changes, the mechanism equation of key variable parameters will also change. At the same time, the sensor type and position are not changed when the data under different working conditions



(a) TE (b) Ball mill

Figure 6: Multi-condition feature distribution diagram

are sampled, so the multi-working condition ball mill and TE experiment is a typical multi-working condition data.

5.2. Experimental setup

TE dataset: In the TE process experiments, it is assumed that the historical working condition is the source domain and the current working condition is the target domain. Experiments take the task of predicting component A (label 29), component F (label 34), and component G (label 35) in the component variables. Use working conditions 1 to 11 are historical working conditions, and current working conditions are working conditions 12 to 18;

Ball mill dataset: In the load parameter prediction of the ball mill, due to the limited number of working conditions collected during experiments, when one of the working conditions is the current working condition, the remaining four working conditions are historical working conditions. Experiments predict and compare the three load parameters of MBVR, PD and CVR.

To demonstrate the prediction performance of the proposed method, the soft sensing model composed of the bagging, the JITL-PLS, the JTIL-SVR, the RPLS, the MW-PLS, the MW-SVR and the EL are used to compare the DASP method. After optimization, we set the following parameters for the comparison methods. For the JITL-PLS and JTIL-SVR, we select 30 samples that are closest to the current test sample to train the model. For the MW-PLS and the MW-SVR, we set the moving window size to 100 samples. During the experiment, PLS and SVR model parameters are automatically updated through the toolbox by Matlab2018b. For the RPLS, we set the forgetting factor to 0.98. The basic model we chose is the decision tree for the bagging and the EL. We set the number of learning cycles to 20 for bagging and we set the number of trees to 100 for random forest-based ensemble learning.

5.3. Evaluation index

In order to quantify the prediction performance of various methods, root mean square error (RMSE) is used as the evaluation standard of measurement accuracy, and the calculation formula is as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2} \tag{32}$$

where y_i and $\hat{y}_i$ represent the true value and predicted value of the i-th sample, respectively. N is the number of samples.

5.4. Experimental results

TE dataset: Table 3 shows the experimental results of TE process data using 1-11 working conditions to predict A,F and G of working conditions 12-18.The result record contains the average value of the ten tests, and the symbol " $\rightarrow$ " means to transfer the historical working condition to the current working condition. Figure 7-Figure 9 (a)-(h) show the single results of 10, 11 working conditions

Table 3: Comparison of RMSE of different methods for TE predicting results

Ingredient	Method	Current working condition						
		→12	→13	→14	→15	→16	→17	→18
29	Bagging	1.9258	2.1338	1.8273	1.1630	1.6436	1.4071	2.1024
	JITL-PLS	1.8302	2.7666	2.0433	2.0706	1.7029	2.1134	1.9512
	JITL-SVR	0.8850	2.9821	2.2101	2.4396	2.0349	2.6441	2.6421
	RPLS	1.0923	2.8461	3.7280	3.8763	4.2674	2.8916	2.9901
	MW-PLS	0.8127	1.2759	1.1841	2.0422	2.5417	3.5234	4.5488
	MW-SVR	1.3369	3.1269	2.8717	3.6683	2.5617	3.5756	4.1237
	EL	1.0358	2.5994	1.7896	1.3410	1.8202	1.8477	1.7436
	DASP	0.7455	1.4930	1.1445	0.8500	0.7231	0.8147	1.1094
34	Bagging	0.3589	0.8426	0.6436	0.5318	0.2916	0.4572	0.4702
	JITL-PLS	0.4328	0.5106	1.0972	0.8099	0.2420	0.2960	0.5865
	JITL-SVR	0.1656	0.5089	0.4895	0.5269	0.2212	0.2843	0.3316
	RPLS	0.1357	0.2296	0.2247	0.4020	0.2536	0.4422	0.4086
	MW-PLS	0.1540	0.4021	0.6298	0.9046	1.8994	2.0049	2.3122
	MW-SVR	0.3029	0.7067	0.2814	0.3270	0.9869	0.8670	1.1034
	EL	0.3914	0.6764	0.5028	0.5533	0.2665	0.3194	0.4410
	DASP	0.1198	0.2041	0.1668	0.1739	0.2061	0.1932	0.2812
35	Bagging	0.2677	0.5965	0.3965	0.4870	0.3656	0.6177	0.5231
	JITL-PLS	0.3356	0.5755	0.4300	0.5512	0.3629	0.7479	0.5233
	JITL-SVR	0.1324	0.2050	0.1994	0.1740	0.1908	0.1979	0.3373
	RPLS	0.1042	0.1339	0.2071	0.1540	0.1222	0.2457	0.2195
	MW-PLS	0.0870	0.1050	0.1101	0.1771	0.2790	0.3513	0.4383
	MW-SVR	0.1121	0.2557	0.4493	0.6085	0.7645	0.8437	1.0187
	EL	0.2540	0.3934	0.4768	0.4996	0.4034	0.4436	0.4773
	DASP	0.0876	0.1070	0.0967	0.0967	0.0926	0.0970	0.1237

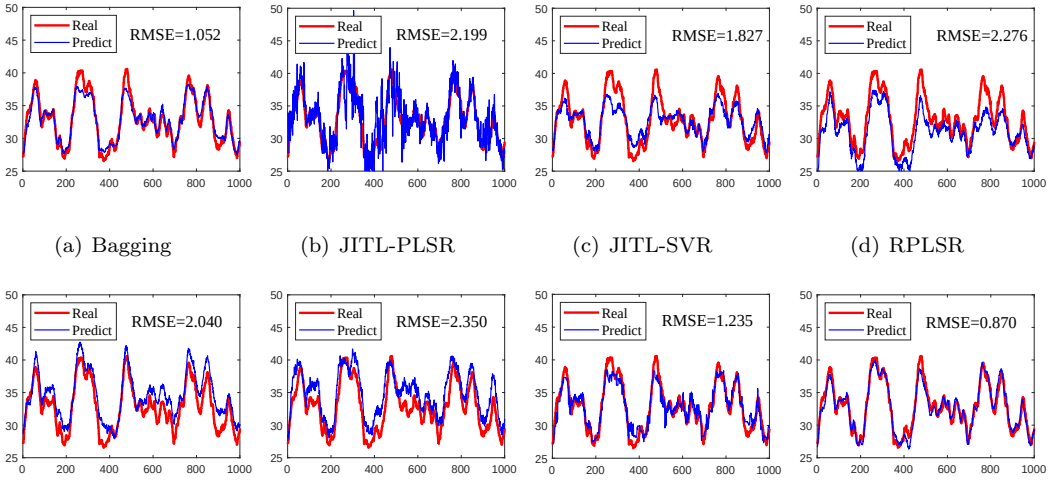
predicting 15 working condition. From these experiments, it can be seen that the fitting degree of regression prediction based on DASP method is higher, whose RMSE between the real value and the predicted value is smaller.

Ball mill dataset: In order to verify the effectiveness in the actual work environment, DASP is selected to transfer three component variables MBVR, PD and CVR in ball mill, which predict result is 10 times average. The experimental results are shown in the Table 4. Figure 10-Figure 12 (a)-(h) are the single results of 3, 5 working conditions predicting 4 working condition.

The method proposed in this paper uses pseudo-labels for joint distribution alignment, in order to design a more fair comparison experiment, for RPLS, MW-SVR, MW-PLS, we use the predicted value of the first local model as a pseudo-label to update the model in real time. For EL, bagging, JITL-PLS, JITL-SVR, we use all historical working condition samples as the training set, and the current working condition as the test set for experiment. From these results, it can be seen that the prediction effect of JITL is not ideal. Compared with JITL, EL and bagging has improved some prediction effects, but it does not substantially reduce the data difference between different working conditions, so the model

Table 4: Comparison of RMSE of different methods for predicting the load parameters of the ball mill under various working conditions

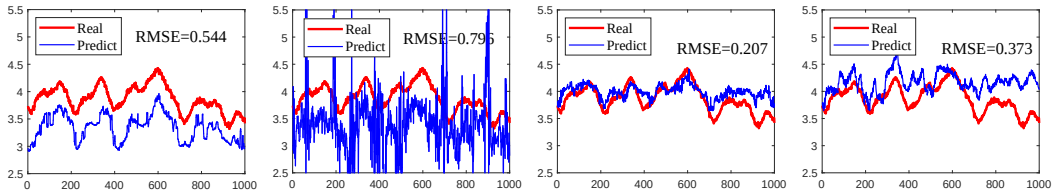
Parameter	Method	Current working condition			
		→ 2	→ 3	→ 4	→ 5
MBVR	Bagging	0.2817	0.4225	0.3838	0.5756
	JITL-PLS	0.3688	0.4490	0.7043	0.9147
	JITL-SVR	0.1510	0.2588	0.5026	0.3571
	RPLS	0.6552	0.6519	1.2892	3.3013
	MW-PLS	0.2527	0.6149	1.5063	4.0229
	MW-SVR	0.7784	0.3603	1.2474	1.4655
	EL	0.2851	0.2970	0.4119	0.4468
	DASP	0.1228	0.1107	0.0679	0.3749
PD	Bagging	0.0720	0.1950	0.1135	0.4210
	JITL-PLS	0.0695	0.1149	0.3406	0.3467
	JITL-SVR	0.0342	0.0681	0.1280	0.1534
	RPLS	0.1215	0.1530	0.2225	0.6087
	MW-PLS	0.0451	0.0456	0.1803	0.4052
	MW-SVR	0.0738	0.0935	0.1298	0.1839
	EL	0.0444	0.0733	0.0833	0.1506
	DASP	0.0319	0.0223	0.0387	0.0543
CVR	Bagging	0.0927	0.2368	0.1557	0.3845
	JITL-PLS	0.0908	0.1428	0.1777	0.2471
	JITL-SVR	0.0819	0.1355	0.1413	0.1947
	RPLS	0.1588	0.2059	0.2640	0.5257
	MW-PLS	0.1055	0.1850	0.2718	0.6061
	MW-SVR	0.0800	0.0944	0.0922	0.1042
	EL	0.0963	0.1329	0.1468	0.1805
	DASP	0.0350	0.026	0.0163	0.0181



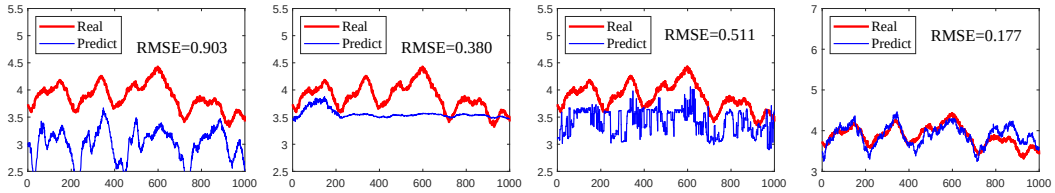
(a) Bagging (b) JITL-PLSR (c) JITL-SVR (d) RPLSR

(e) MW-PLSR (f) MW-SVR (g) EL (h) DASP

Figure 7: TE component A prediction results

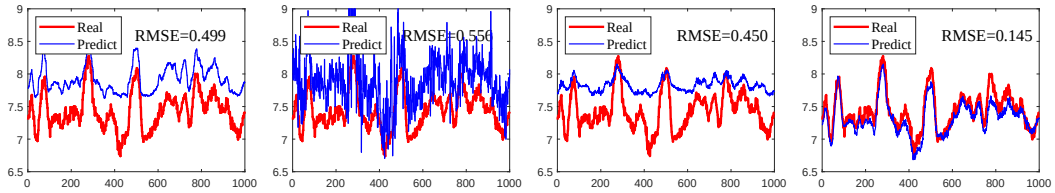


(a) Bagging (b) JITL-PLSR (c) JITL-SVR (d) RPLSR

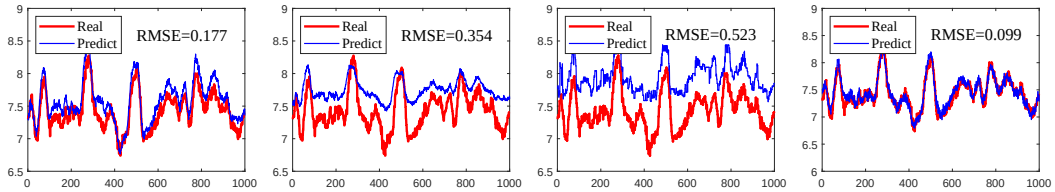


(e) MW-PLSR (f) MW-SVR (g) EL (h) DASP

Figure 8: TE component F prediction results

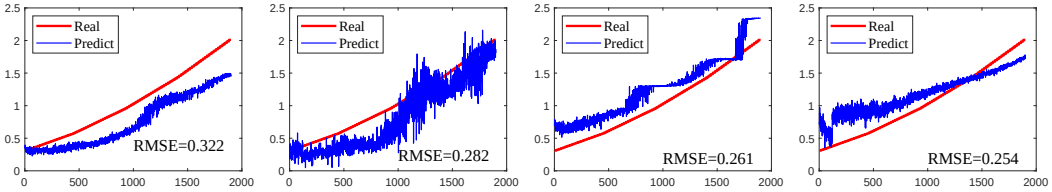


(a) Bagging (b) JITL-PLSR (c) JITL-SVR (d) RPLSR

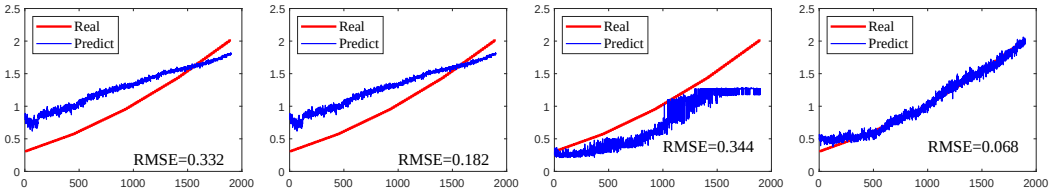


(e) MW-PLSR (f) MW-SVR (g) EL (h) DASP

Figure 9: TE component G prediction results

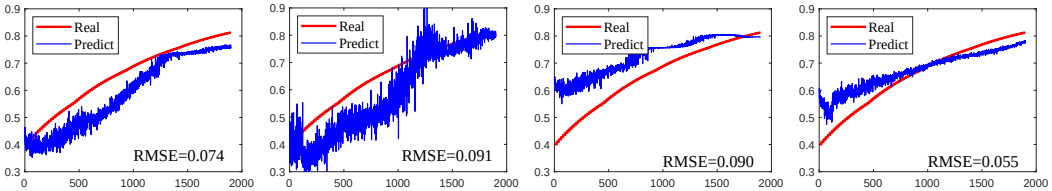


(a) Bagging (b) JITL-PLSR (c) JITL-SVR (d) RPLSR

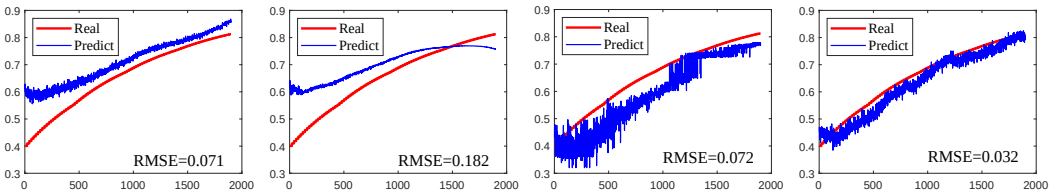


(e) MW-PLSR (f) MW-SVR (g) EL (h) DASP

Figure 10: MBVR prediction results of ball mill

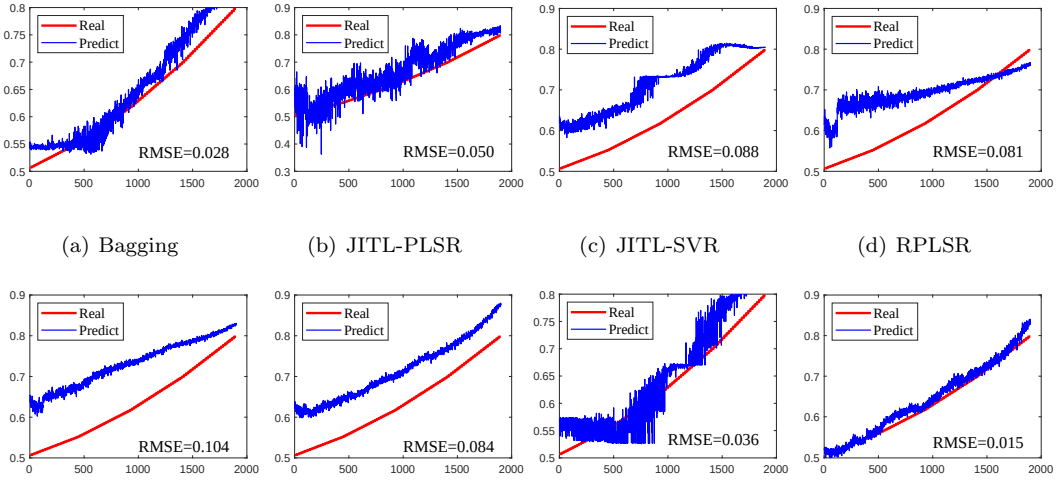


(a) Bagging (b) JITL-PLSR (c) JITL-SVR (d) RPLSR



(e) MW-PLSR (f) MW-SVR (g) EL (h) DASP

Figure 11: PD prediction results of ball mill



(a) Bagging (b) JITL-PLSR (c) JITL-SVR (d) RPLSR

(e) MW-PLSR (f) MW-SVR (g) EL (h) DASP

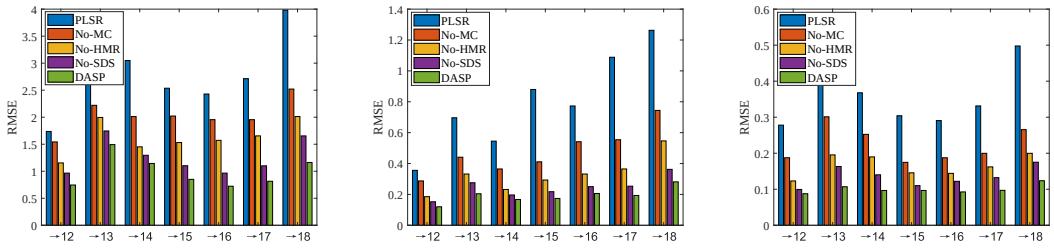
Figure 12: CVR prediction results of ball mill

performance has not improved much. For RPLS, MW-SVR and MW-PLS, When using pseudo-labels to replace real labels to update the model, these methods gradually increase the prediction error as samples are added. The comparison methods have very unsatisfactory prediction effects for each component. As the working conditions change, when there is a big difference between the historical working conditions and the current working conditions, problems such as under-fitting will occur. Therefore, it can only roughly keep up with the true value in the trend, but there are large fluctuations and large errors.

Compared with other forecasting models, the DASP method proposed in this paper shows outstanding advantages in regression problems. It has good forecasting effects in different data sets or in forecasting components, and its forecasting values are well realized. The tracking of the true value highlights the good label prediction effect under unsupervised multi-working conditions, and further proves the effectiveness and robustness of the algorithm.

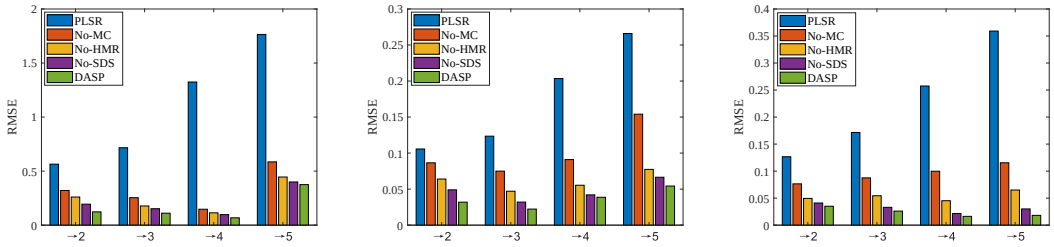
5.5. Impact of each part

In order to verify the influence of each part of the method on the performance of the model, the PLSR model was selected, which did not go through the multi-view clustering (No-MC) model, did not have the hypergraph manifold regularization (No-HMR) model, and did not use similar working condition selecting (No-SDS) model and DASP direct modeling to compare the prediction results of all components under different conditions. As shown in Figure 13 and Figure 14 are the experimental results for two data sets. It can be seen that the prediction error of DASP is the smallest, and the introduction of each item can further improve prediction accuracy of the model. The reason is that due to the large differences between the data working conditions, direct modeling without distribution alignment will lead to unsatisfactory prediction results, and the effect of joint distribution adaptation



(a) component A (b) component F (c) component G

Figure 13: Comparison of TE component soft sensor RMSE results of different prediction methods



(a) MBVR (b) PD (c) CVR

Figure 14: Different methods of ball mill parameter soft sensor RMSE results

is significantly improved, but in the process of feature transformation, if the model does not establish constraints on feature and label, which will destroy its data structure and have a great impact on the effect of domain adaptation. And by selecting similar working conditions to improve the generalization of the model can make it have the same good effect under different forecasting conditions.

5.6. Parameter Sensitivity

In essence, label discretization is equal clustering of continuous data. The selection of category number will directly affect the range of each segment in the discrete process. In this experiment, the

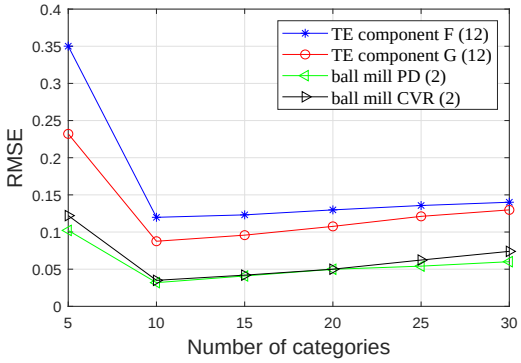
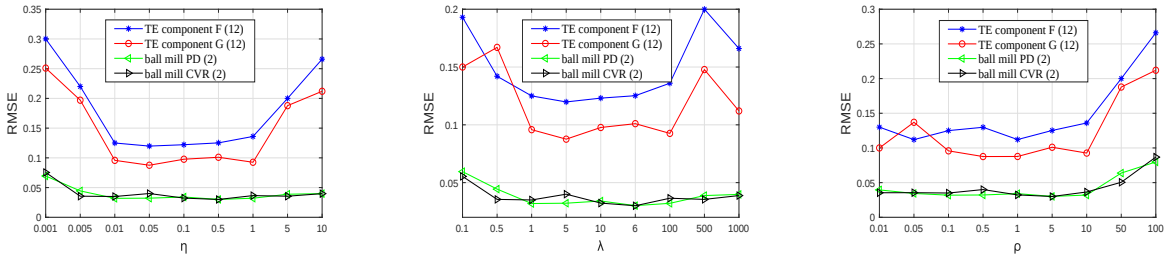


Figure 15: Number of categories analysis

optimal category number is determined by discretization of labels into different categories and running DASP. As shown in the figure 15. By experimenting with random tasks in two data sets, it was observed. Within a reasonable range, the prediction ability gradually decreases with the number of optimal categories. If the number of categories is too small or too many, the prediction accuracy will be reduced. The experiment runs DASP with a wide range of values for parameters η, λ and ρ on several random tasks to compare its performance in Figure 16 (a), (b)and(c). DASP can achieve a robust performance with regard to a wide range of parameter values. Specifically, the best choices of these parameters are: $\lambda \in [1, 100]$, $\eta \in [0.01, 1]$, $\rho \in [0.01, 1]$. To sum up, the performance of DASP stays robust with a wide range of regularization parameter choices.



(a) η (b) λ (c) ρ

Figure 16: Parameter sensitivity analysis

6. Conclusion

In this paper, a multi-source unsupervised soft sensor method based on joint distribution alignment and mapping structure preservation is adopted. This method preserves the mapping relationship between feature and label, and uses joint distribution adaptation to reduce known modal data and unknown modalities. The difference of distance between the state data improves the performance of the unsupervised soft sensor model. In order to verify the effectiveness of the method, it was applied to the soft sensor of TE and the load parameters of wet ball mill with multiple working conditions, and the soft sensor modeling of multiple working conditions was completed. The experimental results show that the method proposed in this paper can effectively improve the prediction accuracy of the model.

Acknowledgement

This work was supported in part by National Natural Science Foundation of China (61973226, 62073232), Key Research and Development Projects of Shanxi, China (201903D121143), the Major Science and Technology Projects of Shanxi, China (20181102017), Natural Science Foundation of Shanxi, China (201801D221181).

References

[1] B. Alakent, Soft-sensor design via task transferred just-in-time-learning coupled transductive moving window learner, Journal of Process Control 101 (2021) 52–67. doi:10.1016/j.jprocont.2021.03.006.

[2] Y. Lü, H. YANG, A multi-model approach for soft sensor development based on feature extraction using weighted kernel fisher criterion, Chinese Journal of Chemical Engineering 22 (2) (2014) 146–152. doi:https://doi.org/10.1016/S1004-9541(14)60007-0.

[3] C. Z. Yan Qin, B. Huang, A new soft-sensor algorithm with concurrent consideration of slowness and quality interpretation for dynamic process, Chemical Engineering Science 199 (18) (2019) 28–39. doi:doi.org/10.1016/j.ces.2019.01.011.

[4] Q. Liu, S. J. Qin, Perspectives on big data modeling of process industries, Acta Automatica Sinica 42 (02) (2016) 161–171. doi:10.16383/j.aas.2016.c150510.

[5] J. Liu, On-line soft sensor for polyethylene process with multiple production grades, Control Engineering Practice 15 (7) (2007) 769–778, special Issue on Award Winning Applications. doi:10.1016/j.conengprac.2005.12.005.

[6] A. Gretton, K. Borgwardt, M. J. Rasch, B. Schoelkopf, A. Smola, A kernel two-sample test, Journal of Machine Learning Research 13 (2012) 723–773.

[7] M. Long, J. Wang, G. Ding, J. Sun, P. Yu, Transfer feature learning with joint distribution adaptation, in: Proceedings of the IEEE international conference on computer vision, 2013, pp. 2200–2207. doi:10.1109/ICCV.2013.274.

[8] J. Wang, W. Feng, Y. Chen, H. Yu, P. S. Yu, Visual domain adaptation with manifold embedded distribution alignment, in: Proceedings of the 26th ACM international conference on Multimedia, 2018, pp. 402–410. doi:10.1145/3240508.3240512.

[9] Z. Ge, F. Gao, Z. Song, Mixture probabilistic pcr model for soft sensing of multimode processes, Chemometrics and Intelligent Laboratory Systems 105 (1) (2011) 91–105. doi:10.1016/j.chemolab.2010.11.004.

[10] C. Mei, Y. Su, G. Liu, Y. Ding, Z. Liao, Dynamic soft sensor development based on gaussian mixture regression for fermentation processes, Chinese Journal of Chemical Engineering 25 (2017) 116–122. doi:10.1016/j.cjche.2016.07.005.

[11] S. Tan, F. Wang, J. Peng, Y. Chang, S. Wang, Multimode process monitoring based on mode identification, Industrial and Engineering Chemistry Research 51 (2012) 374–388. doi:10.1021/ie102048f.

[12] L. I. Yuan, W. U. Haoyu, C. Zhang, L. Feng, Multi-modal process fault detection method based on improved partial least squares, Journal of Computer Applications 38 (12) (2018) 3601–3606.

[13] P. Kadlec, R. Grbić, B. Gabrys, Review of adaptation mechanisms for data-driven soft sensors, Computers and Chemical Engineering 35 (1) (2011) 1–24. doi:10.1016/j.compchemeng.2010.07.034.

[14] W. Shao, X. Tian, Semi-supervised selective ensemble learning based on distance to model for nonlinear soft sensor development, Neurocomputing 222 (2017) 91–104. doi:10.1016/j.neucom.2016.10.005.

[15] Z. Chai, C. Zhao, B. Huang, H. Chen, A deep probabilistic transfer learning framework for soft sensor modeling with missing data, IEEE Transactions on Neural Networks and Learning Systems (2021) 1–12.doi:10.1109/TNNLS.2021.3085869.

[16] Z. Chai, C. Zhao, B. Huang, Multisource-refined transfer network for industrial fault diagnosis under domain and category inconsistencies, IEEE Transactions on Cybernetics (2021) 1–13.doi:10.1109/TCYB.2021.3067786.

[17] S. Chen, L. Han, X. Liu, Z. He, X. Yang, Subspace distribution adaptation frameworks for domain adaptation, IEEE Transactions on Neural Networks and Learning Systems 31 (12) (2020) 5204–5218. doi:10.1109/TNNLS.2020.2964790.

[18] A. Kumagai, T. Iwata, Unsupervised domain adaptation by matching distributions based on the maximum mean discrepancy via unilateral transformations, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2019, pp. 4106–4113. doi:10.1609/aaai.v33i01.33014106.

[19] S. J. Pan, I. W. Tsang, J. T. Kwok, Q. Yang, Domain adaptation via transfer component analysis, IEEE Transactions on Neural Networks 22 (2) (2011) 199–210. doi:10.1109/TNN.2010.2091281.

[20] M. Kan, J. Wu, S. Shan, X. Chen, Domain adaptation for face recognition: Targetize source domain bridged by common subspace, International Journal of Computer Vision 109 (2014) 94–109. doi:10.1007/s11263-013-0693-1.

[21] B. Quanz, J. Huan, M. Mishra, Knowledge transfer with low-quality data: A feature extraction issue, IEEE Transactions on Knowledge and Data Engineering 24 (10) (2012) 1789–1802. doi:10.1109/TKDE.2012.75.

[22] J. Liang, R. He, Z. Sun, T. Tan, Aggregating randomized clustering-promoting invariant projections for domain adaptation, IEEE Transactions on Pattern Analysis and Machine Intelligence 41 (5) (2019) 1027–1042. doi:10.1109/TPAMI.2018.2832198.

[23] Y. Liu, W. Tu, B. Du, L. Zhang, D. Tao, Homologous component analysis for domain adaptation, IEEE Transactions on Image Processing 29 (2020) 1074–1089. doi:10.1109/TIP.2019.2929421.

[24] D. Yonggui, L. Sisi, Y. Gaowei, C. Lan, Soft sensor of wet ball mill load parameter based on domain adaptation with manifold regularization, CIESC Journal 69 (03) (2018) 1244–1251. doi:10.11949/j.issn.0438-1157.20170918.

[25] Z. Fang, Z. Zhang, Simultaneously combining multi-view multi-label learning with maximum margin classification, in: 2012 IEEE 12th International Conference on Data Mining, 2012, pp. 864–869. doi:10.1109/ICDM.2012.88.

[26] Y. Guo, Convex subspace representation learning from multi-view data, in: Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence, AAAI'13, AAAI Press, 2013, pp. 387–393.

[27] M. White, Y. Yu, X. Zhang, D. Schuurmans, Convex multi-view subspace learning, in: Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1, Vol. 3, 2012, pp. 1673–1681.

[28] C. Lu, S. Yan, Z. Lin, Convex sparse spectral clustering: Single-view to multi-view, IEEE Transactions on Image Processing 25 (6) (2016) 2833–2843. doi:10.1109/TIP.2016.2553459.

[29] M. Brbić, I. Kopriva, Multi-view low-rank sparse subspace clustering, Pattern Recognition 73 (2018) 247–258. doi:https://doi.org/10.1016/j.patcog.2017.08.024.

[30] D. Zhou, J. Huang, B. Schlkopf, Learning with hypergraphs: Clustering, classification, and embedding, in: Advances in Neural Information Processing Systems 19: Proceedings of the 2006 Conference, 2007, pp. 1601–1608. doi:10.7551/mitpress/7503.003.0205.

[31] S. Agarwal, K. Branson, S. Belongie, Higher order learning with graphs, in: Proceedings of the 23rd international conference on Machine learning, Vol. 148, 2006, pp. 17–24. doi:10.1145/1143844.1143847.

[32] Y. Wang, P. Li, C. Yao, Hypergraph canonical correlation analysis for multi-label classification, Signal Processing 105 (2014) 258–267. doi:10.1016/j.sigpro.2014.05.032.

[33] Y. Liu, C. Yang, M. Zhang, Y. Dai, Y. Yao, Development of adversarial transfer learning soft sensor for multigrade processes, Industrial and Engineering Chemistry Research 59 (2020) 16330–16345. doi:10.1021/acs.iecr.0c02398.

[34] S. Ben-David, J. Blitzer, K. Crammer, F. Pereira, Analysis of representations for domain adaptation, in: Advances in Neural Information Processing Systems 19: Proceedings of the 2006 Conference, 2007, pp. 137–144. doi:10.7551/mitpress/7503.003.0022.

[35] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, Foundations and Trends in Machine Learning 3. doi:10.1561/2200000016.

[36] A. Y. Ng, M. I. Jordan, Y. Weiss, On spectral clustering: Analysis and an algorithm, in: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS, MIT Press, 2001, pp. 849–856.

[37] M. Belkin, P. Niyogi, V. Sindhwani, Manifold regularization: A geometric framework for learning from labeled and unlabeled examples, Journal of Machine Learning Research 7 (1) (2006) 2399–2434.

[38] N. L. Ricker, Decentralized control of the tennessee eastman challenge process, Journal of Process Control 6 (1996) 205–221. doi:10.1016/0959-1524(96)00031-5.