A ROBUST AND ADAPTIVE HYBRID FUZZY-BAYESIAN WEIGHTED FRAMEWORK FOR ONLINE QUALITY PREDICTION IN BATCH PROCESSES

JIN-LONG ZHAO[1,2], MING-YI YANG[1,2], ZHI-GANG XU[1,2], CHENG XIE[1,2], PENG-FEI YIN[3]

[1] Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China
[2] University of Chinese Academy of Sciences, Beijing 100049, China
[3] Xi'an North Huian Chemical Industries Co., Ltd, Xi'an, 710302, Shanxi Province, China
E-MAIL: myyang@sia.cn

Abstract:

Industrial batch process quality prediction faces challenges of time-varying dynamics and data noise. Global modeling cannot adapt to process changes, while traditional just-in-time learning (JITL) lacks sample reliability evaluation, restricting prediction performance. This study proposes a novel online learning method that integrates fuzzy Bayesian network (FBN) into JITL for robust local modeling. FBN is built from historical data to quantify variable dependencies and assign sample reliability weights; a composite similarity index combining spatial proximity and reliability is used for local sample selection. A weighted regression model is trained for prediction, with an online update mechanism for historical data. Industrial experiments show the method outperforms conventional approaches in accuracy and generalization, verifying the effectiveness of combining dynamic local learning with probabilistic uncertainty modeling for real-time quality prediction.

Keywords:

Batch Processes; Quality Prediction; Fuzzy Bayesian Network; Online Learning

1. Introduction

Industrial batch processes, such as those in pharmaceuticals and chemicals, play a crucial role in modern manufacturing[1], and quality prediction therein is essential for optimizing operations, ensuring product consistency and cutting production costs. Yet traditional quality prediction methods often struggle with the time-varying characteristics and complex production environment of such processes[2, 3], and many key quality variables cannot be directly sensor-measured but need estimation or inference, thus requiring soft sensor technology for real-time monitoring. Traditional industrial quality prediction methods are mostly model-driven, relying on physical and chemical models that demand profound domain expertise[4], while data-driven soft sensors leverage machine learning and statistical techniques to efficiently mine historical data patterns without domain-specific knowledge, providing flexible, low-cost solutions for handling multidimensional and nonlinear relationships[5]. Nevertheless, batch process data is plagued by inter-batch differences and drift, posing great challenges to traditional soft sensor methods[6, 7] , and online learning methods, which can adapt to data changes and update models in real time, exhibit significant advantages in addressing these problems[8].

Online learning methods have emerged as a key research direction in soft sensing for their real-time model update capability to adapt to dynamically changing industrial process data, with various strategies proposed for diverse application scenarios in recent years[9, 10]. T Just-in-Time Learning (JITL) calculates the similarity between current and historical samples to construct local models only during prediction, which cuts computational burden and boosts real-time performance[11]. For example, the JITL online learning framework proposed by Chen et al. (2015) dynamically adjusts parameters with real-time data, showing good adaptability in time-varying industrial processes[12]. However, JITL performance relies heavily on accurate similarity measurement and is sensitive to noise and outliers, leading to unstable predictions with poor-quality data[13, 14]. The sliding window-based online learning approach is another common strategy, which retains the latest data subset via a fixed-size window to focus on recent process changes—Albert et al. (2007) applied this to stream data analysis, effectively tackling large-scale data real-time processing[15], but window size selection is a core bottleneck: an overly small window may lose long-term trends, while an excessively large one introduces redundant information and impairs computational efficiency[16].

Despite advances in online learning, industrial batch data are often corrupted by heavy noise, outliers and drift-induced uncertainty. Traditional JITL methods only adopt Euclidean distance and other deterministic metrics for similarity calculation, which fails to reflect the true sample similarity under noisy or anomalous data and easily selects "false similar" samples that deteriorate model performance. To address this issue, this paper proposes an Adaptive Fuzzy-Bayesian Weighted Online Learning Method (AFBW-OLM). Its core innovation is extending similarity from pure spatial distance to a composite index combining proximity and sample reliability. Specifically, it uses Euclidean distance to measure spatial proximity and introduces a fuzzy Bayesian network to compute sample confidence for reliability evaluation in uncertain scenarios. The organic combination of the two dimensions enables the model to prioritize spatially close and high-confidence samples for local modeling, effectively suppressing the interference of noise and outliers on similarity measurement. In addition, the fuzzy Bayesian network quantifies data uncertainty via membership and non-membership functions, and the Bayesian confidence it outputs serves as the basis for sample weighting—assigning higher contribution weights to high-confidence samples in local modeling, thus ultimately improving the model's prediction accuracy and adaptability in complex industrial quality prediction scenarios.

The remainder of this paper is organized as follows. “Related Work” discusses the work related to the fuzzy Bayesian networks, and online learning techniques in industrial quality prediction. “Proposed Method” provides a comprehensive description of the AFBW-OLM method proposed. “Experiment” introduces the experiment setup and analysis. Finally, the “Conclusions” section presents the conclusions drawn from the study.

2. Related work

2.1. Fuzzy Bayesian Networks

Bayesian Networks (BNs) are foundational probabilistic graphical models that represent variable dependencies via Directed Acyclic Graphs (DAGs) and quantify uncertainties using Conditional Probability Tables (CPTs)[17]. However, traditional BNs require discrete states, leading to arbitrary discretization of continuous data that risks information loss and bias[18, 19]. Fuzzy Bayesian Networks (FBNs) address this limitation by integrating fuzzy set theory with BNs, enabling robust sample weighting for continuous data while preserving probabilistic interpretability[20, 21].

FBN-based sample weighting is implemented through a four-stage framework. First, continuous variables are mapped to fuzzy linguistic states (e.g., "low", "medium", "high") via membership functions that measure the degree of a numerical value’s affiliation with each state. Gaussian membership functions (GaussianMF) are adopted in this study for their smooth, differentiable properties, defined for a variable x as:

$$\mu(x) = \exp\left(-\frac{(x-\mu_c)^2}{2\sigma^2}\right) \tag{1}$$

where μ_c denotes the central value (mean) of the fuzzy state, and σ is the width (standard deviation) controlling the degree of fuzziness.

Second, the FBN is structured as a DAG where nodes represent variables and directed edges encode causal parent-child dependencies. A topological sort (e.g., Kahn’s algorithm) is applied to enforce a non-cyclic node evaluation order, ensuring valid probabilistic inference.

Third, CPTs are learned from training data to quantify $P\left(X_i \mid Pa\left(X_i\right)\right)$. To avoid combinatorial explosion in high-dimensional data, a hard state simplification is adopted for efficient conditional probability table learning. This operation does not compromise the uncertainty preservation of continuous data, since the fuzzification stage with Gaussian membership functions has already encoded the underlying data uncertainty into the model. Formally, for a variable X with fuzzy states $\left\{S_1, S_2, \ldots, S_m\right\}$, the hard state s^* of an observation x is determined by:

$$s^* = \arg\max_i \mu_{S_i(x)} \tag{2}$$

where $\mu_{S_i(x)}$ is the membership degree of x to state S_i, as defined by the Gaussian membership function in the first stage. Laplacian smoothing is applied to handle zero-frequency state combinations, with the smoothed conditional probability computed as:

$$P\left(X_i = x_i \mid Pa\left(X_i = \pi\right)\right) = \frac{N\left(x_i, \pi\right) + \alpha}{N\left(\pi\right) + \alpha \cdot \left|Val\left(X_i\right)\right|} \tag{3}$$

where $N\left(x_i, \pi\right)$ is the count of samples with node X_i taking value x_i and its parents in combination π; $N\left(\pi\right)$ is the total count of samples with parent combination π; $Val\left(X_i\right)$ is the number of fuzzy states for node X_i; and α is the smoothing parameter (typically $\alpha = 1$). After smoothing, counts are normalized to form probabilistic CPTs[22].

Fourth, a sample’s weight (confidence) is defined as the joint probability of its fuzzy state combination, computed by multiplying the conditional probabilities of each node (given parent states) in topological order:

$$P(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P(X_i \mid \mathrm{Pa}(X_i)) \tag{4}$$

where $\mathrm{Pa}(X_i)$ denotes the parent nodes of X_i. For each sample, continuous values are first mapped to maximum-membership fuzzy states; the joint probability of this state combination is then calculated as the sample weight, reflecting the sample's consistency with the FBN's learned probabilistic patterns.

FBNs outperform alternative weighting methods by avoiding discretization-induced information loss (a limitation of discrete BNs) while retaining probabilistic interpretability (a limitation of pure fuzzy logic)[23, 24]. This balance makes FBN-based weighting broadly applicable to domains with continuous variables and uncertain causal relationships, such as industrial monitoring and predictive modeling.

3. Adaptive Fuzzy-Bayesian Weighted Online Learning Method (AFBW-OLM)

This chapter presents the Adaptive Fuzzy-Bayesian Weighted Online Learning Method (AFBW-OLM) for quality prediction in industrial batch processes. The proposed method's architecture can be described by the following Figure 1.

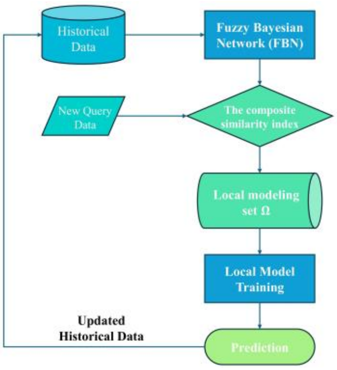

FIGURE 1. The structure of AFBW-OLM

Given a historical batch process dataset $\mathrm{D}=\{(x_i,y_i)\}_{i=1}^{N}$, where $x_i \in \mathbb{R}^d$ is a d-dimensional process feature vector and y_i is the corresponding quality label, and a query sample x_q for which the quality value needs to be predicted, the AFBW-OLM proceeds as follows.

First, a fuzzy Bayesian network is constructed from the historical dataset to model the probabilistic dependencies among process variables, as detailed in Section 2.1. Each continuous variable in the dataset is mapped to a set of fuzzy linguistic states using Gaussian membership functions, as defined in formula (1). The "hard state" of each variable is then determined by selecting the state with the maximum membership degree according to formula (2). The conditional probability tables (CPTs) of the FBN are learned from the hard-state representations of the historical data, with Laplacian smoothing applied as in formula (3) to handle zero-frequency state combinations. For each historical sample x_i, its reliability weight m_i is computed as the joint probability of its hard-state combination, obtained by multiplying the conditional probabilities of all nodes in topological order, as expressed in formula (4):

$$m_i = P(s(x_i)) = \prod_{t=1}^{d} P\left(X_{\pi(t)} = s_{\pi(t)}(x_i) \mid \mathrm{Pa}(X_{\pi(t)}) = s_{\mathrm{Pa}(X_{\pi(t)})}(x_i)\right), \tag{5}$$

where $\pi(t)$ denotes the topological order of nodes, $Pa\left(X_{\pi(t)}\right)$ is the set of parent nodes of $X_{\pi(t)}$, and $s_{Pa\left(X_{\pi(t)}\right)}$ is the tuple of hard states of the parent nodes. The weight m_i reflects the sample's consistency with the learned probabilistic structure; higher values indicate greater reliability and lower susceptibility to noise or outliers.

Next, a weighted similarity metric is used to quantify the relevance between the query sample x_q and each historical sample x_i. Unlike traditional JITL methods that rely solely on Euclidean distance, the proposed comprehensive similarity metric integrates both the normalized Euclidean distance and the FBN-derived sample weight m_i. First, the Euclidean distance $d_i = \|x_q - x_i\|_2$ is normalized to $[0,1]$ across all historical samples:

$$d_i^{\mathrm{norm}} = \frac{d_i - d_{\min}}{d_{\max} - d_{\min}}, \tag{6}$$

where $d_{\min} = \min_i d_i$ and $d_{\max} = \max_i d_i$. The sample weight is normalized analogously:

$$m_i^{\mathrm{norm}} = \frac{m_i - m_{\min}}{m_{\max} - m_{\min}}, \tag{7}$$

with $m_{\min} = \min_i m_i$ and $m_{\max} = \max_i m_i$. The composite similarity index is then defined as:

$$\mathrm{sim}(\mathbf{x}_q, \mathbf{x}_i) = -\frac{m_i^{\mathrm{norm}}}{d_i^{\mathrm{norm}} + \epsilon}, \tag{8}$$

where ϵ is a small positive constant (e.g., 10^{-12}) introduced to avoid division by zero. The negative sign ensures that a smaller normalized distance and a larger normalized confidence jointly yield a larger similarity value; hence the most similar samples—those with the highest $\mathrm{sim}(\mathbf{x}_q, \mathbf{x}_i)$ are both physically close to the query and highly reliable according to the FBN.

Based on the comprehensive similarity values, the top-$K_{neighbor}$ historical samples with the largest $\mathrm{sim}(\mathbf{x}_q, \mathbf{x}_i)$

are selected to form the local modeling set Ω. Then train a local regression model on Ω, taking Gaussian Process Regression (GPR) model as an example, where the FBN weights m_j are incorporated into the kernel matrix to prioritize the contribution of reliable samples. Specifically, the weighted kernel matrix $K_w \in \mathbb{R}^{K\times K}$ is defined as:

$$K_w = M^{1/2} K M^{1/2}, \tag{9}$$

where $M = diag(m_1, m_2, \ldots, m_k)$ is a diagonal matrix of the original sample weights (or their normalized versions), and K is the standard squared exponential kernel matrix:

$$K_{ij} = \exp\left(-\frac{\| x_i - x_j \|^2}{2l^2}\right) + \sigma_n^2 \delta_{ij}, \tag{10}$$

with length-scale parameter l, noise variance σ_n^2, and Kronecker delta δ_{ij}. The predictive mean for the query sample x_q is then computed as:

$$\hat{y}_q = k_q^{\top} (K_w + \sigma_n^2 I)^{-1} y_\Omega, \tag{11}$$

where k_q is the vector of kernel values between x_q and all samples in Ω, and y_Ω is the vector of quality labels for the local modeling set.

In summary, the AFBW-OLM operates in two phases: an offline phase where the FBN is constructed and all historical sample weights are precomputed using formulas (1)–(5), and an online prediction phase where the weighted similarity (6)-(8) is calculated, local samples are selected, and a weighted Gaussian process regression model (9)-(11) is used to generate the prediction for the query sample. This adaptive weighting strategy effectively suppresses the influence of noisy and unreliable data, thereby improving the accuracy and robustness of quality prediction in industrial batch processes.

The pseudocode of the AFBW-OLM algorithm is shown below.

Algorithm 1 : Adaptive Fuzzy-Bayesian Weighted Online Learning Method (AFBW-OLM)

Input: New batch sample X_{new}, Historical dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$, FBN structure G, Fuzzy states $S = \{low, medium, high\}$, Neighbor number $K_{neighbor}$, Smoothing factor α, Small constant $\epsilon = 10^{-6}$

Output: Quality prediction $\hat{y}_q$, Updated historical dataset $D_{updated}$

1: Offline Phase:

2: Fuzzify continuous features in D via Gaussian membership functions to get hard states $s(x_i)$;

3: Construct FBN (DAG) and learn CPTs with Laplacian smoothing α;

4: Calculate reliability weight m_i for each x_i via FBN joint probability

Algorithm 1 : Adaptive Fuzzy-Bayesian Weighted Online Learning Method (AFBW-OLM)

formula to form $D_w = \{(x_i, y_i, m_i)\}$;

5: Online Phase:

6: Fuzzify query sample x_q to get $s(x_q)$ (consistent with Step 2);

7: Compute normalized Euclidean distance $d(x_q, x_i)$ for all $x_i \in D_w$;

8: Calculate weighted similarity $\mathrm{sim}(x_q, x_i)$ by integrating m_i and $d(x_q, x_i)$;

9: Sort D_w by $\mathrm{sim}(x_q, x_i)$ and select top-$K_{neighbor}$ as Ω;

10: Construct weighted kernel matrix K_w for Ω based on GPR kernel formula;

11: Train weighted GPR model on Ω with optimized hyperparameters;

12: Compute kernel vector k_q between x_q and samples in Ω;

13: Predict $\hat{y}_q$ for x_q using the trained weighted GPR model;

14: Update historical dataset D with new sample $(x_q, \hat{y}_q)$ to obtain $D_{updated}$;

15: Output $\hat{y}_q$

4. Experimental results

In this section, we comprehensively evaluate the proposed Adaptive Fuzzy-Bayesian Weighted Online Learning Method (AFBW-OLM) for quality prediction in industrial batch processes.

4.1 Data source

The penicillin fermentation process, a typical fed-batch fermentation with strong time-varying dynamics, is adopted as the study case. The dataset used in this work is generated by the Pensim2.0 simulation platform[25], which is specifically designed to simulate the penicillin fermentation process (available at: http://simulator.iit.edu/web/software.html).

A total of 25 batches are generated under different initial conditions and operating profiles. Among them, 20 batches are used as the training set, and the remaining 5 batches constitute the test set. To simulate realistic industrial scenarios, the training set is contaminated with Gaussian noise, while the test set includes batches with significantly altered initial conditions (e.g., substrate concentration and reactor volume) to evaluate the generalization capability of the models. Moreover, to assess adaptability to time-varying behavior, the initial states of the test batches are periodically adjusted, thereby emulating the complex and dynamic nature of real-world batch processes.

4.2 Benchmark Methods and Comparison Strategy

To demonstrate the effectiveness of the proposed AFBW-OLM, we compare it with mainstream soft sensing models from offline and online perspectives. Offline models, representing static modeling paradigms, are trained once on the full training dataset and fixed for all predictions, including Gaussian Process Regression (GPR), Support Vector Regression (SVR), Partial Least Squares (PLS) and Multi-Layer Perceptron (MLP).

Online models generate adaptive predictions for each query sample via on-the-fly local modeling or incremental parameter updating. We compare two groups of online models to verify the fuzzy-Bayesian weighting strategy: conventional JITL integrated with the four base learners (JIT-GPR/SVR/PLS/MLP) using only Euclidean distance for similarity, and AFBW-OLM combined with the same learners that incorporates fuzzy-Bayesian confidence weights into the similarity metric. The key hyperparameter (number of similar samples for local modeling) is cross-validated and kept consistent for JITL and AFBW-OLM to ensure fair comparison.

4.3 Implementation Details

The Fuzzy Bayesian Network (FBN) at the core of AFBW-OLM is offline-constructed using the training dataset: each continuous process variable is fuzzified into low/medium/high linguistic states via Gaussian membership functions, the FBN structure is defined by prior process knowledge to encode variable causal dependencies, and Conditional Probability Tables (CPTs) are learned with Laplacian smoothing ($\alpha = 1$). The joint probability of each training sample is calculated and stored as its Bayesian confidence weight for online similarity calculation.

All base learners use optimal hyperparameters selected via 5-fold cross-validation to avoid tuning bias (e.g., hyperparameters for GPR with squared exponential kernel, SVR with RBF kernel, PLS, and MLP). Experiments are run on a Windows 10 machine (Intel i7-4790 CPU, 12 GB RAM) in Python. Two metrics evaluate model performance: Mean Squared Error (MSE, reflecting the magnitude of prediction squared error) and Coefficient of Determination (R^2 , measuring fit between predicted and true values), with calculation formulas provided below.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2 \tag{12}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{n}\left(y_i - \bar{y}_i\right)^2} \tag{13}$$

In the above equations, y_i denotes the true quality value of the i -th test sample, $\hat{y}$ is the corresponding predicted value, $\bar{y}$ represents the mean of the true quality values over the entire test set, and n is the total number of test samples. For each model, calculate two metrics in all test batches and take the average of multiple tests to ensure the statistical reliability of the comparison results.

4.4 Results and Analysis

The experimental results, visualized in Figures 2–5 and quantified in Tables 1 and 2, comprehensively validate the superiority of the proposed AFBW-OLM in both noise-intensive scenarios and cross-batch generalization for industrial batch process quality prediction.

Test Dataset 5, contaminated with elevated Gaussian noise, serves as the most stringent test for model robustness. As evidenced by the R^2 and MSE metrics in this scenario, all models experience a performance decline, yet the degree of degradation varies significantly. Offline global models (GPR, SVR, PLS, MLP) show the most notable drop in accuracy, as their fixed global structure struggles to maintain fidelity amid noisy process data. Conventional JITL variants, while more resilient due to local modeling, still exhibit noticeable performance attenuation because Euclidean distance fails to differentiate between reliable and noise-corrupted historical samples. In sharp contrast, the proposed AFBW-OLM framework, particularly AFBW-GPR, achieves the highest R^2 and the lowest MSE on Test Dataset 5. This improvement is attributed to the fuzzy-Bayesian weighting mechanism, which effectively down-weights noisy samples during local neighbor selection, thereby preserving the integrity of the local modeling set even under high noise.

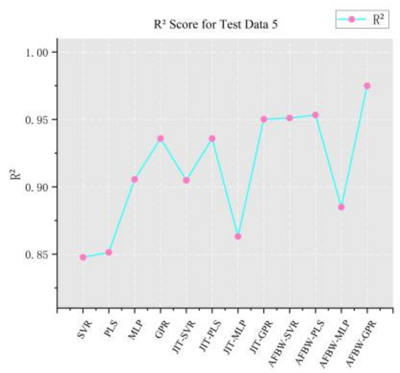

FIGURE 2. R^2 scores of different models on a batch of test data

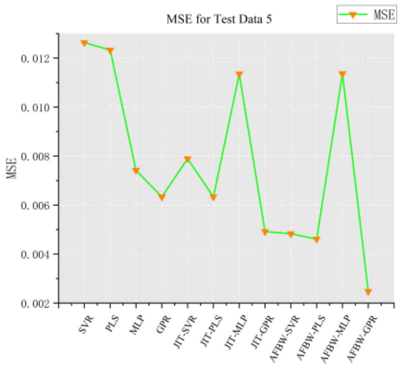

FIGURE 3. MSE of different models on a batch of test data

FIGURE 4. Comparative Analysis of Average R^2 Scores Across Models

The average performance across all five test datasets, presented in the aggregate R^2 and MSE plots, further confirms the generalizability of AFBW-OLM. Across all four base learners (GPR, SVR, PLS, MLP), the AFBW-OLM framework consistently outperforms both the standalone offline models and the conventional JITL counterparts. AFBW-GPR attains the highest average R^2 score (exceeding 0.99) and the lowest average MSE, demonstrating a substantial improvement over JIT-GPR. Tables 1 and 2 provide granular quantitative evidence, showing that AFBW-OLM outperforms the corresponding JITL variant on every individual test dataset. The performance gap is most pronounced on Test Dataset 5, where the noise-induced unreliability of historical samples is most severe, validating the adaptive weighting strategy as the key to mitigating noise interference.

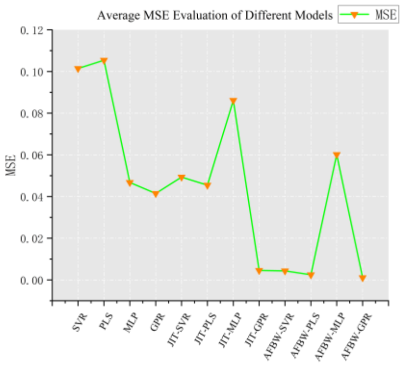

FIGURE 5. Comparative Analysis of Average MSE Scores Across Models

Collectively, these results demonstrate that the integration of fuzzy-Bayesian confidence weights into the JITL framework not only enhances prediction accuracy but also strengthens the model's robustness against noise and process variations. The consistent superiority of AFBW-OLM across different base learners further verifies that the proposed weighting mechanism is a general and effective enhancement to the just-in-time learning paradigm for industrial soft sensing.

TABLE 1. R^2 scores for different models on different batch test data

Models	*TestData1*	*TestData2*	*TestData3*	*TestData4*	*TestData5*	*Average value*
SVR	0.9203	0.8917	0.8339	0.9121	0.8477	0.8811
PLS	0.9297	0.9265	0.8502	0.9104	0.8514	0.8936
MLP	0.9614	0.9845	0.9597	0.9508	0.9056	0.9524
GPR	0.9834	0.9714	0.8901	0.9444	0.9358	0.945
JIT-SVR	0.9491	0.9312	0.9543	0.9631	0.9048	0.9405
JIT-PLS	0.9574	0.9352	0.9664	0.9644	0.9358	0.9518
JIT-MLP	0.9265	0.9418	0.9019	0.9395	0.8632	0.9146

Models	*TestData1*	*TestData2*	*TestData3*	*TestData4*	*TestData5*	*Average value*
JIT-GPR	0.9844	0.9704	0.9557	0.9651	0.9502	0.9651
AFBW-SVR	0.9794	0.9706	0.9565	0.9774	0.951	0.967
AFBW-PLS	0.9871	0.9865	0.9893	0.9864	0.9533	0.9805
AFBW-MLP	0.9607	0.9447	0.9101	0.9417	0.8849	0.9284
AFBW-GPR	0.9979	0.998	0.9959	0.989	0.9749	0.9911

TABLE 2. MSE scores for different models on different batch test data

Models	*TestData1*	*TestData2*	*TestData3*	*TestData4*	*TestData5*	*Average value*
SVR	0.1217	0.1133	0.1432	0.1163	0.0126	0.1014
PLS	0.161	0.103	0.1341	0.1166	0.0123	0.1054
MLP	0.0843	0.0023	0.0521	0.0876	0.0074	0.0467
GPR	0.0031	0.0042	0.1022	0.0917	0.0063	0.0415
JIT-SVR	0.0577	0.1011	0.0744	0.0056	0.0079	0.0493
JIT-PLS	0.0511	0.0979	0.0667	0.0054	0.0063	0.0455
JIT-MLP	0.1318	0.0932	0.0956	0.0988	0.0114	0.0862
JIT-GPR	0.0029	0.0044	0.0054	0.0053	0.0049	0.0046
AFBW-SVR	0.0038	0.0044	0.0053	0.0034	0.0048	0.0043
AFBW-PLS	0.0024	0.002	0.0013	0.0021	0.0046	0.0025
AFBW-MLP	0.0073	0.0911	0.0926	0.0984	0.0114	0.0602
AFBW-GPR	0.0004	0.0003	0.0005	0.0017	0.0025	0.0011

5. Conclusion

This paper proposed an Adaptive Fuzzy-Bayesian Weighted Online Learning Method (AFBW-OLM) to address the challenges of noise, outliers, and process drift in industrial batch process quality prediction. By integrating a fuzzy Bayesian network into the just-in-time learning framework, the method extends the conventional similarity measure from a simple spatial distance to a composite index that accounts for both physical proximity and sample reliability. The Bayesian confidence derived from the network effectively suppresses the influence of unreliable data during local model construction. Experimental results on the penicillin fermentation process demonstrate that AFBW-OLM, particularly when combined with Gaussian process regression, consistently outperforms both offline models and traditional JITL approaches across multiple test batches, achieving higher prediction accuracy and robustness under noisy and time-varying conditions. The results verify that uncertainty-aware sample weighting effectively boosts soft sensor performance for complex industrial scenarios.

Acknowledgements

This paper is supported by the Liaoning Provincial Young Scientist Fund (Category A) (formerly known as Liaoning Provincial Science Fund for Distinguished Young Scholars) (2026JH6/101100029), the Doctoral Research Start-up Project of the Natural Science Foundation of Liaoning Province (2025-BS-0198), the Liaoning Provincial Xingliao Talent Plan Project (XLYC2503093, YS2023004), Program of the Basic Research Plan, Shenyang Institute of Automation, CAS (2025JC1G03, 2025JC3K03), the Youth Innovation Promotion Association, CAS (2021203), and the National Major Special Project in the Field of Energetic Materials.

References

[1] N. Hamzah, S. A. Ali, and S. F. A. Karim, "Improvement of Process Quality via Integration of Statistical Process Control and Engineering Process Control in Batch Process." pp. 218–223.

[2] B. Alakent, “Soft sensor design using transductive

moving window learner," *Computers & Chemical Engineering*, vol. 140, pp. 106941, 2020/09/02/, 2020.

[3] W. Shao, S. Chen, and C. J. Harris, "Adaptive Soft Sensor Development for Multi-Output Industrial Processes Based on Selective Ensemble Learning," *IEEE Access*, vol. 6, pp. 55628–55642, 2018.

[4] K. Liu, X. Zhao, M. Mou, and Y. Hui, "Quality prediction of multi-stage batch process based on integrated ConvBiGRU with attention mechanism," *Applied Intelligence*, vol. 55, no. 2, pp. 123, 2024/12/10, 2024.

[5] Q. Jiang, Z. Wang, S. Yan, and Z. Cao, "Data-Driven Soft Sensing for Batch Processes Using Neural Network-Based Deep Quality-Relevant Representation Learning," *IEEE Transactions on Artificial Intelligence*, vol. 4, no. 4, pp. 602–611, 2023.

[6] S. Zhang, Y. Chen, W. Zhang, and R. Feng, "A novel ensemble deep learning model with dynamic error correction and multi-objective ensemble pruning for time series forecasting," *Information Sciences*, vol. 544, pp. 427–445, 2021/01/12/, 2021.

[7] B. Sagi, and T. Thyagarajan, "Data-driven soft sensor for online composition estimation with adaptive-window regression and QP-constrained kalman estimator," *ISA Transactions*, 2025/12/16/, 2025.

[8] Y. Shiraki, Y. Nakayama, S. Natori, K. Suda, Y. Ono, and H. Kaneko, "Adaptive soft sensor considering process state in film manufacturing process and identification of critical process variables," *Results in Chemistry*, vol. 9, pp. 101677, 2024/07/01/, 2024.

[9] X. Shengjie, X. Li, and X. Liang, "Soft Sensor Development Based on Deep Extended Variational Autoencoder with Just-in-time Learning." pp. 1215–1220.

[10] J. Chen, J. Chen, and Z. Zhong, "A weight clustering algorithm based on sliding window model for stream data," *Scientific Reports*, vol. 15, no. 1, pp. 22521, 2025/07/02, 2025.

[11] W. Zheng, Y. Liu, Z. Gao, and J. Yang, "Just-in-time semi-supervised soft sensor for quality prediction in industrial rubber mixers," *Chemometrics and Intelligent Laboratory Systems*, vol. 180, pp. 36–41, 2018/09/15/, 2018.

[12] H. Jin, X. Chen, J. Yang, L. Wang, and L. Wu, "Online local learning based adaptive soft sensor and its application to an industrial fed-batch chlortetracycline fermentation process," *Chemometrics and Intelligent Laboratory Systems*, vol. 143, pp. 58–78, 2015/04/15/, 2015.

[13] C.-l. Mei, Y. Chen, H. B. Jiang, Y. Ding, X. Chen, and G. J. Liu, "Just-in-time modeling with a combination of input and output similarity criterions for soft sensor modeling in fermentation processes," *Chemical engineering transactions*, vol. 61, pp. 1045–1050, 2017.

[14] H. Yu, S. Yin, and H. Luo, "Robust Just-in-time Learning Approach and Its Application on Fault Detection," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 15277–15282, 2017/07/01/, 2017.

[15] A. Bifet, and R. Gavalda, "Learning from time-changing data with adaptive windowing." pp. 443–448.

[16] M. M. Dornelles, and N. S. T. Hirata, "Selection of Windows for W-Operator Combination from Entropy Based Ranking." pp. 64–71.

[17] Q. A. Wang, J. Chen, Y. Q. Ni, Y. F. Xiao, N. B. Liu, S. K. Liu, and W. S. Feng, "Application of Bayesian networks in reliability assessment: A systematic literature review," *STRUCTURES*, vol. 71, JAN, 2025.

[18] M. Grzegorczyk, "Being Bayesian about learning Bayesian networks from ordinal data," *INTERNATIONAL JOURNAL OF APPROXIMATE REASONING*, vol. 170, JUL, 2024.

[19] I. K. Baek, S. H. Lee, Y. H. Jang, H. Park, J. Kim, S. W. Cheong, S. K. Shim, J. G. Han, J. K. Han, G. S. Jeon, D. H. Shin, K. S. Woo, and C. S. Hwang, "Implementation of Bayesian networks and Bayesian inference using a Cu0.1Te0.9/HfO2/Pt threshold switching memristor," *NANOSCALE ADVANCES*, vol. 6, no. 11, pp. 2892–2902, MAY 29, 2024.

[20] L. M. Zhang, X. G. Wu, Y. W. Qin, M. J. Skibniewski, and W. L. Liu, "Towards a Fuzzy Bayesian Network Based Approach for Safety Risk Analysis of Tunnel-Induced Pipeline Damage," *RISK ANALYSIS*, vol. 36, no. 2, pp. 278–301, FEB, 2016.

[21] T. Y. Xiong, Z. P. Liu, and M. H. Zhang, "Evaluating the efficacy of fuzzy Bayesian networks for financial risk assessment," *DEMONSTRATIO MATHEMATICA*, vol. 58, no. 1, MAR 13, 2025.

[22] M. Ghazizadeh-Ahsaee, M. Naghibzadeh, and B. S. Gildeh, "Learning parameters of fuzzy Bayesian Network based on imprecise observations," *Int. J. Know.-Based Intell. Eng. Syst.*, vol. 18, no. 3, pp. 167–180, 2014.

[23] H. Khalfaoui, A. Azmani, A. Farchane, and S. Safi, "Symbiotic Combination of a Bayesian Network and Fuzzy Logic to Quantify the QoS in a VANET: Application in Logistic 4.0," *Computers*, 12, 2023].

[24] J. M. Kościelny, M. Bartyś, and A. Sztyber, "Diagnosing with a hybrid fuzzy–Bayesian inference approach," *Engineering Applications of Artificial Intelligence*, vol. 104, pp. 104345, 2021/09/01/, 2021.

[25] G. Birol, C. Ündey, and A. Çinar, "A modular simulation package for fed-batch fermentation: penicillin production," *Computers & Chemical Engineering*, vol. 26, no. 11, pp. 1553–1565, 2002/11/15/, 2002.