Data-Based Fault Detection in Chemical Processes: Managing Records with Operator Intervention and Uncertain Labels

Mahdieh Askarian[a,c], Raul Benítez[b], Moises Graells[c], Reza Zarghami[a]*

[a] School of Chemical Engineering. College of Engineering, University of Tehran. PO Box 11155-4563, Tehran, Iran

[b] Automatic Control Department. Universitat Politècnica de Catalunya. EUETIB, Comte d'Urgell 187, 08028-Barcelona, Spain

[c] Chemical Engineering Department. Universitat Politècnica de Catalunya. EUETIB, Comte d'Urgell 187, 08028-Barcelona, Spain

ABSTRACT

Developing data-driven fault detection systems for chemical plants requires managing uncertain data labels and dynamic attributes due to operator-process interactions. Mislabeled data is a known problem in computer science that has received scarce attention from the process systems community. This work introduces and examines the effects of operator actions in records and labels, and the consequences in the development of detection models. Using a state space model, this work proposes an iterative relabeling scheme for retraining classifiers that continuously refines dynamic attributes and labels. Three case studies are presented: a reactor as a motivating example, flooding in a simulated de-Butanizer column, as a complex case, and foaming in an absorber as an industrial challenge. For the first case, detection accuracy is shown to increase by 14% while operating costs are reduced by 20%. Moreover, regarding the de-Butanizer column, the performance of the proposed strategy is shown to be 10% higher than the filtering strategy. Promising results are finally reported in regard of efficient strategies to deal with the presented problem.

KEYWORDS: fault detection, mislabeling, label noise, underlying states, operational intelligence, interactive learning.

* Corresponding author. Tel.: +34 93 413 72 75.

Emails: mahdiehaskarian@ut.ac.ir (M. Askarian), raul.benitez@upc.edu (R. Benitez), moises.graells@upc.edu (M. Graells), rzarghami@ut.ac.ir (R. Zarghami)

1. INTRODUCTION

Monitoring of chemical processes, as many other activities, is required for determining the need for corrective actions and subsequent efficient operation. Indeed, abnormal situation management (ASM) is an essential task for loss prevention and safe operation of chemical plants. To achieve this aim, multiple protection layers (Fig. 1) are applied in industrial plants as per the international standard IEC61511 (2003), each one consisting of equipment and/or administrative controls coordinated with other protection layers (Isermann, 1994). Most automatic protection layers are triggered by actuators, while sensors' readings indicate the violation of limiting thresholds. Despite all the progress in automatic risk reduction systems (hardware and software), the operator supervision and the corrective action in ASM is still indispensable. Indeed, the operator and the automatic controls, together, constitute operational intelligence (Rajaram & Jaikumar, 2000).

The fault detection (FD) system is a core component of ASM that has attracted a lot of attention recently. Moreover, it is expected to be explicitly included in the standard IEC61511 in the near future. In fact, a fault consists of an unpermitted deviation of at least one property or parameter of a system from its acceptable, usual or standard condition (Isermann & Balle, 1997). FD methods are categorized in three main groups: quantitative model-based methods, qualitative model-based methods and data-driven methods (Venkatasubramanian, Rengaswamy, Yin, & Kavuri, 2003). Qualitative model-based FD methods are not often deployed for complex chemical process, because the corresponding analytical description is rarely available. In addition, quantitative model-based FD methods, so-called inference methods, are developed based on explicit structural knowledge and causalities (Korbicz, Koscielny, Kowalczuk, & Cholewa, 2012). These methods, which rely on experts' knowledge in a specific domain, are often costly and time-intensive to obtain. Thus, FD is commonly addressed by process history based methods, since for operating plants a large amount of historical process data is available (MacGregor & Cinar, 2012; Qin, 2012).

Data-driven FD systems early developed based on multivariable statistical analysis e.g. principal component analysis (PCA) (Lu, Yao, Gao, & Wang, 2005), partial least squares (PLS) (Chiang, Braatz, & Russell, 2001). Recently, FD has been considered as a classification problem as well, and Machine Learning provides various tools for classification, which are categorized below (Isermann, 2006):

- **Geometric classifier**. e.g. *k*-nearest neighbourhood (*k*NN) (Pandya, Upadhyay, & Harsha, 2013);

- **Probabilistic classifier**. e.g. Gaussian naïve Bayes (GNB) (Askarian, et al., 2015; Sáez-Atienzar, et al., 2015) and the hidden Markov model (HMM) (Li, Fang, & Huang, 2015);

- **Approximation classifier**. e.g. polynomial support vector machines (SVM) (Danenas & Garsva, 2015; Namdari & JazayeriRad, 2014);

- **Soft computing techniques**. e.g. fuzzy classifier (Serdio, Lughofer, Pichler, Buchegger, & Efendic, 2014) and artificial neural networks (Duda, Hart, & Stork, 2001).

The main advantage of FD using classification methods is the ability at dealing with unstructured information and implicit knowledge. However, each method poses some limitations that are discussed in detail by Isermann (2006). The major weak point of classifiers is vulnerability to mislabeling, which is the issue explored in this work.

In Machine Learning, the standard approach consists in training a classifier from a labeled dataset to predict the class of new samples accordingly. Usually, labels are considered given and the labeling process is assumed to be reliable (Bootkrajang & Kabán, 2012). However, in industrial practice and process plants, assigning labels to training data may need attention and careful examination. Indeed, true labels corresponding to the state of system are usually unavailable. Mislabeling may occur for several reasons including expert errors, lack of information or data labeling by non-experts (de França & Coelho, 2015). Label uncertainty is an important issue in classification, because most classifiers are built on the hypothesis of a perfectly labeled training set. Some Machine Learning literature exist regarding effects of uncertain labels, which shows that mislabeling may detrimentally affect the classification performance and the reliability of the learned models (Brodley & Friedl, 1999; Frénay, de Lannoy, & Verleysen, 2011).

Numerous methods have been proposed to deal with label noise. Filter approaches aim at identifying and removing any mislabeled instances (Brodley & Friedl, 1999; Zhang, Li, Yang, & Yong, 2014). A residual-based fault detection is developed which solely relies on sensor's data; and labels required for pattern recognition of fault is not demanded

(Serdio, Lughofer, Pichler, Buchegger, Pichler, et al., 2014). Some algorithms are naturally robust to label noise (Teng, 2005). Some methods have been modified to take label noise into account in an embedded fashion or to model the process of label corruption as part of modeling the data (Swartz, Haitovsky, Vexler, & Yang, 2004). The current literature on learning with label noise is a lively mixture of theoretical and experimental studies that clearly demonstrate both the complexity and the importance of the problem. Frénay and Verleysen (2014) comprehensively discussed different families of algorithms that have been proposed to deal with label noise.

A particular case of mislabeled training data occurs while operator-process interactions are involved in industrial practice. In case of abnormal events in chemical plants, operators take preventive actions as early as possible. On the one hand, one of such actions could be taken as an indication of the fault and used for training an automatic fault detection system. On the other hand, human error and lack of information about the underlying state of the system can mislead the operator, so that an unnecessary action is taken. Furthermore, the operator recognition (normal, symptom, fault…) strongly depends on his or her personal characteristics and preferences (conservative / risky, experienced / untrained etc.). Although the operator is the ultimate decision maker regarding ASM, it is an actor interfering the process history, as well. Such a case is of special complexity and it is referred as mislabeling, label uncertainty and label noise, as addressed in the Machine Learning literature.

In addition, the dynamic response of a chemical process following a fault and/or an operator action increases the complexity of situation (Si, Hu, Yang, & Zhang, 2011). Unnecessary actions alter process history and have adverse impacts on attributes of faulty and normal situations. Of course, plant decision makers are trading-off the cost of the action and the expected loss, and experimenting seems not an option. Nevertheless, incorrect interpretation of the system state by the operator causes mislabeling, and his or her unnecessary action impacts on the attributes.

It is worth to mention that most computer science literature dealing with label noise interprets noise as an outlier (Frénay & Verleysen, 2014). In other words, mislabeling is considered for each instance of attributes. This strategy is appropriate for static problems such as image processing. However, fault detection in process systems requires a

different approach due to its dynamic behavior. This important aspect has received scarce attention in chemical engineering, and it deserves more investigation.

These challenges motivate the development of a robust strategy for efficiently using the training datasets with uncertain labels produced in operating plants by human intervention. This strategy should be regarded as a part of a general operators' decision support framework for FD and ASM. Since the main objective is aiding the operator to make better decisions, operator actions cannot be readily identified as true faulty labels. This will lead to a model of the operator preferences that could anticipate the operator actions, not the real process faults. Conversely, questioning the actions and the labels is the starting point of the strategy proposed in this work.

This study mainly discusses about impact of human intervention on FD of a dynamic system, which is an unattended topic in the chemical engineering literature, and contributes a novel strategy to deal with this situation. The proposed FD approach includes a loop for fault detection via three parties -classifier, state space model and operator- which successively leads to evolution. The underlying state of the system is estimated using a state space model, which takes the advantage of sequential data analysis of time series. Then, the deviation from the normal state is characterized by the Mahalanobis distance, which eventually allows relabeling attributes. Iterative training of the Gaussian naïve Bayes classifier is proposed to improve the operator perception and refine attributes. The solution approach provides useful tools for the plant operator to assess the need of preventive actions respect to faults. Next, this strategy is validated and discussed in three case studies, a reactor, as an illustrative example; the de-Butanizer column, as a complex example; and an industrial sweetening unit in a gas refinery as a real challenge.

2. ILLUSTRATIVE EXAMPLE

A continuous stirred tank reactor (CSTR) is a most basic unit operation in chemical engineering which is a tank equipped with an impeller. Reactants and products are continuously added and withdrawn to the tank where reaction takes place. Mixing with impeller is required to achieve uniform composition. A fault and the subsequent operator action are simulated in a jacketed, non-adiabatic, and perfectly mixed tank reactor. A

single first-order exothermic and reversible reaction of two components, A↔B, takes place. A scheme of the equipment module is shown in a Figure 2, as well as input and output variables. Volume is assumed to be constant, and inlet and outlet flowrates are equal. Density, specific heat and heat of reaction are also assumed constant. Based on the mass and energy balances, equations in appendix A.1 are derived for simulation.

The change of the feed stream concentration or temperature (input variables u_{1t} or u_{2t}, respectively) causes the change of the outlet composition and temperature. In order to keep the reactor temperature under a specified threshold (320 ºC) the operator can decrease the jacket temperature u_{3t}. Although, this problem can be easily solved by using a proportional-integral-derivative (PID) controller, a manual control by the operator is discussed to illustrate the problems associated to human decision-making in industrial processes. In other words, operator-process interactions are required in open loop systems. The input variables are assumed unobservable to the operator; the only information that can be monitored and recorded is that of the output variables. Furthermore, the process model is assumed unavailable to the operator, who can only rely on the measurement of the reactor temperature to keep it below the limit.

Scenario 1: C_{Af} has a step change that is considered a fault in the operation. If the operator is unaware of the state of system, the reactor temperature will exceed the limit (Fig. 3 red line). Therefore, when T increases too fast (> 0.2 ºC/s), the operator decreases T_j, so that T is kept under 320˚C (Fig. 3 blue line).

Scenario 2: A minor step change in C_{Af} also increases T too fast, but it remains under the specified threshold (Fig. 4 red line). Therefore, it is called pseudo-fault. Since C_{Af} is not observable, the conservative operator takes an action based on his or her intuition (experience of Scenario 1), although this time it is unnecessary. This is an overuse of cooling utility and extra cost. On the other hand, the operator labels the process data of this time interval as faulty. In addition, the preventive action changes the trend of attributes (Fig. 4 blue line).

Hence, a robust FD system is expected to reveal pseudo-fault as normal in order to prevent unnecessary and expensive action. It is clear that using this raw information for designing a classifier will lead to a low FD performance. This performance is quantified

in Section 4.1. Although various scenarios can be defined, the scope of this illustrative example is limited to decision making by a conservative operator.

The presented case study leads to a twofold problem:

- Avoiding bad decisions in case of pseudo-fault produced the extra cost of unnecessary preventive actions;
- Avoiding uncertain labels and attributes that will mislead the automatic diagnosis system.

3. METHODOLOGY

The initial FD system, which is designed based on basic information (namely attributes labeled by the operator), infers the state of the system in a way very similar to operator intuition. In other words, careless use of this information leads to model the pattern of operator behavior and preferences. Thus, this cannot support successful decision in regard of the optimization of the costs corresponding to unnecessary preventive actions or potential accidents. An improved strategy is required to deal with mislabeling and the subsequent impact of operator actions on attributes (system outputs).

In order to improve FD performance, retraining with a refined dataset including reliable labels and attributes is inevitable. In general, improving operator perception is required to provide a chance of producing new datasets. In hindsight and without operator actions, it is often much clearer whether the state of system is a pseudo-fault or a risky fault. However, the operator is not aware of the underlying state of system, and taking no action is risky when perceiving a potential fault. Indeed, taking no action allows further learning, but may lead to an accident. On the other hand, the performance of the initial FD system is not reliable enough to make decisions based on its prediction. Thus, a gradual improvement of the FD system can be achieved by an interactive and iterative learning procedure with an acceptable risk at each phase.

Figure 5 illustrates the interactive strategy proposed for training a classifier, which can provide a promising decision support system. It consists of three main parts shown by dashed bounding boxes: initializing, re-training, and validating. Rectangular and elliptical blocks represent available data and operations on data, respectively. The dotted elliptical blocks are flexible and can accept different tools. A grey block represents that it may be unavailable in some cases. In following sub-sections, the fundamentals of the most

significant blocks involved in this work are explained. They are included the dynamic linear model (DLM) to estimate the underlying state; Mahalanobis distance to characterize deviations from normal state, Gaussian naïve Bayes (GNB) to classify, a validation confusion matrix to evaluate performance, and an interaction index to forward to next interactive learning phases. Then, input-output information of these tools is demonstrated. Finally, the rules and interaction of these tools are described in the general strategy.

3.1. Description of tools

3.1.1. State space models

This work proposes to take advantage of the state space model to analyze time series and estimate the underlying states of a dynamic system. Given m observations, $Y_t = (y_{1t}, y_{2t}, \ldots, y_{mt})$, the n unobservable state variables, $\theta_t = (\theta_{1t}, \theta_{2t}, \ldots, \theta_{nt})$, are recursively computed using a state space model. Figure 6 represents the information flow of the state space model, which satisfies the following assumptions:

- θ_t is a Markov chain. In other words, θ_t and $(\theta_{0:t-2}, Y_{1:t-1})$ are conditionally independent given θ_{t-1};
- Y_t's are independent conditioned on θ_t; and Y_t depends on θ_t only (Petris, Petrone, & Campagnoli, 2009).

The dynamic linear model (DLM) is a special case of a general state space model, being linear and Gaussian. A DLM is specified by a pair of observation equation and state equation:

$$Y_t = F_t \theta_t + v_t \qquad v_t \sim N_m(0, V_t) \tag{1}$$
$$Y_t \in R^{m \times 1} \qquad F_t \in R^{m \times n} \qquad v_t \in R^{m \times 1} \qquad V_t \in R^{m \times m}$$

$$\theta_t = G_t \theta_{t-1} + w_t \qquad w_t \sim N_n(0, W_t) \tag{2}$$
$$\theta_t \in R^{n \times 1} \qquad G_t \in R^{n \times n} \qquad w_t \in R^{n \times 1} \qquad W_t \in R^{n \times n}$$

where G_t and F_t are system identification matrices and v_t and w_t are two independent sequences of independent Gaussian random vectors with mean zero and variance matrices V_t and W_t, respectively. Maximum likelihood estimation (MLE) is a common approach for parameter identification of the model (Petris, et al., 2009).

The current value of the state variables can be recursively estimated based on the attributes up to time t. In a DLM, the Kalman filter allows updating the current inference on the state variables as new data become available (Lendek, Babuška, & De Schutter, 2008). Passing from the filtering density $\pi(\theta_t|Y_{1:t})$ to $\pi(\theta_{t+1}|Y_{1:t+1})$ is fully detailed described in appendix A.2.

3.1.2. Mahalanobis distance

The Mahalanobis distance, D_M, of a vector $\theta_t =(\theta_{1t}, \theta_{2t}, …, \theta_{nt})^{\mathrm{T}}$ from another group of state variables, with mean $\mu=(\mu_1, \mu_2… \mu_n)^{\mathrm{T}}$ and covariance matrix, Σ, is defined as (Yu, 2013):

$$D_M (\theta_t) = \sqrt{(\theta_t - \mu)^T \Sigma^{-1}(\theta_t - \mu)} \tag{3}$$

3.1.3. Gaussian naïve Bayes classifier

Gaussian naïve Bayes (GNB) classifier is a supervised learning method based on Bayes' theorem that can be applied for fault detection. The probability of each fault as a class can be determined regarding attributes as below (Mehranbod, Soroush, Piovoso, & Ogunnaike, 2003):

$$\pi\left(F_t|Y_t\right) = \frac{\pi\left(Y_t|F\right)\pi\left(F\right)}{\pi\left(Y_t\right)} \tag{4}$$

where $\pi(Y_t/F)$ is the probability of attributes conditioned on the fault F and $\pi(F)$ is the a priori probability, which gives the probability of the class before measuring any attributes. The divisor $\pi(Y_t)$ is merely a scaling factor to assure that a posteriori probabilities are normalized.

3.1.4. Key performance indicators

The performance of classifiers is evaluated by comparing predicted faults and true faults. In this order, FD outcomes are arranged in a validation confusion matrix (Monroy, Villez, Graells, & Venkatasubramanian, 2012) presented in Table 1.

Table 1: Validation confusion matrix

		True label	
		F	¬F
Predicted label	F	a	b
	¬F	c	d

where a is the number of samples corresponding to faulty situations and diagnosed as such (true positive); b is the number of samples diagnosed as faulty but were not (false positive); c is the number of samples corresponding to faulty but not diagnosed situations (false negative) and d is the number of samples not happened and not diagnosed (true negative). Then, performance indexes including accuracy, $F1$, and true positive rate (TPR) are calculated as follows:

$$Accuracy = \frac{a+d}{a+b+c+d} \tag{5}$$

$$F1 = \frac{2a}{2a+b+c} \tag{6}$$

$$TPR = \frac{a}{a+c} \tag{7}$$

3.1.5. Interaction index

Due to the unavailability of true labels in real practice, labels estimated by an operator and a classifier play some roles in interactive learning of the FD system described in the subsequent sections. As such, a comparison of classifier prediction and labels assigned by the operator based on his or her intuition is made by an interaction confusion matrix in Table 2.

Table 2: Interaction confusion matrix

		Operator label	
		F	¬F
Predicted label	F	a'	b'
	¬F	c'	d'

where a' is the number of samples to which the operator and the classifier assign fault labels to the state of a system; b' is the number of samples diagnosed as faulty in contrast with operator opinion; c' is the number of samples corresponding to faulty but not

diagnosed situations; and *d'* is the number of samples for which both the operator and the classifier agree that no fault happens.

It is generally agreed that the operator and the automatic FD system are complementary for ASM (Sheridan & Johannsen, 1976). However, it is required to take into account the conflict between operator labels and classifier predictions to manage uncertain records. For instance, a conservative operator hastily assigns a fault to the system regarding any minor disturbance or so called pseudo-fault. While the classifier predicts this situation as the normal state, it is reflected in *c'*. On the other hand, a risky operator may not react to a faulty situation which is diagnosed as such and is categorized as *b'*.

The main idea is to gradually encourage an operator to provide a chance of generating more informative dataset, i.e. new trends of attributes in subsequent learning phases. It is required to determine to which extent the operator takes no action in potentially faulty situations at each phase. Thus, an interaction index is introduced in Eq. 8, which quantifies the conflict.

$$I_{interaction} = (\frac{d'}{b'+d'} - \frac{a'}{c'+a'})(Accuracy \times ARL)$$

(8)

where the first and second terms reflect extremes of risky and conservative operators conflicts with a classifier prediction, respectively; and a correction factor consists of accuracy of a classifier and *ARL* corresponding to acceptable risk level of a chemical plant. Indeed, the risk level is defined as multiplication of probability and consequence of a faulty situation (Bao, Khan, Iqbal, & Chang, 2011). An acceptable risk level depends on the type of chemical plants and the policy of owners. Moreover, note that the various classifiers facing ideal datasets (certain labels) have different performance levels. Designing an FD system with a high performance classifier, the conflict (i.e. *c'* and *b'*) reflects in an interaction confusion matrix is mainly due to wrong human perception. Thus, accuracy of classifier is a correction factor of $I_{interaction}$.

Based on the proposed procedure in Section 3.2, higher $I_{interaction}$ encourages more change in attributes of datasets in the next learning phase. In fact, this index is the extent of changes between learning phases. For example, involving a conservative operator leads to higher *c'*; and higher $I_{interaction}$ subsequently. If the classifier has high performance, the available interaction confusion matrix is more reliable, and keeps $I_{interaction}$ high. On the other hand while the chemical process deals with high risk, *ARL* is low. As such, $I_{interaction}$

is adjusted accordingly to specify the acceptable change ratio in the next learning phase. In general, $I_{interaction}$ is a dynamic index which is updated in any learning phase, and specifies speed of learning.

Finally, a general scheme of these significant blocks and their input-output information is represented in Figure 7. Parameters involved in each block as a part of a data-based FD system are fitted based on subsets of a typical dataset.

3.2. Iterative and interactive procedure

The operator continuously needs to make decisions, as well as the training data set can be continuously refined. However, in order to make a quantitative assessment, the proposed strategy (Fig. 5) is evaluated in different discrete phases in an offline manner. The dataset at each phase, D_p, includes attributes of the process system, $Y_{1:T}^{rp}$, and paired labels, $F_{1:T}^{rp}$, as below:

$$D_p = \left\{ (Y_{1:T}^{rp}, F_{1:T}^{rp}) \mid r = 1:R \right\} \qquad p=0,1,2,\ldots,\ P \qquad (9)$$

where T is the time horizon of the rth run; and R is the number of runs included in the pth phase. Total number of phases, P, depends on the required iterations. Each phase of data is spilt into training and testing subsets.

Step 1: The initial dataset, $D_i|i=0$, is a collection of R runs paired with operator labels. Note that in the initial learning phase, the only information source regarding labels is operator. After training the classifier, the predicted labels of the testing subset are compared with operator labels (Table 2). In addition, based on the comparison of predicted labels with true labels (Table 1), the key performance indicators (KPIs) (Eqs. 5-6) are evaluated. It should be noted that $Y_{1:T}^{rp}$ and paired true labels of the illustrative case study (CSTR) are originated from the process model. Since it is not generally applicable, the process model block in Figure 5 is in grey.

Step 2: The conflict between operator labels and GNB prediction is reflected in $I_{interaction}$ index (Eq. 8). In the next phase, operator action is taken or not in regard of this ratio, so that there is a chance to produce a new dataset at the expense of an acceptable risk level. In other words, $D_i|i=1$ includes $R \times I_{interaction}$ runs without action and $R \times (1 - I_{interaction})$ runs with action when potential faulty situations are faced.

Step 3: Parameters of the state model, DLM, are identified using MLE. In fact, DLM identification is based on the normal samples predicted by GNB, which is trained in the previous learning phase. Then, each attribute sample of $D_i|i=1$, Y_t^{rp}, is recursively projected to $\theta_t^{rp} = (\theta_{1t}, \theta_{2t}, \theta_{3t})$ using the trained state model to explore the underlying state of the system.

Step 4: Mahalanobis distance, D_M, of each sample of the three dimensional state trajectories from the predicted normal subset is evaluated. The points beyond a specified threshold (median of all points) are labeled as faulty and others are labeled as normal.

Step 5: After retraining the classifier based on $D_i|i=0,1$, the labels predicted for the testing subset.

Step 6: Gradual refining of D_i is achieved by iteratively repeating steps 2-5.

Finally, it is worth to note that various strategies dealing with mislabeling (Frénay & Verleysen, 2014) may be investigated and compared. However, this framework only focuses on the impact of human intervention on FD of a dynamic system, which is a new topic in the chemical engineering literature. In addition, fault prognostic is beyond the scope of this work.

4. RESULTS AND DISCUSSION

4.1. CSTR

In Section 2, the CSTR case study has been introduced to illustrate some problems posed to FD by the operator-process interaction. In this section, the proposed data-driven algorithm is validated on the database generated by the CSTR model. The time horizon of each run is considered 400 s, while the step change (fault and pseudo-fault) happened at $t=200$ s for both scenarios (Figs. 3-4). In the simulation, random noise (in range of $\pm 2\%$) was added to the input variables and propagated to the output variables. In addition, the conservative operator takes preventive actions while symptoms of a potential fault appear. Thus, operator labels of samples after action are all considered faulty.

The initial dataset of the CSTR case study is a set of runs as defined below:

$$D_0 = \left\{ (Y_{1:400}^{r0}, F_{1:400}^{r0}) \mid r = 1:200 \right\} \tag{10}$$

D_0 is spilt into 70:30 ratio for training and testing the classifier. It is notable that true labels of samples after action regarding Scenario 1 are faulty, while regarding Scenario 2 are normal.

The performance of the trained GNB classifier regarding D_0 is assessed in terms of $F1$ and accuracy. The result reveals that the initial training phase seems inefficient for FD due to low discrimination performance (Table 3). Since the underlying state of system is hidden, operator-process interaction is under uncertain condition. Therefore, unnecessary preventive actions lead to mislabeling and masked attributes. In addition, based on the validation confusion matrix, the extra cost of utility consumption is proportional with the false positive rate (FPR) as follows:

$$FPR = \frac{b}{b+d} \qquad (11)$$

which means that the operator wrongly distinguishes the situation as faulty and unnecessarily reacts. In fact, this initial FD system is not useful for reducing the excess cost. The main reason is that GNB emulates the operator behavior, but not the process at this iteration.

Table 3: FD performance using iterative procedure

	Learning phase	Initial	1st	2nd	3rd	4th
	$I_{interaction}$ %	0	5	13	21	32
$F1$	Normal	0.82	0.89	0.92	0.93	0.94
	Fault	0.68	0.77	0.81	0.84	0.85
Accuracy		0.77	0.86	0.89	0.90	0.91
TPR %		99.16	98.58	98.81	98.95	99.07
FPR % ∝ Unnecessary utility consumption		30.26	17.79	13.17	11.46	9.51
FNR % ∝ Accident		0.84	1.42	1.19	1.05	0.93

In order to improve FD performance, it is essential to provide a successive refinement of the datasets, including reliable labels and comprehensive attributes. First, the datasets are gradually enriched by generating new trends of attributes. Then, labels are automatically assigned to skip human error. This goal is achieved by following steps 2-4 of the proposed procedure (Section 3.2). Thus, the operator behavior toward the process is systematically changed by considering $I_{interaction}$. Because it is assumed that the operator is conservative, b' is equal to zero; and the conflict of the operator and the classifier is

limited to c' (Table 2). Moreover, for sake of simplicity, the correction factor in Eq. 8 is considered as one herein. Note complementary scenarios, investigation the effects of *ARL* and classifier accuracy, can be explored in further works; and it is not in line with the main goal of the present study. Here, operational intelligence intends to improve process operation in terms of reducing unnecessary actions. In this case study, for the first phase of retraining, $Y_{1:400}^{r1}$ consists of 190 random runs with action and 10 random runs without action (95:5 ratio based on $I_{interaction}$). The normal subset is separated by GNB which is trained in the initial phase. Then, this subset is used for identification of DLM by MLE in the first learning phase. Relabeling the new dataset is accomplished by exploring the underlying state of the system. In other words, each sample, $Y_t^{r1} \in R^{3\times1}$, is projected to $\theta_t^{r1} \in R^{3\times1}$. After projecting $Y_{1:400}^{r1}$ to three-dimensional state trajectories, D_M is calculated for each sample. Most potential faulty samples of Scenarios 1 and 2 can be discriminated by a median threshold (Fig. 8). Therefore, $F_{1:400}^{r1}$ consists of fault label for each sample that exceeds the threshold and normal label for others.

The GNB classifier is retrained and tested by 70:30 of available information, $\{D_0, D_1\}$ respectively. Table 3 shows that significant improvement in terms of $F1$ and accuracy is achieved via the first learning phase, which leads to a reduction of utility consumption as well. On the other hand, the accident rate is proportional with the false negative rate (FNR) as follows:

$$\text{FNR} = \frac{c}{a+c} \tag{12}$$

The accident rate increases, because the operator takes a risk by not responding to some potential faulty situations. Indeed, a cost for providing comprehensive information and learning is quite reasonable; this is reflected in the accident probability in this case study.

It is notable that the conflict between the operator and the classifier increases in this retraining phase in which $I_{interaction}$ is equal to 13%. Indeed, the proposed FD algorithm is successful in deviating from the prior operator perception and increasing the accuracy of prediction simultaneously. In other words, the FD system classifies the pseudo-fault as the normal state, despite the operator idea at the initial phase. It is mainly due to reflection the process state rather than the pattern of operator behavior.

There is a promotion of iteratively continuing this procedure to refine datasets and improve FD performance. In general, using two automatic discrimination tools in parallel (DLM and GNB) is complementary, since it reduces misclassification and results more reliable for the operator. In fact, the proposed methodology consists in a loop for assigning labels via three parties (classifier, state space model and operator), which leads to evolutionary improvement of the main learning elements: the data-based models and the understanding of the operator. As such, the operator perception regarding the state of system is more realistic after this interactive learning process. Table 3 represents the accuracy of discrimination of fault by FD system increases in 2nd to 4th retraining phases. In other words, the sole classifier is not efficient (note the initial accuracy), and the proposed strategy improves FD accuracy. In addition, unnecessary utility consumption has decreasing trend, which reveals that the last FD system successfully supports the operator in making decisions that are more profitable. In addition, the negative accident rate in the 2nd to 4th retraining phases is a promising result. Although the first retraining phase is more vulnerable to accidents, it provides a chance to take more information about the response of the system. In fact, the chance of more wrong decisions in the initial learning phase is possible. Thereafter, this prior information is useful for the discrimination of faults and the prevention of accidents (or any unwanted situation).

4.2. Detection of Flooding in a Column

4.2.1. Flooding in de-Butanizer column

The goal of this part is to evaluate the FD system by addressing a common reported fault of the chemical industry in a simulation environment. Distillation is a widespread unit operation, which is used to separate some components in a feed stream. Figure 9 illustrates the basic components of a distillation system i.e. De-Butanizer.

The vapor from the reboiler flows up the column, countercurrent to the liquid flowing down the column. The components in the feed stream (No. 1-2 in Fig. 9) are separated according to their relative volatilities. Butane is the most volatile component and tends to concentrate in the vapor flowing up the column. Components with lower volatility tend to concentrate in the liquid flowing down the column. Eventually, the vapor enriched in Butane exits from the top of the distillation column, and after passing the condenser is

collected in the reflux drum. A portion of the condensed liquid is fed back into the column. The rest of the liquid exits the process as the top product (Ludwig, 1997).

Flooding is a common abnormal process condition that leads to poor separation and off specification products (Kister, 2006). In runaway flooding, the liquid level continues to rise, and if it is not stopped, the column can actually overflow. Furthermore, flooding causes overpressure in the column, which potentially leads to burst of the rupture disc and to have process downtime. Less frequently, vapor slugging through the liquid also cause trays uplift and damage.

Flooding may happen due to an excess of hydraulic traffic across the column which is represented in Figure 10. Usually, one cause of flooding is too much reflux rate which interrupts mass balance. Since reflux is assumed as a heat sink, excess reflux also interrupts energy balance and disrupts temperature profile. Consequently, it leads to condense of vapor and increase of liquid level in a tray which makes flooding worse. An operator usually recognizes flooding through an increase of deferential pressure of the column. Thereafter, inevitable flaring of the off-specification products results in economic loss. Meanwhile, in order to recover the normal state, the manual corrective action is reconfiguration of controllers (Fig. 10). In other words, an operator moderates the fault by changing set points and retuning control loops which is assumed as a supervision control.

Another cause of potential flooding is an increase of the relative vapor velocity (Fig. 10). Excess reboiler heat duty causes too much vapor to be boiled up, which increases the vapor flow up the column. This abnormal hydraulic traffic leads to an increase of the differential pressure, which is monitored by an operator. Although, a temporary perturbation via this pseudo-fault happens, flaring and corrective actions by an operator are not required in most of the cases. Since excess vapor in the column increase temperature of trays, a temperature controller automatically decreased the reboiler duty (Fig. 10). Thus, the state of column is automatically recovered to normal state in a short time. Generally, operator interaction with process in this case is sort of mislabeling the FD dataset.

4.2.2. Relabeling strategy

The dataset required for developing the FD system is provided by a simulation of the de-Butanizer in the Dynamic Aspen-Plus environment. This complex case study includes 5 closed control loops (Fig. 9). In order to resemble data streaming, the occurrence time of the fault is unknown. In other words, either the fault or the pseudo-fault is randomly triggered on unfixed time in each simulation run. Records of 30 process variables are collected in the initial dataset:

$$D_0 = \left\{ (Y_{1:3000}^{r0}, F_{1:3000}^{r0}) \mid r = 1:100 \right\} \tag{13}$$

The initial trained GNB classifier based on D_0 is evaluated in terms of $F1$ and accuracy. In this order, the true labels are determined by the Kister-Hass factor (Kister, 1992) based on the hydraulic state of the de-Butanizer column. Then, the iterative procedure of FD is implemented. The results (Table 4) reveal that the final FD system produces higher accuracy than the sole classifier in the initialization. The rates of fault impacts, i.e. flaring and runaway flooding, are also decreased. Therefore, the proposed strategy promises a successive refinement of the datasets, and consequently an efficient diagnosis system. Regardless of the classifier types and the state space model, this positive trend is expected, because the strategy is focused on the refinement of attributes and labels.

Table 4: Diagnosis performance of flooding using iterative procedure

	Learning phase	Initial	1st	2nd	3rd	4th	5th	6th
	$I_{interaction}$ %	0	3	9	13	16	18	20
$F1$	Normal	0.83	0.85	0.87	0.89	0.91	0.92	0.93
	Fault	0.65	0.72	0.78	0.82	0.86	0.88	0.90
Accuracy		0.70	0.76	0.81	0.85	0.88	0.90	0.92
TPR%		99.62	99.48	99.43	99.46	99.48	99.49	99.60
FPR % ∝ Unnecessary flaring		31.03	20.69	18.17	15.98	13.51	11.02	9.58
FNR % ∝ Runaway flooding		0.38	0.52	0.47	0.44	0.42	0.41	0.40

It is worth to consider that the control loops in the process make it robust with regards to small perturbations due to a pseudo-fault. This robustness is a kind of passive fault-tolerance (Patton, 2015). In this way, the interaction of an operator with the process is less demanded, which leads to lower potential human error. Consequently, the change of $I_{interaction}$ in sequential learning stages is smoother for closed loop systems rather than for

open loop ones. However, for a severe process fault exceeding this tolerance, the control system cannot compensate the fault. Thus, an active fault-tolerant system requires fault-detection methods and a reconfiguration mechanism.

Moreover, the correction factor of $I_{interaction}$ (Eq. 8) is a controller of the speed of learning. The less the plant can tolerate unwanted situations, the more the ARL and $I_{interaction}$ is reduced. In addition, an inaccurate classifier adversely impacts on the confidence of the FD system. Consequently, $I_{interaction}$ adjusts the extent of changes and the chance of wrong decisions. In general, learning from the experience improves operator understanding, and enhances the FD system. Certainly, there is a trade-off between the cost of learning (the cost of the risk, depending on the situation and the process) and the reward from learning (the better FD and flaring reduction, in this particular case). Optimization of this problem is beyond the scope of this work and deserves further research.

4.2.3. Filtering strategy

There are various strategies dealing with mislabeling (Frénay & Verleysen, 2014). The aim of this subsection is the comparison of the proposed approach with a filtering approach. In order to improve the quality of the training data, the samples with uncertain labels can be removed following the procedure described in Section 3.2 except Stages 3-4. Alternatively, any sample inconsistently labelled by the operator and the classifier is omitted from the database. The idea is to refine the training data via selecting reliable samples; and to develop a more efficient classifier.

The filtering strategy is iteratively implemented on the de-Butanizer dataset. Figure 11 shows its performance in terms of accuracy, and it is compared with the proposed strategy. Despite the increasing trend of accuracy in sequential learning stages of the filtering strategy, but it is not as efficient as relabeling strategy. The main reason is that the relabeling strategy tries to consider the pseudo-fault as the normal state, despite the operator idea at the initial phase. Indeed, it is important to include the pseudo-fault subset into normal subsets, which leads to the improvement of the discrimination boundary. This strategy shows to have higher performance than discarding the pseudo-fault subset. As such, the FD system upgrades to distinguish pseudo-faults from risky faults.

4.3. Industrial Challenge

The real nature of the problem is next illustrated in an industrial challenge, for which the methods and tools proposed are applied in a limited way. The problem is first described and next it is analysed in regard of the available information and the data processing produced for the first learning stage.

Implementing the iterative strategy proposed in an industrial plant, and accordingly obtaining new data from it, is beyond the scope of this study since the necessary feedback cannot be afforded in the short term. Therefore, the results obtained reveal the opportunities to improve the operator decision-making, and indicate the course for further action.

The removal of sour gas components from gas streams, using chemical solvents such as amines, is a requirement in most gas processing plants. The acid gas constituents (H_2S and CO_2) react with an aqueous solvent in a high-pressure absorber which is a column equipped with trays (Fig. 12). Subsequently, the solvent is directed to a regenerator and stripped from the acid gas in at elevated temperature to reuse it.

One of the most frequent problems in a gas-sweetening unit is amine foaming in the absorber, which results in loss of proper vapor-liquid contact, solution hold up and poor solution distribution. The adverse consequences include off-specification product, excessive amine loss, reduced gas-treating capacity, and energy loss. For resolving the problem, antifoam is injected into the amine recirculation system at several sensitive points. Antifoams are chemicals formulated to allow expansion of liquid film of amine bubbles and formation of thin spots, which subsequently lead to rupture of bubbles (Sheilan, Spooner, Street, & van Hoorn, 2005).

In practice, the operator monitors online data to estimate the state of the system. The following symptoms are usually helpful to the operator for foaming detection:

- Fluctuating pressure drop in the absorber
- Increase of flash gas
- Amine carryover from absorber or flash tank
- Swinging liquid levels in any vessel
- Decrease in H_2S removal with increase in CO_2 removal
- Off-specification treated gas

Furthermore, it is possible to get samples of the amine solution to check foaming potential using the shake test. However, this offline test is not common due to delay in getting results and hazard of sour gas in the amine solution. On the other hand, developing an accurate first principle model of this industrial process is impracticable. As such, achieving true labels regarding the dataset is almost challenging. Therefore, online monitoring and state estimation typically is preferred to redundant shake test, which is herein required for validation.

In case of severe foaming, the operator can inject fast and massive antifoam by a centrifugal pump. Indeed, an online foaming sensor and a close control loop are not available in the plant. On the other hand, the operator may be misled by temporary pressure fluctuation of the absorber column due to vortex of liquid in the bottom of the column (a pseudo- fault). Then, unnecessary antifoam injection by an operator can lead to long-term process drawbacks, such as the reduction of amine filter efficiency or the formation of heat stable salts. Wrong decision-making leads to extra costs, direct and indirect, including antifoam consumption, more frequent filter maintenance, and corrosion due to salts. Therefore, it is required that the operator makes right decisions regarding the convenience of antifoam injection (Kister, 2006; Sheilan, et al., 2005).

An efficient FD system can aid the operator to deal with this dilemma. First, records of 26 in-situ sensors of pressure, level, flow and temperature in significant parts of the gas-sweetening unit of a real plant are provided. An initial dataset including observations and operator labels is as follows:

$$D_0 = \left\{ (Y_{1:4000}^{r0}, F_{1:4000}^{r0}) \mid r = 1:24 \right\} \tag{14}$$

The GNB classifier is fitted based on the training subset of D_0. In order to evaluate performance, the 3-folded cross-validation is implemented. The cross validation is important to check the classification robustness against the particular choice for the training and testing datasets. The predicted labels of the testing subset are compared with the true labels, which have been achieved according to the shake tests (Table 5). Thereafter, the conflict between operator labels and FD predictions is investigated. $I_{interaction}$ reveals the existence of unnecessary actions and indicates a promising opportunity for improving FD in further refining phases. In addition, the low standard

deviation values (σ) of the 3-folded validation (Table 5) show the stability of the classifier in prediction.

Table 5: The initial learning phase of the foaming detection system

	$I_{interaction}$	Accuracy
1st fold	0.8382	0.6246
2nd fold	0.8322	0.6218
3rd fold	0.8384	0.6291
3-folded Validation [μ,σ]	[0.8332,0.0031]	[0.6251,0.0052]

In order to improve the performance of the initial FD system, its implementation in the industrial plant is required, so that new datasets can be generated. However, this is highly resource consuming and getting more information from industrial practice, this is beyond the scope of the work presented. On the other hand, this case study highlights the industrial challenges of operator-process interaction on FD, and the significance of this problem, which clearly needs more attention and further investigation.

5. CONCLUSIONS

This work addresses efficient data-driven FD for complex chemical processes in the framework of operational intelligence, for which the effect of the inevitable operator-process interaction on process history records needs to be taken into account.

On the one hand, prior operator knowledge is required, as well as operator preventive actions, which are in turn considered true indications of faulty situations. However, some historical records in the database may be mislabeled due to human error. This has an adverse impact on the performance of an automatic FD system based on such data and such labels. On the other hand, the operator actions, necessary or not, alter the system dynamics and may interrupt the recording of process data (dynamic attributes) corresponding to a true fault.

Results of the illustrative CSTR case study showed that the GNB classifier trained based on the initial dataset just provides a model of operator preferences. Furthermore, it also showed that this policy incurred in unnecessary utility consumption. Thus, there was a motivation to provide refined datasets by managing label uncertainty and dynamic attributes of the system.

In order to resolve mentioned FD challenge, an iterative framework was proposed. Relabeling was done based on the estimation of the underlying state and a discrimination criteria defined as median of Mahalanobis distance. In addition, operator behavior toward the system was also shown to improve by considering the interaction index and an acceptable risk level. Indeed, this procedure provided the chance of generating new attributes at each training phase.

The FD system was developed and tested for a CSTR and flooding of a simulated de-Butanizer column. Iterative retraining of the GNB classifier through successive refined datasets led to higher detection performance in terms of accuracy and $F1$. The proposed FD system aided the operator to make better decisions in regard of the extra cost connected to unnecessary actions. Furthermore, the results revealed that the proposed FD strategy is more efficient than the filtering strategy. Nevertheless, collecting new information on the system response in the absence of operator actions was shown to be at expense of increased accident rate in the early phases of learning, as should be expected in regard of natural learning. Thereafter, this information was shown to be useful for more efficient discrimination in further learning phases, which leaded to decreasing rate of accidents (or unwanted situations). The cost of learning, and the trade-off with the associated risks and losses, suggests a challenging optimization problem and further research lines including incremental learning can be envisaged from this work. Thus, more parameters need to be involved; e.g. fault intensity and its consequences.

Furthermore, a foaming case study in a gas refinery introduces an industrial challenge which reveals that FD under operator-process interaction is an important open research problem. However, the cost of obtaining data from true faulty situations in industrial practice is difficult.

Finally, the main issue in this work was fault detection, which is a binary classification (normal - abnormal). A limitation of the proposed scheme is the incapacity of dealing with multiple faults, which needs more general discrimination boundaries. In fact, the bottleneck is the criteria based on Mahalanobis distance which is binary. In addition, for fault prognosis, it is required to include additional modules into the proposed framework in a further work.

NOMENCLATURE

a	number of true positive samples
a'	number of faulty samples by the operator and the classifier
b	number of false positive samples
b'	number of samples diagnosed as faulty in contrast with operator opinion
c	number of false negative samples
c'	number of samples labeled faulty by operator but not diagnosed
C_A	concentration of A in the outlet stream
C_{Af}	concentration of A in the feed stream
C_B	concentration of B in the outlet stream
d	number of true negative samples
d'	number of samples which both the operator and the classifier have agreement on no fault happened.
D_M	Mahalanobis distance
D_p	dataset at each phase
F	fault
F_t	identification matrices of observation equation
G_t	identification matrices of state equation
$I_{interaction}$	interaction index
m	number of attributes
p^{th}	counter of phase
R	number of runs
r^{th}	counter of runs
T	temperature of the reactor
t	time
T	time horizon
T_f	temperature of the inlet stream
T_j	temperature of the jacket
v_t	Gaussian random vectors of observation equation
V_t	variance matrices of observation equation
w_t	Gaussian random vectors of state equation
W_t	variance matrices of state equation
Y_t	attributes

Greek symbol

θ_t	state variables
Σ	covariance matrix
μ	mean
σ	standard deviation
π	probability

Acronyms

ASM	abnormal situation management
ARL	acceptable risk level
CSTR	continuous stirred tank reactor
DLM	dynamic linear model
FD	fault detection
FDA	Fisher discriminant analysis
FPR	false positive rate
FNR	false negative rate
GNB	Gaussian naïve Bayes
HMM	hidden Markov model
KPI	key performance indicator
MLE	maximum likelihood estimation
PCA	principal component analysis
PID	proportional, integral and derivative
PLS	partial least squares
SVM	support vector machines
TPR	true positive rate

ACKNOWLEDGEMENTS

REFERENCE:

Askarian, M., Escudero, G., Graells, M., Zarghami, R., Jalali-Farahani, F., & Mostoufi, N. (2015). Fault Diagnosis of Chemical Processes with Incomplete Observations: A Comparative Study. *Computers & Chemical Engineering, 84*, 104-116.

Bao, H., Khan, F., Iqbal, T., & Chang, Y. (2011). Risk-based fault diagnosis and safety management for process systems. *Process Safety Progress, 30*, 6-17.

Bootkrajang, J., & Kabán, A. (2012). Label-noise robust logistic regression and its applications. In P. A. Flach, T. D. Bie & N. Cristianini (Eds.), *Machine learning and knowledge discovery in databases* (pp. 143-158). Berlin Heidelberg: Springer.

Brodley, C. E., & Friedl, M. A. (1999). Identifying mislabeled training data. *Journal of Artificial Intelligence Research, 11*, 131-167.

Chiang, L. H., Braatz, R. D., & Russell, E. L. (2001). *Fault detection and diagnosis in industrial system*. London: Springer.

Danenas, P., & Garsva, G. (2015). Selection of support vector machines based classifiers for credit risk domain. *Expert Systems with Applications, 42*, 3194-3204.

de França, F. O., & Coelho, A. L. (2015). A biclustering approach for classification with mislabeled data. *Expert Systems with Applications, 42*, 5065-5075.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern classification* (2nd ed.). New York: John Wiley & Sons.

Frénay, B., de Lannoy, G., & Verleysen, M. (2011). Label noise-tolerant hidden Markov models for segmentation: application to ECGs. In *Machine learning and knowledge discovery in databases* (pp. 455-470). Belgium: Springer.

Frénay, B., & Verleysen, M. (2014). Classification in the presence of label noise: a survey. *IEEE Transactions on Neural Networks and Learning Systems, 25*, 845-869.

IEC61511. (2003). Functional Safety: Safety instrumented systems for the process industry sector. Geneva, Switzerland: International Electrotechnical Commission.

Isermann, R. (1994). On the applicability of model-based fault detection for technical processes. *control engineering practice, 2*, 439-450.

Isermann, R. (2006). *Fault-diagnosis systems: an introduction from fault detection to fault tolerance*: Springer Science & Business Media.

Isermann, R., & Balle, P. (1997). Trends in the application of model-based fault detection and diagnosis of technical processes. *control engineering practice, 5*, 709-719.

Kister, H. Z. (1992). *Distillation design* : McGraw-Hill New York.

Kister, H. Z. (2006). *Distillation troubleshooting*: John Wiley & Sons

Korbicz, J., Koscielny, J. M., Kowalczuk, Z., & Cholewa, W. (2012). *Fault diagnosis: models, artificial intelligence, applications*: Springer Science & Business Media.

Lendek, Z., Babuška, R., & De Schutter, B. (2008). Distributed Kalman filtering for cascaded systems. *Engineering Applications of Artificial Intelligence, 21*, 457-469.

Li, Z., Fang, H., & Huang, M. (2015). Diversified learning for continuous hidden Markov models with application to fault diagnosis. *Expert Systems with Applications, 42*, 9165-9173.

Lu, N., Yao, Y., Gao, F., & Wang, F. (2005). Two-dimensional dynamic PCA for batch process monitoring. *AIChE Journal, 51*, 3300-3304.

Ludwig, E. E. (1997). *Applied process design for chemical and petrochemical plants* (Vol. 2): Gulf Professional Publishing.

MacGregor, J., & Cinar, A. (2012). Monitoring, fault diagnosis, fault-tolerant control and optimization: Data driven methods. *Computers & Chemical Engineering, 47*, 111-120.

Mehranbod, N., Soroush, M., Piovoso, M., & Ogunnaike, B. A. (2003). Probabilistic model for sensor fault detection and identification. *AIChE Journal, 49*, 1787-1802.

Monroy, I., Villez, K., Graells, M., & Venkatasubramanian, V. (2012). Fault diagnosis of a benchmark fermentation process: a comparative study of feature extraction and classification techniques. *Bioprocess and Biosystems Engineering, 35*, 689-704.

Namdari, M., & JazayeriRad, H. (2014). Incipient fault diagnosis using support vector machines based on monitoring continuous decision functions. *Engineering Applications of Artificial Intelligence, 28*, 22-35.

Pandya, D., Upadhyay, S., & Harsha, S. (2013). Fault diagnosis of rolling element bearing with intrinsic mode function of acoustic emission data using APF-KNN. *Expert Systems with Applications, 40*, 4137-4145.

Patton, R. J. (2015). Fault-tolerant control. *Encyclopedia of Systems and Control*, 422-428.

Petris, G., Petrone, S., & Campagnoli, P. (2009). *Dynamic linear models with R*. New York: Springer.

Qin, S. J. (2012). Survey on data-driven industrial process monitoring and diagnosis. *Annual reviews in control, 36*, 220-234.

Rajaram, K., & Jaikumar, R. (2000). Incorporating operator–process interactions in process control: A framework and an application to glucose refining. *International Journal of Production Economics, 63*, 19-31.

Sáez-Atienzar, S., Martínez-Gómez, J., Alonso-Barba, J. I., Puerta, J. M., Galindo, M. F., Jordán, J., & de la Ossa, L. (2015). Automatic quantification of the subcellular localization of chimeric GFP protein supported by a two-level Naive Bayes classifier. *Expert Systems with Applications, 42*, 1531-1537.

Serdio, F., Lughofer, E., Pichler, K., Buchegger, T., & Efendic, H. (2014). Residual-based fault detection using soft computing techniques for condition monitoring at rolling mills. *Information Sciences, 259*, 304-320.

Serdio, F., Lughofer, E., Pichler, K., Buchegger, T., Pichler, M., & Efendic, H. (2014). Fault detection in multi-sensor networks based on multivariate time-series models and orthogonal transformations. *Information Fusion, 20*, 272-291.

Sheilan, M. H., Spooner, B. H., Street, D. E., & van Hoorn, E. (2005). *Amine treating and sour water stripping*. Netherlands: Amine Experts.

Sheridan, T. B., & Johannsen, G. (1976). *Monitoring behavior and supervisory control*: Perseus Publishing.

Si, X. S., Hu, C. H., Yang, J. B., & Zhang, Q. (2011). On the dynamic evidential reasoning algorithm for fault prediction. *Expert Systems with Applications, 38*, 5061-5080.

Swartz, T. B., Haitovsky, Y., Vexler, A., & Yang, T. Y. (2004). Bayesian identifiability and misclassification in multinomial data. *Canadian Journal of Statistics, 32*, 285-302.

Teng, C. M. (2005). Dealing with data corruption in remote sensing. In A. F. Famili, J. N. Kok, J. M. Pena, A. Siebes & A. Feelders (Eds.), *Advances in Intelligent Data Analysis VI* (pp. 452-463). Berlin Heidelberg: Springer.

Venkatasubramanian, V., Rengaswamy, R., Yin, K., & Kavuri, S. N. (2003). A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. *Computers & Chemical Engineering, 27*, 293-311.

Yu, J. (2013). A new fault diagnosis method of multimode processes using Bayesian inference based Gaussian mixture contribution decomposition. *Engineering Applications of Artificial Intelligence, 26*, 456–466.

Zhang, C., Li, D., Yang, J., & Yong, A. (2014). Human cognitive paradigm and its application in semi-supervised learning. *Optik-International Journal for Light and Electron Optics, 125*, 1178-1184.

APPENDIX

A.1. Energy and mass balances equation of the CSTR

$$r(t) = k_f C_A(t) - k_r C_B(t) \tag{11}$$

$$k_f = k_{f0}.e^{\frac{-E_f}{R.T(t)}} \tag{12}$$

$$k_r = k_{r0}.e^{\frac{-E_r}{R.T(t)}} \tag{13}$$

$$\frac{dC_A(t)}{dt} = \frac{F}{V}.(C_{AF}(t) - C_A(t)) - r(t) \tag{14}$$

$$C_B(t) = C_{Af}(t) - C_A(t) \tag{15}$$

$$\frac{dT(t)}{dt} = \frac{F}{V}.(T_F(t) - T(t)) - \frac{H}{C_p.\rho}.r(t) - \frac{U.A}{C_p.\rho}.(T(t) - T_j(t)) \tag{16}$$

Table 6: CSTR model parameters

Volumetric flowrate (m³/s)	$F=1$	Forward activation energy (J/mol)	$E_f=49614$
Volume of CSTR (m³)	$V = 150$	Reverse activation energy (J/mol)	$E_r=50242$
Density of A-B mixture (kg/m³)	$\rho =833$	Forward pre-exponential factor (1/s)	$k_{f0}=$ $3.4\cdot10^7$
Specific heat of A-B mixture (J/kg·K)	$C_p = 0.12$	Reverse pre-exponential factor (1/s)	$k_{r0} =2\cdot10^7$
Heat of reaction (J/mol)	$H=-5960$	Overall heat transfer coefficient (W/m²·K)	$U=7$
Universal gas constant (J/mol·K)	$R=8.31451$	Area (m²)	$A=100$

A.2. Recursive Kalman filter of DLM

Consider the DLM specified by:

$$Y_t = F_t\boldsymbol{\theta}_t + v_t \qquad\qquad v_t \sim N_m(0,V_t) \tag{17}$$

$$\boldsymbol{\theta}_t = G_t\boldsymbol{\theta}_{t-1} + w_t \qquad\qquad w_t \sim N_n(0,W_t) \tag{18}$$

Let consider:

$$\boldsymbol{\theta}_{t-1} \mid Y_{1:t-1} \sim N(m_{t-1}, C_{t-1}) \tag{19}$$

Then the following statements hold.

(i) The one-step-ahead predictive distribution of $\boldsymbol{\theta}_t$ given $Y_{1:t-1}$ is Gaussian, with parameters

$$a_t = E(\boldsymbol{\theta}_t \mid Y_{1:t-1}) = G_t m_{t-1} \tag{20}$$

$$R_t = Var(\boldsymbol{\theta}_t \mid Y_{1:t-1}) = G_t C_{t-1} G_t' + W_t \tag{21}$$

(ii) The one-step-ahead predictive distribution of Y_t given $Y_{1:t-1}$ is Gaussian, with parameters

$$f_t = E(Y_t \mid Y_{1:t-1}) = F_t a_t \tag{22}$$

$$Q_t = Var(Y_t \mid Y_{1:t-1}) = F_t R_t F_t' + V_t \tag{23}$$

(iii) The filtering distribution of $\boldsymbol{\theta}_t$ given $Y_{1:t}$ is Gaussian, with parameters

$$m_t = E(\boldsymbol{\theta}_t \mid Y_{1:t-1}) = a_t + R_t F_t' Q_t^{-1} e_t \tag{24}$$

$$C_t = Var(\boldsymbol{\theta}_t \mid Y_{1:t}) = R_t - R_t F_t' Q_t^{-1} F_t R_t \tag{25}$$

where $e_t = Y_t - f_t$ is the forecast error.

B. Supplementary data

Datasets associated with this article including CSTR and gas-sweetening unit measurements can be found, in the online version.

LIST OF TABLES:

LIST OF FIGURES: