

FAULT DETECTION AND DIAGNOSIS BY SUPPORT VECTOR MACHINES: APPLICATION TO VINYL-CHLORIDE-MONOMER PROCESS

Siriwatida Srirabai, and Chanin Panjapornpon*

Received: December 25, 2021; Revised: August 08, 2022; Accepted: November 07, 2022

Abstract

Monitoring process status and identifying process operational faults are essential for improving the process safety in petrochemical plants that interactions between various process streams and units are associated. This paper presents a deployment of a support vector machine technique for detecting and identifying operational fault cases with a case study of a vinyl chloride monomer plant. An integrated simulation environment between MATLAB and UniSim Design dynamic simulator is utilized for evaluating the performance of the proposed fault detection and identification framework. Under the real-time software-in-the-loop simulation, the confusion matrix results and receiver operating characteristics supported that the proposed framework provides high accuracy of fault classification.

Keywords: Fault detection and diagnosis; machine learning application; process safety; support vector machine; vinyl chloride monomer

Introduction

Automated monitoring and fault diagnosis is essential for analyzing product quality, tracking process performance, and ensuring process safety. However, due to various circumstances, identifying abnormal or fault conditions are difficult in practice (e.g., the complexity of interconnected units and flow streams, sensor or transmitter malfunction, and human errors). As a consequence, fault detection and diagnosis (FDD) could assist in identifying the root cause of a problem and reduce the misunderstanding that may lead to potential risks. Various industries, particularly petrochemical

processes, have expressed interest in the model-based FDD. Based on the literature, several model-based fault identification and diagnostic techniques have been developed. Quantitative-based, qualitative-based, and process history-based techniques are the three methodologies that can be used (Venkatasubramanian *et al.*, 2003a; 2003b; 2003c; Mouzakitis, 2013; Park *et al.*, 2020). The quantitative-based method uses transformations in observed process data to calculate the residual or feature space used to determine fault characteristics. The observer, parity space, and extended Kalman filter

Department of Chemical Engineering, Center of Excellence on Petrochemicals and Materials Technology, Faculty of Engineering, Kasetsart University, Bangkok 10900, Thailand. Tel. 0-2797-0999 Ext. 1230, Fax: 0-2561-4621, E-mail: siriwatida.s@ku.ac.th; fengcnp@ku.ac.th

* Corresponding author

(EKF) are the most popular methods in this approach. The model quality has a significant impact on diagnostic accuracy. When used with complex processes, the detection accuracy may be less effective (Sotomayor and Odloak, 2005; Mouzakitis, 2013). The qualitative approach performs fault diagnostics by representing input-output data as qualitative functions, such as casual or hierarchical models. However, it seems challenging to apply to a large-scale operation because it requires establishing relationships between variables (James *et al.*, 2018). In contrast to qualitative and quantitative approaches, the process history database approach builds the detection model based on historical input-output data without prior knowledge or process Information. It is extensively used in complicated processes for real-time process monitoring and multi-class categorization because of this advantage. The support vector machine (SVM), principal component analysis, and neural network are examples of methods in this area (Zhou *et al.*, 2003; Cai *et al.*, 2015).

SVM applications for process monitoring and fault classification have recently seen some advances (Gao and Hou, 2016; Taqvi *et al.*, 2018; Han *et al.*, 2019; Onel *et al.*, 2019). The Tennessee Eastman process was applied to feature selection using nonlinear SVM and two-class C-SVM classification (Onel *et al.*, 2019) and multi-class classification using kernel SVM with data normalization and PCA dimension reduction (Gao and Hou, 2016). Han and colleagues (Han *et al.*, 2019) developed the least squares support vector machine (LS-SVM) with hyperparameter tuning for fault detection in centrifugal coolers. Taqvi and colleagues (Taqvi *et al.*, 2018) used Aspen plus dynamics and the MATLAB co-simulation environment to demonstrate a multi-kernel SVM for fault identification and diagnostics in the distillation column.

To improve process monitoring and fault identification, this paper presents the SVM fault detection and identification of multi-class operational faults for a dynamic process. The domain knowledge-based feature selection is used in the proposed SVM-FDD framework, while the learning process uses k-fold cross-validation and hyperparameter optimization. The multiclassification investigates the concepts of one-versus-all and one-versus-one models. A case study of a vinyl chloride monomer process is developed in an integrated simulation framework between the UniSim Design process simulator and MATLAB software to test the real-time implementation of the proposed SVM-FDD method.

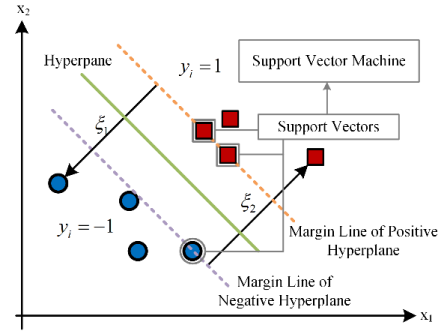


Figure 1. Support vector machine with two classifications

Support Vector Machine

Support vector machine is a supervised learning method that analyses data for classification. Consider the training data $\{x_1, y_1\}, \dots, \{x_n, y_n\}$ where $x_i \in R^m$ is the m-dimensional feature vector of the training data i and $y_i \in \{-1, 1\}$ is the class label of x_i . Binary classification is applied to separate two classes: a positive class $y_i = 1$ and a negative class $y_i = -1$ with the centerline called an optimal hyperplane (Cortes and Vapnik, 1995). Figure 1 shows the illustration of the linear support vector machine. The hyperplane is described by Equation (1)

$$wx + b = 0 \quad (1)$$

where $x \in R^m$ is the m-dimensional input vector of the training data $\{x_1, y_1\}, \dots, \{x_n, y_n\}$, w is the normal vector, and b is the bias.

The optimal hyperplane is created with the largest margin ξ_i between classes. The margin parallels the hyperplane and passes through the nearest positive and negative points on the support vectors in Figure 1.

$$y_i(wx + b) \geq 1 - \xi_i, \quad i = 1, \dots, n \quad (2)$$

For nonseparable data, the optimal hyperplane can be obtained by solving the optimization problem in Equation (3)

$$\begin{aligned} \min_{w, b, \xi} \quad & \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i \\ \text{subject to} \quad & y_i(wx_i + b) \geq 1 - \xi_i \\ & \xi_i \geq 0, \quad i = 1, \dots, n \end{aligned} \quad (3)$$

where C is an error penalty and ξ_i is the largest margin of x_i .

To simplify the constrained optimization problem, the Karush-Kuhn-Tucker (KKT) method is applied to formulate the dual problem shown in Equation (4)

$$L(w, b, \alpha, \gamma) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i - \sum_{i=1}^n \alpha_i [y_i (w x_i + b) - 1 + \xi_i] - \sum_{i=1}^n \gamma_i \xi_i \quad (4)$$

where α_i, γ_i is the Lagrange multiplier. Equation (5) is solved for obtaining the parameters of the KKT condition at the optimum point.

$$\begin{aligned} \frac{\partial L}{\partial \xi} &= \alpha_i + \gamma_i = C \\ \frac{\partial L}{\partial b} &= \sum_{i=1}^n \alpha_i y_i = 0 \\ \frac{\partial L}{\partial w} &= w - \sum_{i=1}^n \gamma_i y_i x_i = 0 \\ \alpha_i, \gamma_i, \xi_i &\geq 0 \end{aligned} \quad (5)$$

Substituting Equation (5) into Equation (4) leads to a dual problem in Equation (6).

$$\begin{aligned} \max_{\alpha} \quad & \sum_j \alpha_j - \frac{1}{2} \sum_i \sum_j y_i y_j \alpha_i \alpha_j x_i' x_j \\ \text{subject to} \quad & \sum_i y_i \alpha_i = 0 \\ & 0 \leq \alpha_i \leq C, \quad i = 1, \dots, n \end{aligned} \quad (6)$$

α_i^* is the optimal value. Therefore, w^* and b^* could be calculated from Equation (7)

$$\begin{aligned} w^* &= \sum_{i=1}^n \alpha_i y_i x_i \\ b^* &= \frac{-\max_{i: y_i = -1} w^* \cdot x_i + \min_{i: y_i = 1} w^* \cdot x_i}{2} \end{aligned} \quad (7)$$

The decision hyperplane can be calculated by Equation (8)

$$f(x) = \text{sgn}(w^* x + b^*) = \text{sgn}\left(\sum_{i=1}^n \alpha_i^* y_i x_i x + b^*\right) \quad (8)$$

where $\text{sgn}(x) = \frac{x}{|x|}$, $x \neq 0$. The one-versus-all and one-versus-one concepts are deployed by organizing the SVM data as a binary class to support the multi-class classification. The one-versus-all approach creates a multi-class dataset into a binary

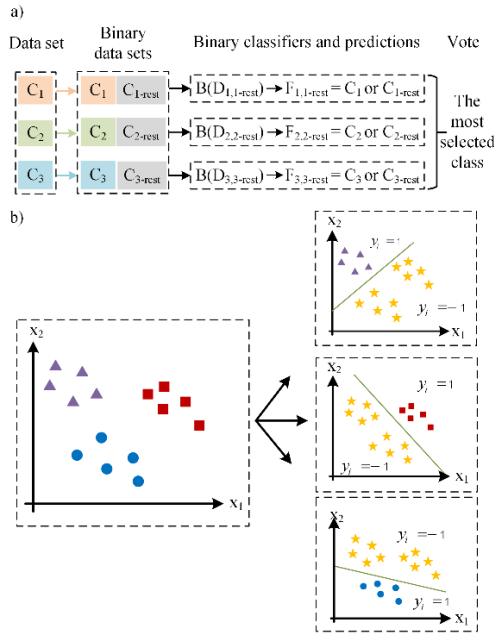


Figure 2. Illustration of a) data preparation and b) classification results for one-versus-all multi-class classifier

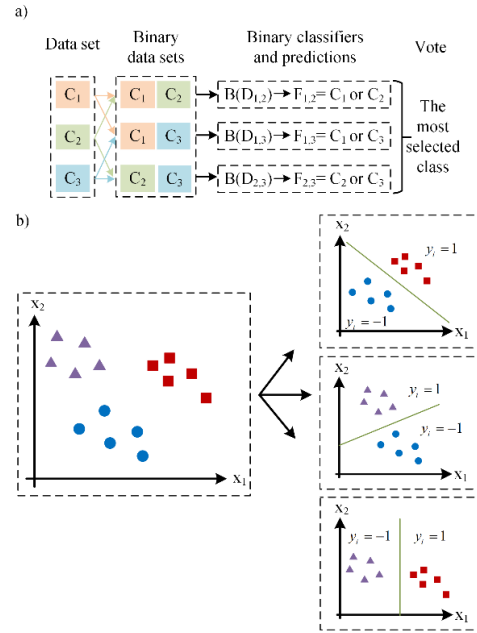


Figure 3. Illustration of a) data preparation and b) classification results for one-versus-one multi-class classifier

screened to eliminate insignificant input variables using domain knowledge from the process operation. Furthermore, less sensitive measured variables, such as control loop outputs, are deleted. As a result, the number of variables has been reduced to 26. Tables 1 and 2 contain descriptions of selected features as well as the ranges of operational faults and normal conditions.

As illustrated in Figure 5, the proposed modeling framework for constructing a multi-class SVM classifier for the VCM process consists of three primary steps: 1) data preparation, 2) model training and validation, and 3) model prediction. To produce the training datasets and assess the performance of developed algorithms in real-time, a co-simulation environment between MATLAB and UniSim Design, also known as software-in-the-loop simulation, is used. The software-in-the-loop technique is critical for simulating developed algorithms with a virtual process in real-time. Many engineering applications have used this method, including the construction of model-based control systems (Tanakunmas *et al.*, 2017; Chinprasit and Panjapornpon, 2020), process modeling (Liu *et al.*,

2019), and process fault detection (Taqvi *et al.*, 2018; Liu *et al.*, 2019).

The UniSim Design simulator generates VCM process data for 10 random datasets via the MATLAB actxserver. Each dataset provides dynamic process data that has been collected under the specified conditions (normal and 12 faults). After 30 sec, the stepwise faults are introduced into the VCM plant, which is simulated for 100 sec. As a result, every second, dynamic data is collected. The VCM datasets are separated into two parts: the training dataset (80%) and the validation dataset (the remaining 20%). After that, the dataset is normalized using Equation's Z-Score approach (12).

$$x'_{ij} = \frac{x_{ij} - \bar{x}_i}{\sigma_i} \quad (12)$$

where x'_{ij} is the normalized value, x_{ij} is the value of the i^{th} attribute of an object, $\bar{x}_i$ is the mean value of the i^{th} attribute, and σ_i is the standard deviation value of the i^{th} attribute.

Table 1. Selected features of the VCM process

Item	Description	Item	Description
1	EDC feed stream	14	The bottom flow of 4 th quench
2	A natural gas feed stream	15	Reflux flow of the HCl column
3	Coolant flow of 1 st quench	16	The liquid refrigerant flow of the HCl column
4	The top flow of 1 st quench	17	The top product flow of the HCl column
5	The bottom flow of 1 st quench	18	Reboiler circulate flow of the HCl column
6	Coolant flow of 2 nd quench	19	Medium pressure flow of the HCl column
7	The top flow of 2 nd quench	20	The bottom product flow of HCl column
8	The bottom flow of 2 nd quench	21	Reflux flow of the VCM column
9	Coolant flow of 3 rd quench	22	Coolant flow of the VCM column
10	The top flow of 3 rd quench	23	The top product flow of the VCM column
11	The bottom flow of 3 rd quench	24	Reboiler circulate flow of the VCM column
12	Coolant flow of 4 th quench	25	A low-pressure flow of column 2
13	The top flow of 4 th quench	26	Reboiler outflow of column 2

Table 2. Fault cases defined in the VCM process

Fault Case	Description	Normal (tonne/h)	Fault (tonne/h)
0	Normal		
1	Loss of EDC feed flow	50-57	<50
2	Flow low of natural gas cracking furnace	7.2-7.4	<7.2
3	Flow low of coolant in quencher 1	3.0-3.5	<3.0
4	Flow low of coolant in quencher 2	3.5-4.0	<3.5
5	Flow low of coolant in quencher 3	95-100	<95
6	Flow low of coolant in quencher 4	235-260	<235
7	Flow low of reflux column 1	10.9	<10.9
8	Flow low of coolant in column 1	32-35	<32
9	Flow low of steam in column 1	2.3-2.8	<2.3
10	Flow low of reflux column 2	3.4	<3.4
11	Flow low of coolant in column 2	92-100	<92
12	Flow low of steam in column 2	2.3-3.8	<2.3
13	Flow high of natural gas cracking furnace	7.2-7.4	>7.4
14	Flow high of steam in column 1	2.3-2.8	>2.8
15	Flow high of steam in column 2	2.3-3.8	>3.8

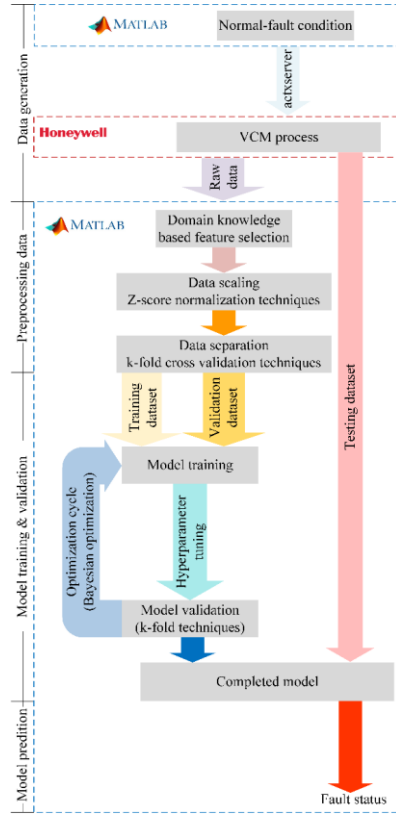


Figure 5. SVM modeling framework and software-in-the-loop simulation

In the model training step, the k-fold cross-validation approach with $k=5$ is applied to the datasets for training and validating the SVM model. Figure 6 depicts a schematic of the 5-fold cross-validation method. The SVM hyperparameters are optimized using Bayesian optimization with the acquisition function: expected improvement per second plus, Iteration: 10, optimized multi-class method: one-vs-one, one-vs-all, and C value, and Iteration: 10. Five random datasets of the VCM plant for each condition (normal and 15 faults) are evaluated using co-simulation between MATLAB and UniSim Design every second for 100 sec of simulation time to test the developed SVM classifier.

Results and Discussion

The performances of the SVM classifiers are evaluated based on the accuracy rate using confusion matrix analysis and the receiver operating characteristic (ROC) curve.

The confusion matrix is a table describing the performance of a classification model. The accuracy of the model is calculated by Equation (13).

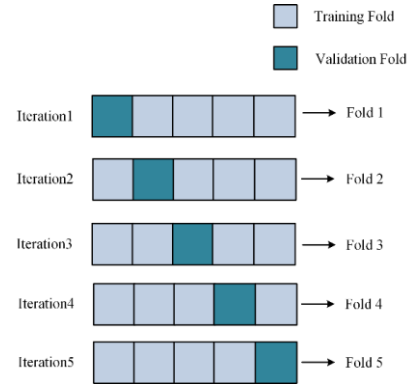


Figure 6. Schematic of 5-fold model cross-validation

		True class	
		positive	Negative
Predicted class	positive	True Positive TP	False Positive FP
	Negative	False Negative FN	True Negative TN

Figure 7. An illustrative example of the confusion matrix

$$\text{Accuracy Rate (Acc)} = \frac{TP+TN}{TP+FP+FN+TN} \quad (13)$$

where true positive (TP) is a positive data point predicted as positive; true negative (TN) is a negative data point predicted as negative; false positives (FP) is a positive data point predicted as negative; false negatives (FN) is a positive data point predicted as negative. The confusion matrix is illustrated in Figure 7.

The receiver operating characteristic (ROC) curve is also used to evaluate performance for the classification model. The ROC curve is a graph showing the plot of two variables, true positive rate (sensitivity) and false-positive rate (1-specificity) as in Equations (14) and (15).

$$\text{Sensitivity} = \frac{TP}{TP+FN} \quad (14)$$

$$\text{Specificity} = \frac{TN}{TN+FP} \quad (15)$$

A performance classification model of the ROC curve is evaluated by area under a graph (AUC). Figure 8 illustrates the AUC curve. When the AUC value is close to 1, the model provides the best classifier. In contrast, poor classification performance is obtained when the AUC value is near 0.

kernel-based approaches, and other learning methods. The SVM classifier outperforms the rest classifiers observed by the validation accuracy. The influence of optimization hyper-parameter C value on the multi-class classification of both one-vs-one and one-vs-all concepts is investigated in the training model. Table 4 summarizes the results of the optimized hyperparameters for the one-vs-one and one-vs-all SVM classifiers. Model 8, a one-vs-one SVM classifier with a C value of 642.7776, provides approximately 100% validation accuracy, as do models 7 and 9. However, Model 8 is better since it only misclassifies the F1. The F1 is a fault in the EDC feed to the furnace that has a significant impact on the change of other process flow variables. As a result, misclassification could happen at the start of the fault prediction. Table 5 shows the classification performance of the SVM model 8 for the training dataset as measured by the confusion matrix. The model accuracy rate is summarized as 100 percent. Fault 1 (F1, loss of EDC feed flow) was the only one that was misclassified by 0.1 percent. As a result, with only one misclassification, this SVM model can accurately categorize training and validation data.



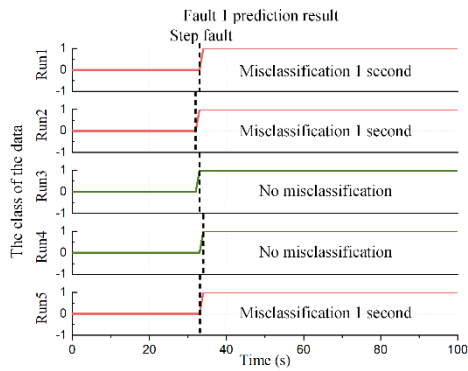
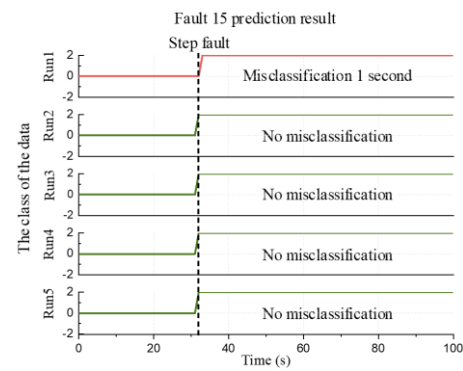
Model	Validation Accuracy
Decision Tree	52.2
Bagged Trees	73.4
Adaboost	69.2
Kernel	95.7
Linear Discriminant	97.3
Naïve Bayes	70.6
SVM	100.0

Model	C-Parameter	Multi-class	Classification error	Validation accuracy	Misclassification
1	0.2862	One vs all	0.0097	94.7	N, F1, F2, F3, F4, F5, F6, F9, F12, F13, F14
2	449.2889	One vs all	0.0023	92.8	N, F1, F8, F9, F10, F11, F12, F14, F15
3	1.0979	One vs one	0.0023	99.2	F1, F2, F3, F4, F5, F6, F12, F13, F14, F15
4	0.0015	One vs one	0.0023	40.5	All cases
5	0.4126	One vs one	0.0023	98.2	N, F1, F2, F3, F4, F5, F6, F12, F13, F14, F15
6	0.0022	One vs all	0.0023	80.6	F1, F2, F3, F4, F5, F6, F8, F9, F10, F12, F13, F14
7	34.5743	One vs one	0.0004	99.9	F1, F6, F12
8	189.5441	One vs one	0.0003	100.0	F1
9	156.0953	One vs one	0.0003	100.0	F1, F12
10	642.7776	One vs one	0.0003	100.0	F1, F12

[illegible]

Table 6. Confusion matrix of the model 8 under the co-simulation environment

		True class																FDR	
		Fault	N	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14		F15
Predicted Class	N	99.7	0.2															0.1	0.3
	F1		100																
	F2			100															
	F3				100														
	F4					100													
	F5						100												
	F6							100											
	F7								100										
	F8									100									
	F9										100								
	F10											100							
	F11												100						
	F12													100					
	F13														100				
	F14															100			
	F15																100		
	PPV	99.7	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	

**Figure 9. The prediction result of SVM classification of the fault 1****Figure 10. The prediction result of the SVM classification of the fault 15**

The model 8 is used in a co-simulation environment for real-time performance with 5 datasets containing 15 fault cases and normal under random conditions. The confusion matrix of the model 8 is illustrated in Table 6. The model accuracy rating is summarized as 99.7 percent. In case N (the normal case), it has a misclassification of 0.2 percent in fault 1 (the loss of EDC feed flow) and 0.1 percent in fault 15 (Flow high of steam in column 2). The error occurs in the first time step of the malfunction. Figure 9 shows that the fault 1 misclassification time is just 1 second of running 1, 2, and 5 and that the fault 15 misclassification time is only 1 sec of running 1 as shown in Figure 10. The SVM classification cannot be identified at the beginning of the process transition from normal to fault class.

The VCM process has a delay in data transmission to other units, making fault detection difficult and resulting in invalid detection in the first second. The performance of model 8 is assessed by the ROC curve. SVM classification has an AUC of

1 in all cases, indicating that the developed method provides high accuracy for fault classification. Figure 11 displays an example of the ROC curves for fault 1 and fault 15 classifications.

Conclusions

Under the co-simulation environment between MATLAB and UniSim Design, fault identification and diagnosis using support vector machines with multi-class classification are applied to the vinyl-choline monomer process. The SVM classification model is trained using domain knowledge-based feature selection, 5-fold cross-validation, hyperparameter optimization, and the one-vs-one and one-vs-all concepts. As a result, the SVM model successfully classifies the VCM process into 16 different conditions: normal and the other 15 fault types. Furthermore, according to the studies, the one-vs-one SVM model with optimized C hyperparameters equal to 189.5441 delivers a high accuracy rate of

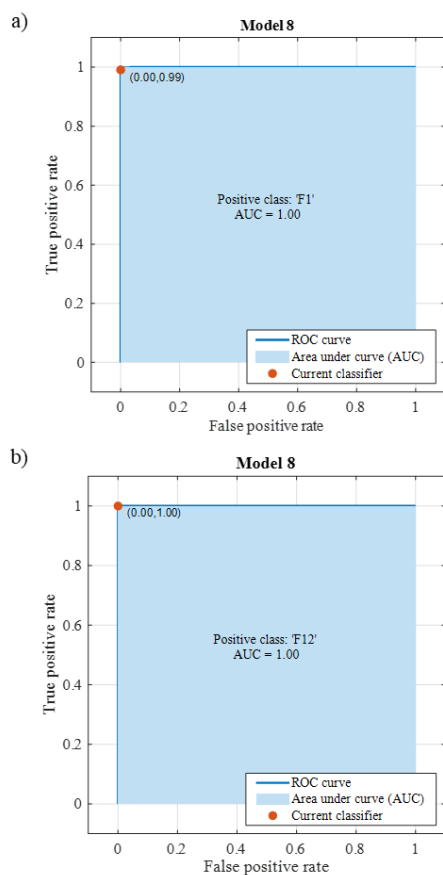


Figure 11. Receiver operating characteristic (ROC) for SVM classification

99.7 percent and a high AUC for fault identification and diagnosis. For multi-class classification, the SVM-FDD provides excellent fault detection and diagnosis accuracy to monitor chemical process operations.

Acknowledgment

This work has been supported by the Department of Chemical Engineering, Faculty of Engineering, Kasetsart University, and the Center of Excellence on Petrochemical and Materials Technology. In addition, the authors would like to acknowledge Honeywell and GRD Tech Co., Ltd. for the Academic License of UniSim Design Suite.

References

- Cai, E., Liu, D., Liang, L., and Xu, G. (2015). Monitoring of chemical industrial processes using integrated complex network theory with PCA. *Chemometrics and Intelligent Laboratory Systems*, 140:22-35. <https://doi.org/10.1016/j.chemolab.2014.10.008>.
- Chinprasit, J. and Panjapornpon, C. (2020). Model predictive control of vinyl chloride monomer process by Aspen Plus Dynamics and MATLAB/Simulink co-simulation approach. *IOP Conference Series: Materials Science and Engineering*, 778:012080. <https://doi.org/10.1088/1757-899X/778/1/012080>.
- Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273-297. <https://doi.org/10.1007/BF00994018>.
- Gao, X. and Hou, J. (2016). An improved SVM integrated GS-PCA fault diagnosis approach of Tennessee Eastman process. *Neurocomputing*, 174:906-911. <https://doi.org/10.1016/j.neucom.2015.10.018>.
- Han, H., Cui, X., Fan, Y., and Qing, H. (2019). Least squares support vector machine (LS-SVM)-based chiller fault diagnosis using fault indicative features. *Applied Thermal Engineering*, 154:540-547. <https://doi.org/10.1016/j.applthermaleng.2019.03.111>.
- James, A.T., Gandhi, O.P., and Deshmukh, S.G. (2018). Fault diagnosis of automobile systems using fault tree based on digraph modeling. *International Journal of System Assurance Engineering and Management*, 9(2):494-508. <https://doi.org/10.1007/s13198-017-0693-6>.
- Liu, B., Chen, H., Zhang, T., and Pei, P. (2019). A vehicular proton exchange membrane fuel cell system co-simulation modeling method based on the stack internal distribution parameters monitoring. *Energy Conversion and Management*, 197: 111898. <https://doi.org/10.1016/j.enconman.2019.111898>.
- Mouzakitis, A. (2013). Classification of fault diagnosis methods for control systems. *Meas. Control.*, 46(10), 303-308. <https://doi.org/10.1177/0020294013510471>.
- Onel, M., Kieslich, C.A., and Pistikopoulos, E.N. (2019). A nonlinear support vector machine-based feature selection approach for fault detection and diagnosis: Application to the Tennessee Eastman process. *AIChE Journal*, 65(3):992-1005. <https://doi.org/10.1002/aic.16497>.
- Park, Y.-J., Fan, S.-K.S., and Hsu, C.-Y. (2020). A review on fault detection and process diagnostics in industrial processes. *Processes*, 8(9):1,123. <https://doi.org/10.3390/pr8091123>.
- Sotomayor, O.A.Z. and Odloak, D. (2005). Observer-based fault diagnosis in chemical plants. *Chemical Engineering Journal*, 112(1-3):93-108. <https://doi.org/10.1016/j.cej.2005.07.001>.
- Tanakunmas, P., Panjapornpon, C., Tawai, A., and Dechakupt, T. (2017). Optimization-based control strategy with wavelet network input-output linearizing constraint for an ill-conditioned high-purity distillation column. *Industrial & Engineering Chemistry Research*, 56(31):8927-8939. <https://doi.org/10.1021/acs.iecr.7b01135>.
- Taqvi, S.A., Tufa, L.D., Zabiri, H., Maulud, A.S., and Uddin, F. (2018). Multiple fault diagnosis in distillation column using multikernel support vector machine. *Industrial & Engineering Chemistry Research*, 57(43):14689-14706. <https://doi.org/10.1021/acs.iecr.8b03360>.
- Venkatasubramanian, V., Rengaswamy, R., and Kavuri, S.N. (2003a). A review of process fault detection and diagnosis: Part II: Qualitative models and search strategies. *Computers & Chemical Engineering*, 27(3):313-326. [https://doi.org/10.1016/S0098-1354\(02\)00161-8](https://doi.org/10.1016/S0098-1354(02)00161-8).
- Venkatasubramanian, V., Rengaswamy, R., Kavuri, S.N., and Yin, K. (2003b). A review of process fault detection and diagnosis: Part III: Process history based methods. *Computers & Chemical Engineering*, 27(3):327-346. [https://doi.org/10.1016/S0098-1354\(02\)00162-X](https://doi.org/10.1016/S0098-1354(02)00162-X).
- Venkatasubramanian, V., Rengaswamy, R., Yin, K., and Kavuri, S.N. (2003c). A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. *Computers & Chemical Engineering*, 27(3):293-311. [https://doi.org/10.1016/S0098-1354\(02\)00160-6](https://doi.org/10.1016/S0098-1354(02)00160-6).

- Yin, Z. and Hou, J. (2016). Recent advances on SVM based fault diagnosis and process monitoring in complicated industrial processes. *Neurocomputing*, 174(part B):643-650. <https://doi.org/10.1016/j.neucom.2015.09.081>.
- Zhou, Y., Hahn, J., and Mannan, M.S. (2003). Fault detection and classification in chemical processes based on neural networks with feature extraction. *ISA Transactions*, 42(4):651-664. [https://doi.org/10.1016/S0019-0578\(07\)60013-5](https://doi.org/10.1016/S0019-0578(07)60013-5).