Intelligent Automation & Soft Computing
DOI:10.32604/iasc.2021.017232
Article

Tech Science Press

Identification of Abnormal Patterns in AR (1) Process Using CS-SVM

Hongshuo Zhang[1], Bo Zhu[1,*], Kaimin Pang[1], Chunmei Chen[1] and Yuwei Wan[2]

[1]College of Mechanical and Electrical Engineering, Kunming University of Science & Technology, Kunming, 650500, China
[2]School of Engineering, Cardiff University, Cardiff, CF24 3AA, UK
[*]Corresponding Author: Bo Zhu. Email: zhubo20110720@163.com

Received: 24 January 2021; Accepted: 01 March 2021

Abstract: Using machine learning method to recognize abnormal patterns covers the shortage of traditional control charts for autocorrelation processes, which violate the applicable conditions of the control chart, i.e., the independent identically distributed (IID) assumption. In this study, we propose a recognition model based on support vector machine (SVM) for the AR (1) type of autocorrelation process. For achieving a higher recognition performance, the cuckoo search algorithm (CS) is used to optimize the two hyper-parameters of SVM, namely the penalty parameter c and the radial basis kernel parameter g. By using Monte Carlo simulation methods, the data sets containing samples of eight patters are generated in experiments for verifying the performance of the proposed model. The results of comparison experiments show that the average recognition rate of the proposed model reaches 96.25% as the autocorrelation coefficient is set equal to 0.5. That is apparently higher than those of the SVM model optimized by the particle swarm optimization (PSO) or the genetic algorithm (GA). Another experiment result demonstrates that the average recognition accuracy of the CS-SVM model also reaches higher than 95% for different autocorrelation levels. At last, a lot of data streams in or out of control are simulated to measure the ARL values. The results turn out that the model has an acceptable online performance. Therefore, we believe that the model can be used as a more effective approach for identification of abnormal patterns in autocorrelation process.

Keywords: Control chart pattern; support vector machine; cuckoo search algorithm; autocorrelation process; Monte Carlo simulation

1 Introduction

Nowadays, the integration concept of informatization and industrialization is gradually being proposed. Many new technologies emerge continuously, such as Virtual Manufacturing (VM), Computer Integrated Manufacturing (CIM), and Radio-frequency Identification (RFID), Concurrent Engineering (CE), [1–5]. Under such a background, manufacturing companies have begun to shift from incremental competition to stock competition. Subsequently, product quality has become one of the core competitiveness of enterprises for success. In the quality engineering field, statistical process control (SPC) is an essential means for ensuring quality in the manufacturing process, which takes the control chart as the basic tool.

The use of traditional control charts must follow the premise that the observed data is independent identically distributed (IID). However, Shiau [6], Montgomery [7] pointed out that process data generally cannot approximately satisfy this premise in actual production process. When there are obvious autocorrelation phenomena presented in the observed data, the control chart often subjects to serious false alarm so that cannot detect true process abnormities effectively. Studies have shown that the autocorrelation phenomena significantly reduce the performance of control chart [8,9].

Academia has carried out much studies on process quality control with autocorrelation phenomena and proposed some approaches. The main category is to use statistical methods to eliminate the influence of autocorrelation on process data. The most representative one is the residual control chart. Montgomery and Woodall [10], Runger [11] proposed to use a fitted time series model to monitor the autocorrelation process. Because the controlled statistics' residuals are independent identically distributed theoretically, the conventional Shewhart control chart can be used to monitor the residuals effectively. Sun [12] and Zhang [13] carried out much research on the residual control chart of the autocorrelation process and validated its effectiveness. With the development of artificial intelligence technology, pattern recognition methods based on machine learning are proposed and introduced into the process quality control field. There are also some machine learning based methods applied to the autocorrelation process that have been reported. For example, Cook and Chiu [14] used the BP neural network to recognize the autocorrelation process' abnormal pattern. Experiments results show that it gets an advantage over the traditional statistical methods. Lin and Guh [15] applied a support vector machine (SVM) based model to abnormal pattern identification of autocorrelation process, which achieved better performance in both recognition accuracy and recognition speed. Zhu and Liu [16] proposed to use random forest for control chart pattern recognition in autocorrelation process, and verified that the performance of their model is better than that of the BP neural networks through simulation experiments.

SVM is a major machine learning method for classification, which has been also used frequently in control chart pattern recognition [17]. SVM first appeared in the 1990s. In contrast with the BP neural network's pursuit of minimizing experience risk, SVM takes minimum of the structural risk as optimizing goal. In that, SVM can better deal with the over-fitting problem, and then often achieve better generalization performance, especially under small sample conditions. Simultaneously, it has much fewer parameters need to be adjusted and accommodate to high-dimensional features. The use of SVM involves selecting an appropriate kernel function and tuning of the related hyper-parameters, which is essential for its generalization performance. Currently, there are some algorithms, such as artificial fish swarm algorithm (AFSA), genetic algorithm (GA), artificial bee colony (ABC), grid search (GS) and particle swarm optimization (PSO) [18–22], have been applied in the optimization of the hyper-parameters of SVM. The cuckoo search algorithm (CS) is a type of combination optimization algorithm appeared in recent years. Compared with other algorithms, it has many advantages, for instance, simpler structure, easy to jump out of local optimal values, and fewer control parameters.

Due to the superiorities of the SVM and the CS, we proposed an autocorrelation process pattern recognition model based on SVM optimized by CS (CS-SVM), which is used as the pattern classifier. In this model, the hyper-parameters of the SVM are optimized by the CS to get higher generalization performance. To verify this model's effectiveness, the data sets of basic patterns in AR (1) type of autocorrelation process, including normal pattern and seven types of abnormal pattern, are generated by Monte Carlo simulation method. Based on the data sets, some verification experiments are conducted. The experiment results show that the proposed model has apparent advantages over some other methods, both in recognition accuracy and training efficiency. At the same time, it has an acceptable on-line detecting performance.

The paper is structured as follows. In Section 2, some related theories and methods are reviewed. In Section 3, the model is established, and its structure graph is presented. In Section 4, the environment and data of the simulation experiments are given and results are discussed. Section 5 concludes the paper with a summary and remarks. Description of the Monte Carlo simulation functions of the eight patterns is displayed in the Appendix A.

2 Related Theories and Methods

2.1 AR (1) Autocorrelation Process

In the modern manufacturing process, the production tempo is getting faster and faster, and the automatic data acquisition technology is used more and more widely. As a result, the observed values of a specific variable get at different moment frequently present certain type of dependence with each other. That is called the autocorrelation phenomenon [23]. The autocorrelation in actual process is usually modeled with certain time series model in researches. Different time series model can be used to fit autocorrelation process, such as AR(p), MA(q), ARMA (p, q), etc. The literature research indicates that the AR (1) model appears most commonly. Its mathematic expression is as follows:

$$X_n = \mu + \phi(X_{n-1} - \mu) + \varepsilon_n \tag{1}$$

In Eq. (1): X_n and $X_{n-1}$are respectively the observed data get at the moment n and $n-1$; μ is the mean value of the process at controlled state; $\phi(-1 \le \varphi \le 1)$ is the autocorrelation coefficient; $\varepsilon_n \sim N(0, \sigma^2)$ is a normally distributed variable used to simulate the randomness of the process.

In actual production process, affected by certain assignable cause, process variable may takes on certain type of fluctuation, which can be seen from the control chart. Academia has defined these fluctuations as control chart patterns (CCP). There are eight major types of CCP, namely the Normal (NOR), the Upward Shift (US), the Downward Shift (DS), the Increasing Trend (IT), the Decreasing Trend (DT), the Cycle (CYC), the Systematic (SYS), and the Mixture (MIX). These CCPs can still take place in the process as there is autocorrelation exists. The Fig. 1 shows eight CCPs in autocorrelation process.

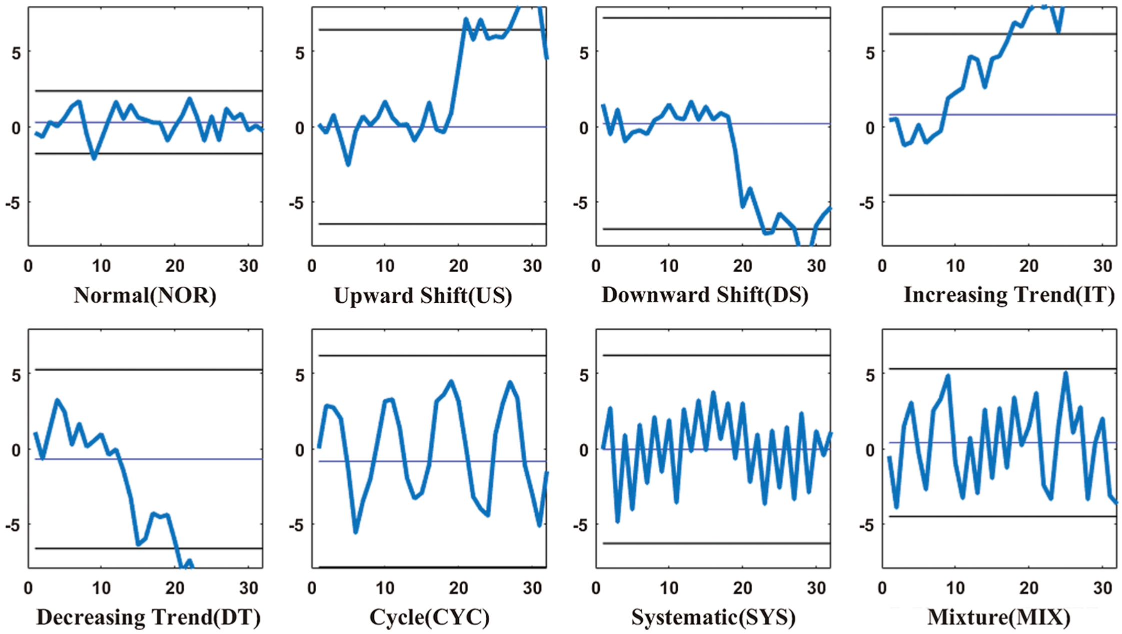


Figure 1: Control chart patterns in autocorrelation process

2.2 SVM Optimized by CS

2.2.1 Support Vector Machine

The classification problem using support vector machines can be described as follows. Given a sample set $(x_1, y_2), \ldots, (x_l, y_l), x \in R^n, y \in \{+1, -1\}$, where n is the dimension of the sample feature vector, y is the class label, l is the number of samples, and a decision function needs to be constructed to classify a new sample as correctly as possible. The basic structure of SVM is shown in Fig. 2.

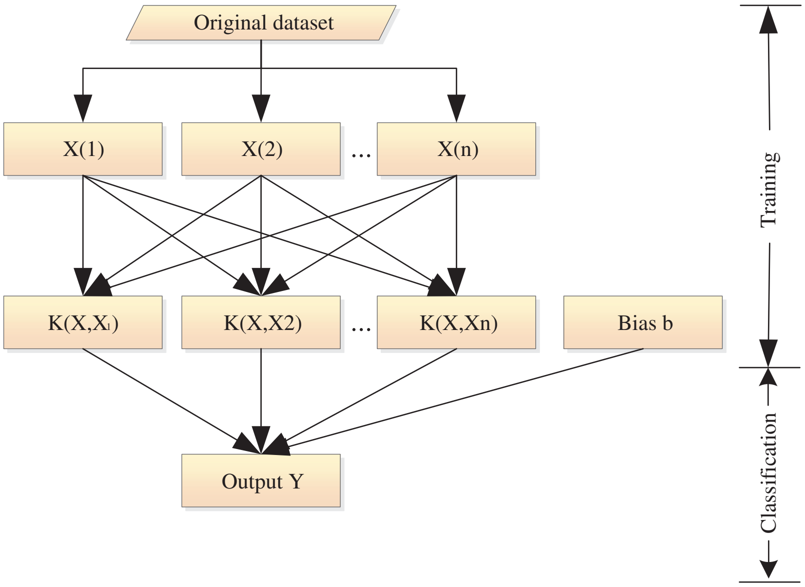


Figure 2: The basic structure of SVM

SVM is essentially a hyper-plane $\omega^T x + b = 0$ that can separate two types of linearly separable samples to the utmost extent, where the $\omega \in R^n$ is the normal vector of the hyper-plane and the b is the bias. In order for that, the distance between the two types of samples $2/\|\omega\|$ should be maximized. That means the $||\omega||$ should be minimized. Simultaneously, for handling samples that are not seriously linear separable, a compromise is made to accept misclassification of a few samples with penalty. Then the target function of SVM is described as follows.

$$\min J(\omega) = \frac{1}{2}\omega^T\omega + c\sum_{i=1}^{n}\xi_i \tag{2}$$

$$S.t. y_i\left(\omega^T x_i + b\right) \geq 1 - \xi_i, i = 1, 2, \cdots, n \tag{3}$$

In Eq. (2): ξ_i is slack variable for each sample; c is the penalty factor, which decides the fault tolerance of the classifier.

The most crucial characteristic of SVM is that it can transfer nonlinear separable samples in lower dimensional space to linear separable samples in higher dimensional space through using kernel function. The kernel function $k(x_i, x)$ is a type of symmetric function obeys the Mercer's theorem, which corresponds to the inner product of two vectors mapped by certain nonlinear transformation from lower to higher dimensional space. After replacing the inner product with the kernel function, the discriminant function of SVM can be obtained as follows:

$$f(x) = \text{sgn}\left[\sum_{SV} a_i y_i k(x_i, x) + b^*\right] \tag{4}$$

There are many types of kernel function, including the radial basis function (RBF), the sigmoid kernel, the linear kernel and the polynomial kernel, etc. Literature reviews show that the SVM with RBF has good performance in control chart pattern recognition. Therefore, the SVM with RBF is used as the pattern classifier in this study. The formula of RBF is as follows:

$$K(x, x_i) = \exp\left(-\|x - x_i\| / \left(2\sigma^2\right)\right) = \exp(-g\|x - x_i\|) \tag{5}$$

The parameter g is one of the two hyper-parameters of the SVM with RBF, and the penalty parameter c is the other. Both of them affect the classifying performance of the SVM significantly. Therefore, looking for the best pair of (c, g) is a crucial step in constructing our SVM based pattern recognition model.

2.2.2 CS Algorithm

CS is a new meta-heuristic search optimization algorithm illuminated by the biological characteristics in nature [24]. It combines the simulation of cuckoo birds' parasitic reproduction process with the Levy flight search principle. In nature, the cuckoo finds its nest position randomly. There are three ideal hypotheses for the cuckoo's search for the optimal nest: (1) The number of host nests n remains unchanged, allowing the host bird to discover and abandon non-self eggs with probability $p_a = [0, 1]$. (2) The best nest with high-quality eggs will be reserved for the next generation; (3) Birds randomly select nests and only lay one egg at a time. Based on the above three ideal conditions, the updated formula for the location and path of the cuckoo's host nest is as follows:

$$x_i^{t+1} = x_i^{(t)} + \alpha \otimes L(s, \lambda) \tag{6}$$

among them

$$L(s, \lambda) = \frac{\lambda\Gamma(\lambda)\sin(\pi\lambda/2)}{\pi}\left(\frac{1}{s^{1+\lambda}}\right), (s \gg 0) \tag{7}$$

In Eq. (6): $x_i^{(t)}$ is the $i - th$ nest's position in t iterations; α is the step length coefficient, usually $\alpha > 0$, which obeys the normal distribution and controls the step length.

After updating the position by Eq. (6), a random number $r(r \in [0, 1])$ is generated. Comparing r with p_a, if $r > p_a$, then randomly update the nest position, otherwise the nest position remains unchanged. The optimal nest position is selected by iterating and updating the nest position continuously.

3 The Recognition Model

The recognition model based on CS-SVM is shown in Fig. 3. It is composed of three modules, namely the data generation module, the CS optimization module and the pattern recognition module. The implementation process of the CS to optimize the hyper-parameters c, g of SVM is the core of this model, which is described in detail as follows:

Step 1: Initialize the basic parameters. When the probability that the host bird finds a non-self bird egg is 0.25, it is sufficient for most optimization problems, so $p_a = 0.25$ was set. The number of iterations is $time = 200$, the number of nests $n = 20$, the range of c is $[0.01, 128]$, the range of g is $[0.001, 128]$. The nest position $X_i, i \in [1, n]$ is randomly initialized, and the fitness function is defined as $f(x), X = [X_1, X_2, \cdots, X_n]^T$. The nest position is brought into the fitness function, and the nest with the best fitness position is selected and reserved for the next generation.

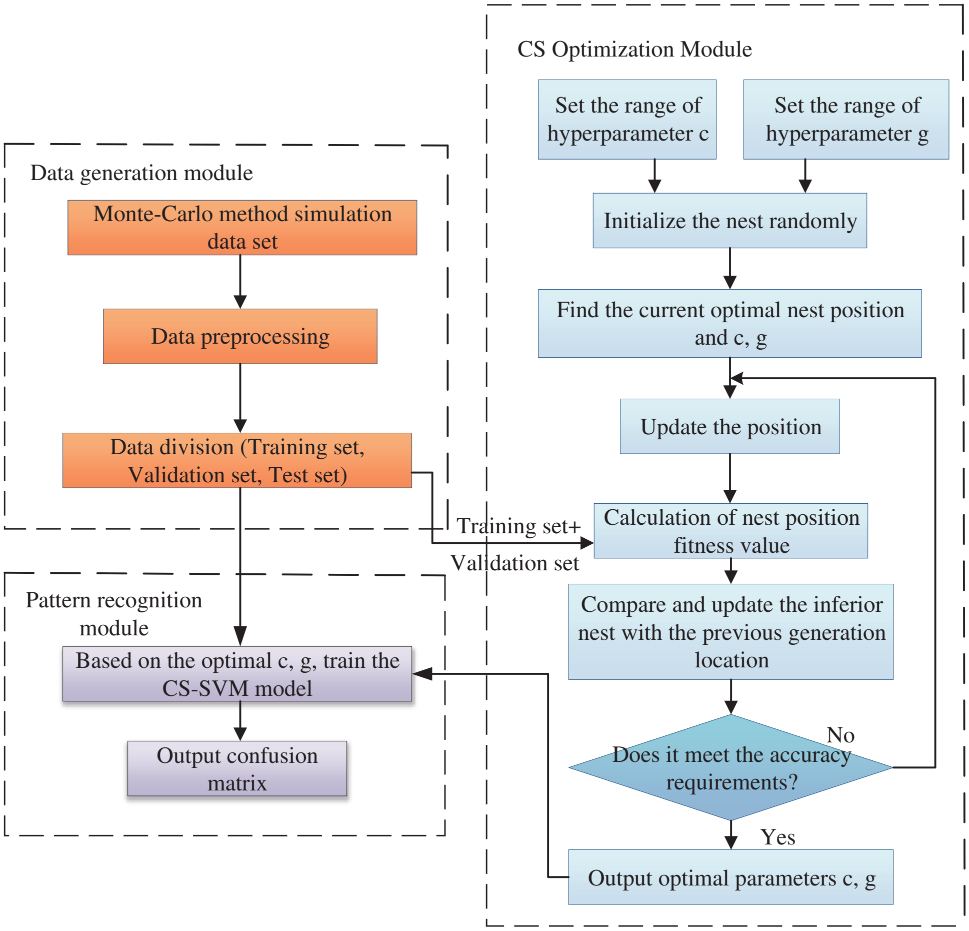

CS Optimization Module
Set the range of hyperparameter c
Set the range of hyperparameter g
Initialize the nest randomly
Find the current optimal nest position and c, g
Update the position
Calculation of nest position fitness value
Compare and update the inferior nest with the previous generation location
Does it meet the accuracy requirements?
No
Yes
Output optimal parameters c, g
Data generation module
Monte-Carlo method simulation data set
Data preprocessing
Data division (Training set, Validation set, Test set)
Training set+ Validation set
Pattern recognition module
Based on the optimal c, g, train the CS-SVM model
Output confusion matrix

Figure 3: Recognition model based on CS-SVM

Step 2: The position and state of the non-optimal nest are updated by using Eq. (6). The updated nest position is used to recalculate the fitness function value, which is compared with that of the best nest position reserved by the previous generation. When the current value is better, the new nest position is reserved to replace the old best nest.

Step 3: After the nest position is updated, the random number r is compared with p_a. Then the fitness function is used to test whether the updated nest position is better in comparison with the previous generation to determine the new optimal nest position.

Step 4: If the maximum number of iterations or required accuracy is reached, the obtained optimal bird nest position is accepted as the global optimal solution; otherwise, turn back to step 2 and continue to iterate and update.

Step 5: Output the position of the global optimal nest, namely the best hyper-parameters c, g.

4 Simulation Experiments

4.1 Experimental Environment and Data

In order to verify the performance of the proposed model, which is established through programming in the MATLAB2018, and hereinto, the SVM is realized by the Libsvm toolbox. Then, some simulation experiments with this model are carried out in the MATLAB2018. The performance indicators of the computer used are CPU2.4GHZ, RAM12.0G. Following the conventional research methods in this field, eight types of basic CCP sample (as described in 2.1) of AR (1) process are generated by Monte carol simulation method. These samples are divided into three mutually exclusive sets, including the training set, the validation set and the test set. The number of samples for each pattern in each set is shown in Tab. 1. To reflect the diversity of abnormal amplitudes in actual processes, exception parameters of the abnormal patterns take values within certain specific range. The parameter choices of all patterns are shown in Tab. 2. With reference to other literatures, the dimension of the pattern sample vector (that is, the width of the online recognition window) is set equal to 32.

Table 1: Sample numbers

Data set	NOR	US	DS	IT	DT	CYC	SYS	MIX
Training set	500	720	720	720	720	720	720	720
Validation set	90	90	90	90	90	90	90	90
Test set	90	90	90	90	90	90	90	90

Table 2: Parameter settings of the eight patterns

Pattern type	Parameter value	Starting points
NOR	μ=80, σ^2=0.25	
US	α=(1.0 : 0.25 : 3.0)	$\alpha \leq 2.0, t = 12$ $\alpha > 2.0, t = 19$
DS	α=(−3.0 : 0.25 : −1.0)	$\alpha \leq -2.0, t = 12$ $\alpha > -2.0, t = 19$
IT	γ=(0.10 : 0.02 : 0.26)	$\gamma \leq 0.18, t = 1$ $\gamma > 0.18, t = 6$
DT	γ=(−0.26 : 0.02 : −0.10)	$\gamma \leq -0.18, t = 1$ $\gamma > -0.18, t = 6$
CYC	l=(1.0 : 0.25 : 3.0), $T = 8$	t = 0
SYS	g=(1.0 : 0.25 : 3.0)	t = 0
MIX	τ=(1.0 : 0.25 : 3.0)	t = 0

For the patterns of US, DS, IT, and DT, different starting points of exception are set for different abnormity amplitudes. Small amplitudes have smaller abnormal starting points, for example, in US, $t = 12$ when $\alpha \leq 2.0$. The purpose is to make the abnormal pattern samples with smaller amplitude contain more information, so that the trained model can identify which more easily.

4.2 Optimization Process of CS-SVM

In each iteration of the CS, the CS-SVM model is trained after being given the hyper-parameters, and calculated recognition accuracy with the verification set. The result is taken as the fitness value. Without loss of generality, the hyper-parameters are optimized when the autocorrelation coefficient $\phi = 0.5$ and also used for other autocorrelation levels. Five times of optimization have been carried out, and the corresponding convergence procedures are shown in Fig. 4. All of them converge rapidly within 20 iterations, and reach the fitness value of 98.33%. The earliest one converges at the 11th generation, which get c = 4.9934, g = 0.3675. Hence, this couple of values are set as the model's hyper-parameters.

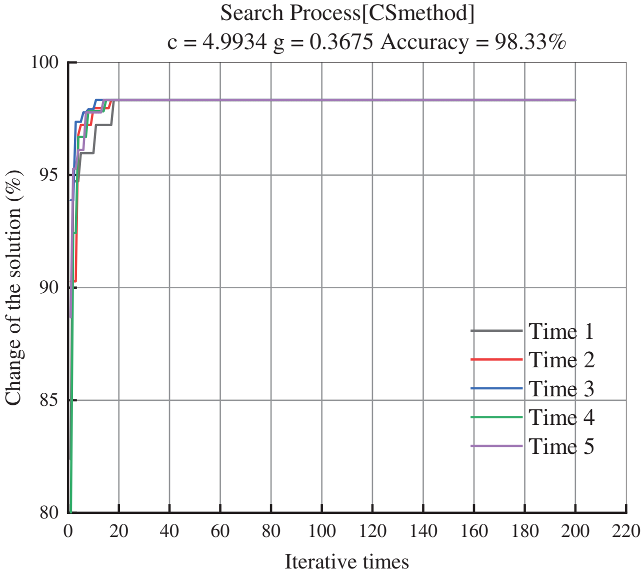


Figure 4: Iteration of CS-SVM model

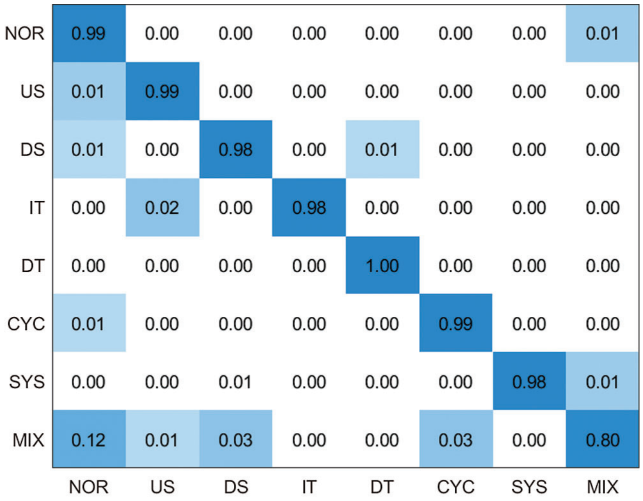
4.3 Recognition Accuracy Test

Recognition accuracy test of this model is carried out on the test set. For the case of $\phi = 0.5$, the confusion matrix of the eight types of pattern is obtained, as shown in Fig. 5. It shows that the recognition accuracies of seven patterns achieve higher than 98%. Among them, the DT pattern gets the highest accuracy, reaching 100%. But the MIX pattern gets a relatively low accuracy, only at 80%. The reason is that the MIX pattern essentially obeys normal distribution, which is different from the NOR pattern only in mean and variance, so that easy to be misjudged as the NOR pattern. The average accuracy of the eight types of pattern reaches 96.25%.

Table 3: The setting of main parameters

Model type	The setting of main parameters
CS-SVM	$time$ = 200, dim = 2, n = 20, pa = 0.25
PSO-SVM	C_1 = 1.5, C_2 = 1.7, maxgen =200, k = 0.6, $Sizepop$ = 20, wV = 1, wP = 1
GA-SVM	Maxgen = 200, $sizepop$ = 20

The average accuracy of processes with some other different level of autocorrelation coefficients ($\phi \in \{-0.9, -0.7, -0.5, -0.3, -0.1, 0, 0.1, 0.3, 0.5, 0.7, 0.9\}$) are shown in Fig. 6. It can be seen that all of these accuracies are above 95%, and the accuracies of the positive correlation cases are slightly higher than those of the negative ones. In addition, the accuracy reaches 98.13% in the case of $\phi = 0$, which means that the process is not affected by autocorrelation. These results mean that this model can handle processes with different levels of autocorrelation.



	NOR	US	DS	IT	DT	CYC	SYS	MIX
NOR	0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.01
US	0.01	0.99	0.00	0.00	0.00	0.00	0.00	0.00
DS	0.01	0.00	0.98	0.00	0.01	0.00	0.00	0.00
IT	0.00	0.02	0.00	0.98	0.00	0.00	0.00	0.00
DT	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00
CYC	0.01	0.00	0.00	0.00	0.00	0.99	0.00	0.00
SYS	0.00	0.00	0.01	0.00	0.00	0.00	0.98	0.01
MIX	0.12	0.01	0.03	0.00	0.00	0.03	0.00	0.80

Figure 5: Confusion matrix for the case of $\phi = 0.5$

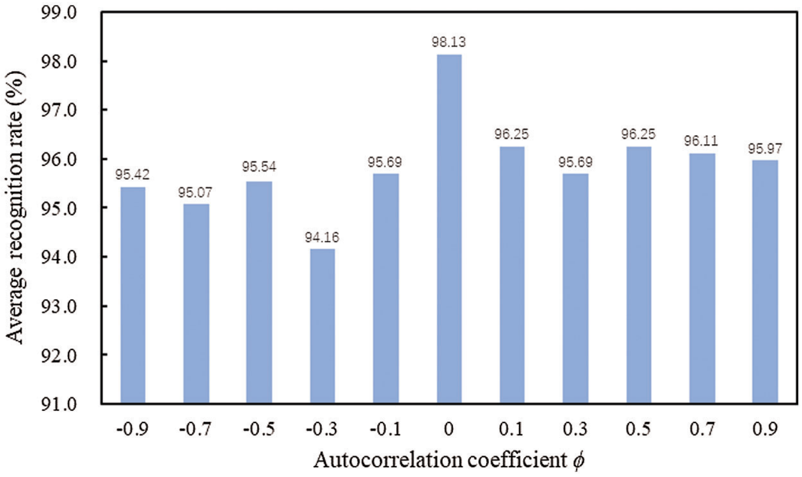


Figure 6: Recognition accuracy in different level of autocorrelation

4.4 Comparative Experiment

In order to further verify the superiority of this model, a comparative experiment was carried out based on the same data sets. The chosen comparative objects are the model based on SVM optimized by particle swarm optimization (PSO-SVM) and the model based on SVM optimized by genetic algorithm (GA-SVM). The reason is that both PSO and GA are intelligent optimization algorithms, which are often used to optimize the hyper-parameters of SVM. Five times of optimization are conducted for the case of autocorrelation coefficients $\phi = 0.5$, and then the average convergence time and average recognition accuracy on the test set are used as the comparative indicators. The main parameter settings of these models are shown in Tab. 3. The comparative results are shown in Fig. 7.

It can be seen from Fig. 7a that the proposed model obtains the highest average recognition accuracy with $c = 4.9934$ and $g = 0.3675$. The PSO-SVM based model takes the second place (the accuracy is 95.54%) with $c = 98.4233$ and $g = 0.0366$. The GA-SVM based model get the lowest accuracy (91.19%) with $c = 100.4169$ and $g = 0.0039$. It can be seen from Fig. 7b that the proposed model takes the shortest time (80.65 minutes) to converge, which is much less than those of the PSO-SVM based model (240.52 minutes) and the GA-SVM based model (293.28 minutes). Thus, we believe that the CS-SVM based model we proposed has the best abnormal pattern identification performance for autocorrelation process in comparison with the two comparative objects.

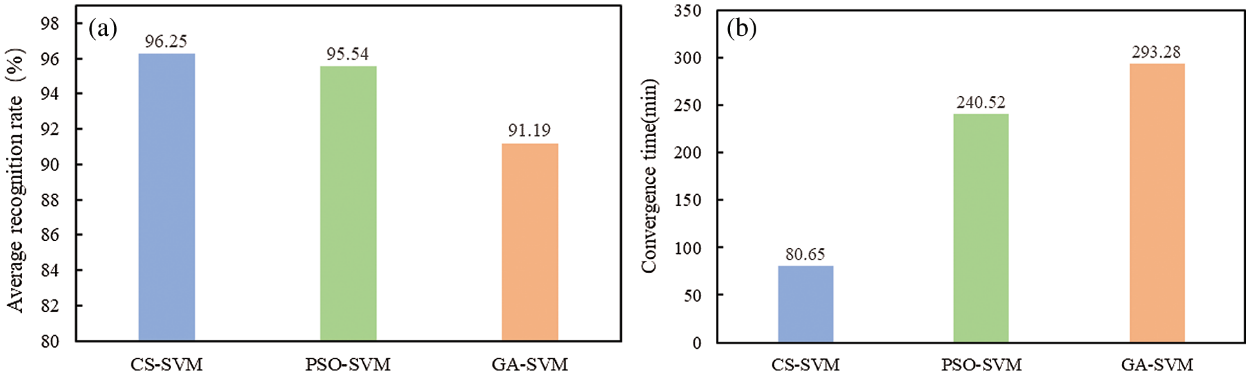


Figure 7: Performance comparison of model optimized by three algorithms. (a) Comparison of recognition accuracy and (b) Comparison of convergence time

4.5 ARL Measure

Average run length (ARL) is a key indicator of the online performance of control chart or other process anomaly detectors. ARL can be measured through simulation method. The usual practice is to simulate lots of data streams first, and then fetch data from each data stream with a sliding window, till a normal pattern sample is misrecognized as abnormal (ARL_0) or an abnormal pattern is identified (ARL). In this study, 3000 in-control data streams and 3000 × 9 × 7 out-of-control data streams are separately generated for each autocorrelation level. The ARL_0 reflexes the probability of occurrence of the type I error in the model, and should be controlled as close as possible to the theoretical value (370 for the univariate process) before measuring ARL. For that reason, the try and error method is applied, i.e., the number of the NOR sample in the train data set is adjusted according to the achieved ARL_0 value till it reaches slightly higher than 370. In addition, the SRL (standard deviation of ARL) is also calculated to measure the stability of ARL performance.

In order to verify the online performance of the proposed CS-SVM based model, an SVM based model in reference [15] is chosen as a comparative object, and the comparative data are obtained from its Tab. 3. The Tab. 4 shows the comparison of the average ARL values and the average SRL values of the proposed model and those of the comparative object, which are calculated for different abnormity magnitudes of each abnormal pattern under different level of autocorrelation. For comparative purpose, the ARL value of the proposed model which is not lower than that of the comparative object is highlighted in bold in the Tab. 4.

It can be seen that the proposed model performs better for the US, DS, IT and DT pattern as $\phi \geq 0$, and better for the SYS and MIX pattern as $\phi < 0$. However, the aggregated average ARL values indicate that the proposed model takes an advantageous position, since it obtains lower ARL values than those of the comparative object for 8 of the 11 levels of autocorrelation. At the same time, the aggregated average SRL values of the proposed model are very close to those of the comparative object. In that, the proposed model can be regarded as having an acceptable online performance.

Table 4: Comparison of ARL and SRL values between the CS-SVM based model and the referenced SVM based model

ϕ	Model	US		DS		IT		DT		CYC		SYS		MIX		Aggregated average	
		ARL	SRL	ARL	SRL	ARL	SRL	ARL	SRL	ARL	SRL	ARL	SRL	ARL	SRL	ARL	SRL
0.9	CS-SVM based	**6.5**	2.5	**6.5**	2.6	**12.3**	2.95	**12.2**	3.1	13.3	3.9	12.3	3.5	12.4	2.3	**10.8**	3.0
	SVM based	9.8	3.6	9.6	4.0	12.3	3.90	12.4	4.2	10.7	1.7	10.8	2.3	10.7	2.8	10.9	3.2
0.7	CS-SVM based	**6.2**	2.5	**6.1**	2.5	**12.1**	2.96	**12.1**	3.0	12.7	3.6	12.2	2.2	13.3	2.9	**10.7**	2.8
	SVM based	9.2	3.4	9.1	3.0	13.6	2.90	12.5	3.1	11.3	3.4	10.3	2.1	10.2	2.6	10.9	2.9
0.5	CS-SVM based	**6.8**	2.6	**6.3**	2.6	**12.8**	2.97	**12.3**	2.0	13.0	4.5	11.9	1.9	12.4	2.2	**10.8**	2.7
	SVM based	8.9	2.2	8.5	1.8	13.5	2.20	13.4	2.2	12.4	3.2	10.1	1.7	10.3	2.0	11.0	2.2
0.3	CS-SVM based	**7.2**	2.8	**6.9**	2.8	**11.1**	2.94	**12.9**	1.5	12.3	3.3	10.7	3.7	12.1	2.4	**10.5**	2.8
	SVM based	8.5	1.4	8.2	1.5	11.4	1.70	13.5	1.7	12.2	2.7	10.5	1.6	10.0	1.8	10.6	1.8
0.1	CS-SVM based	**7.1**	2.7	**7.6**	2.8	**12.9**	2.81	**13.1**	1.9	14.3	2.6	10.4	3.7	11.9	1.9	11.1	2.6
	SVM based	7.9	1.3	7.7	1.1	13.0	1.30	13.2	1.2	13.0	2.8	11.1	1.5	10.7	1.9	10.9	1.6
0	CS-SVM based	**7.6**	1.8	**7.1**	1.8	**13.4**	1.94	12.7	2.0	14.4	2.7	**9.5**	1.7	**10.3**	1.5	10.7	1.9
	SVM based	8.1	1.1	8.4	1.2	11.3	1.30	10.4	1.3	9.5	2.4	10.6	1.6	10.4	1.8	9.8	1.5
–0.1	CS-SVM based	8.1	2.8	7.1	2.8	13.6	2.82	12.8	2.9	15.1	2.9	**8.7**	1.4	**10.7**	1.4	10.9	2.4
	SVM based	7.3	1.1	7.5	1.1	13.8	1.20	13.7	1.2	12.1	2.5	10.3	1.9	10.7	1.5	10.8	1.5
–0.3	CS-SVM based	7.7	2.8	7.8	3.0	13.1	2.90	13.3	3.1	15.5	2.2	**7.3**	3.1	**10.8**	1.7	**10.8**	2.7
	SVM based	7.0	1.1	7.2	1.0	13.2	1.10	13.5	1.1	12.5	2.3	12.0	2.1	12.2	1.8	11.1	1.5
–0.5	CS-SVM based	7.0	2.9	8.4	3.0	14.1	3.08	13.6	3.2	16.2	2.2	**6.9**	2.9	**11.7**	3.5	**11.1**	3.0
	SVM based	7.2	1.0	7.1	1.0	13.5	1.00	13.8	0.9	13.2	2.4	12.4	2.9	12.1	2.3	11.3	1.6
–0.7	CS-SVM based	8.8	2.9	8.5	3.0	13.9	3.00	13.6	3.0	17.0	6.1	**7.2**	2.9	**13.1**	4.6	**11.7**	3.6
	SVM based	6.6	0.9	6.9	0.9	13.4	1.10	13.5	1.1	12.8	2.3	14.1	4.9	15.2	4.6	11.8	2.3
–0.9	CS-SVM based	8.8	3.0	8.8	3.1	13.9	2.97	13.8	3.0	18.2	6.8	**6.7**	2.6	**14.2**	8.4	**12.1**	4.3
	SVM based	7.0	1.1	6.8	1.0	13.3	1.20	13.0	1.3	12.4	2.1	17.6	7.3	17.6	6.5	12.5	2.9

5 Conclusion

This study uses CS to optimize the two hyper-parameters of SVM (CS-SVM), and then based on which to establish a recognition model for abnormal patterns in autocorrelation process. A series of simulation experiments have been conducted to test the performances of this model. The experiment results show that the established model can achieve higher recognition accuracy in comparison with the model based on SVM optimized by the PSO or the GA. In the meantime, it takes much less time to optimize the hyper-parameters for this model. That means this model has higher training efficiency, considering that the parameter optimization procedure is involved in the training process. Furthermore, the model shows good recognition accuracy for each tested autocorrelation level (whether positive or negative autocorrelation), indicating its broad applicability. At last, the ARL values of the model at different autocorrelation levels are measured, which are generally better than those of the comparative model from the reference. That indicates the model also possesses an acceptable online performance. Identification of abnormal patterns in autocorrelation process is still in the exploratory stage, and the proposed model provides a new way for it. Currently, we have verified the model's effectiveness through simulation experiments. Testing the model's effectiveness for other types of autocorrelation processes except AR (1) and studying how to use it in the actual manufacturing process will be our next work.

Acknowledgement: I want to take this chance to thanks my tutor——Bo Zhu. In composing this paper, he gives me much academic and constructive advice and helps me correct my essay. Besides these, he also allowed me to do my teaching practice. At the same time, I want to thank my friends Chunmei Chen, Kaimin Pang and Yuwei Wan. They participated much in this research. Finally, I'd like to thank all my friends, especially my three lovely roommates, for their encouragement and support.

Funding Statement: This research was financially supported by the National Key R&D Program of China (2017YFB1400301).

Conflicts of Interest: The authors declare that they have no interest in reporting regarding the present study.

References

[1] Y. Li, J. Li, P. Shi and X. Qin, "Building an open cloud virtual dataspace model for materials scientific data," *Intelligent Automation and Soft Computing*, vol. 25, no. 3, pp. 615–624, 2019.

[2] X. Zhang, J. Duan, W. Sun and N. I. Badler, "An integrated suture simulation system with deformation constraint under a suture control strategy," *Computers, Materials & Continua*, vol. 60, no. 3, pp. 1055–1071, 2019.

[3] J. Su, Z. Sheng, A. X. Liu and Y. Chen, "A partitioning approach to RFID identification," *IEEE/ACM Transactions on Networking*, vol. 28, no. 5, pp. 2160–2173, 2020.

[4] J. Su, Z. Sheng, A. X. Liu, Y. Han and Y. Chen, "Capture-aware identification of mobile RFID tags with unreliable channels," *IEEE Transactions on Mobile Computing*, vol. 14, no. 8, pp. 1–14, 2020.

[5] S. Wang, T. Yue and M. A. Wahab, "Multiscale analysis of the effect of debris on fretting wear process using a semi-concurrent method," *Computers, Materials & Continua*, vol. 62, no. 1, pp. 17–36, 2020.

[6] Y. R. Shiau, "Inspection allocation planning for a multiple quality characteristic advanced manufacturing system," *International Journal of Advanced Manufacturing Technology*, vol. 21, no. 7, pp. 494–500, 2003.

[7] D. C. Montgomery and C. M. Mastrangelo, "Some statistical process control methods for autocorrelated data," *Journal of Quality Technology*, vol. 23, no. 3, pp. 179–204, 1991.

[8] L. C. Alwan and H. V. Roberts, "Time-series modeling for statistical process control," *Journal of Business & Economic Statistics*, vol. 6, no. 1, pp. 87–95, 1988.

[9] H. D. Maragah and W. H. Woodall, "The effect of autocorrelation on the retrospective X-chart," *Journal of Statistical Computation and Simulation*, vol. 40, no. 1–2, pp. 29–42, 1992.

[10] D. C. Montgomery and W. H. Woodall, "A discussion on statistically-based process monitoring and control," *Journal of Quality Technology*, vol. 29, no. 2, pp. 121, 1997.

[11] G. C. Runger, F. B. Alt and D. C. Montgomery, "Contributors to a multivariate statistical process control chart signal," *Communications in Statistics*, vol. 25, no. 10, pp. 2203–2213, 1996.

[12] J. Sun, "Residual error control chart of autocorrelation process," *Journal of Tsinghua University (Natural Science Edition)*, vol. 42, no. 6, pp. 735–738, 2002.

[13] M. Zhang and W. M. Cheng, "Control chart pattern recognition based on adaptive particle swarm algorithm and support vector machine," *Industrial Engineering*, vol. 15, no. 5, pp. 128–132, 2012.

[14] D. F. Cook and C. U. Chiu, "Using radial basis function neural networks to recognize shifts in correlated manufacturing process parameters," *IIE Transactions*, vol. 30, no. 3, pp. 227–234, 1998.

[15] S. Y. Lin, R. S. Guh and Y. R. Shiue, "Effective recognition of control chart patterns in autocorrelated data using a support vector machine based approach," *Computers & Industrial Engineering*, vol. 61, no. 4, pp. 1123–1134, 2011.

[16] B. Zhu, B. B. Liu, Y. Wan and S. R. Zhao, "Recognition of control chart patterns in auto-correlated process based on random forest," *2018 IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering*, vol. 1, pp. 53–57, 2018.

[17] Y. M. Liu and Z. Y. Zhao, “Recognition of abnormal quality patterns based on feature selection and SVM,” *Statistics and Decision*, vol. 10, no. 502, pp. 47–50, 2018.

[18] Q. Wang and X. Wang, “Parameters optimization of the heating furnace control systems based on BP neural network improved by genetic algorithm,” *Journal on Internet of Things*, vol. 2, no. 2, pp. 75–80, 2020.

[19] T. Huang, Y. Q. Wang and X. G. Zhang, “Color image segmentation algorithm based on GS-SVM,” *Electronic Measurement Technology*, vol. 40, no. 7, pp. 105–108, 2017.

[20] C. Z. Wang, Y. L. Du and Z. W. Ye, “P2P traffic recognition based on the optimal ABC-SVM algorithm,” *Computer Application Research*, vol. 35, no. 2, pp. 582–585, 2018.

[21] Y. Xue, T. Tang, W. Pang and A. X. Liu, “Self-adaptive parameter and strategy based particle swarm optimization for large-scale feature selection problems with multiple classifiers,” *Applied Soft Computing*, vol. 88, no. 4, pp. 106031, 2020.

[22] C. Y. Liu, A. Y. Wang and R. Li, “Mine multi-mode wireless signal modulation recognition based on GA-SVM algorithm,” *Science Technology and Engineering*, vol. 20, no. 6, pp. 2186–2191, 2020.

[23] B. B. Liu, B. Zhu, Y. W. Wan and Y. Y. Song, “Using optimized probabilistic neural networks to identify autocorrelation process abnormalities,” *Manufacturing Automation*, vol. 40, no. 2, pp. 70–73, 2018.

[24] H. R. Xue, K. H. Zhang, B. Li and C. H. Peng, “Transformer fault diagnosis based on cuckoo algorithm and support vector machine,” *Power System Protection and Control*, vol. 43, no. 8, pp. 15–20, 2018.

Appendix A. Mathematical Expressions of Eight CCPs

In this research, the Monte Carlo simulation approach was used to generate the required sets of CCPs for training, test, and validation data set. The CCP herein is expressed in a general form that consists of the process mean, the common-cause variation, and a special disturbance from specific causes.

Process Simulator

$X_t = \mu + \varepsilon_n + d_t$

where

μ = process means in the controlled state

ε_n = common-cause variation at time t (by the polar method), following a normal distribution with zero mean and standard deviation σ, where

σ = process standard deviation when the process is in control and was fixed at 0.25 in this research

d_t = special disturbance at time t

Normal (NOR): $d_t = 0$

Shift (US/DS): $d_t = \lambda \times \alpha$

where

λ = parameter to determine the position of shifting ($\lambda = 0$ before shifting; $\lambda = 1$ after shifting)

α = displacement of mean in terms of σ

Trend (IT/DT): $d_t = \gamma \times t$

where

γ = trend slope in terms of σ

Cycle (CYC): $d_t = l \times \sin\left(\frac{2\pi t}{T}\right)$

where

l = cycle amplitude in terms of σ

T **=** cycle period (T = 8 in this research)

Systematic (SYS): $d_t = g \times (-1)^t$

where

g = magnitude of the systematic pattern in terms of σ, determining the fluctuations above or below the process mean

Mixture (MIX): $d(t,p) = \tau \times (-1)^h$

where

p = a random number ($0 < p < 1$)

τ = magnitude of the mixture pattern in terms of σ, determining the fluctuations above or below the process mean

h = 0 if $p < \tau p$, 1 if $p > \tau p$