Machine Learning Control Charts for Monitoring Serially Correlated Data

Xiulin Xie and Peihua Qiu

Abstract Some control charts based on machine learning approaches have been developed recently in the statistical process control (SPC) literature. These charts are usually designed for monitoring processes with independent observations at different observation times. In practice, however, serial data correlation almost always exists in the observed data of a temporal process. It has been well demonstrated in the SPC literature that control charts designed for monitoring independent data would not be reliable to use in applications with serially correlated data. In this chapter, we suggest using certain existing machine learning control charts together with a recursive data de-correlation procedure. It is shown that the performance of these charts can be substantially improved for monitoring serially correlated processes after data de-correlation.

Xiulin Xie
Department of Biostatistics, University of Florida, 2004 Mowry Road, Gainesville, FL 32610. e-mail: xiulin.xie@ufl.edu

Peihua Qiu
Department of Biostatistics, University of Florida, 2004 Mowry Road, Gainesville, FL 32610. e-mail: pqiu@ufl.edu

1 Introduction

In recent years, machine learning approaches have attracted much attention in different research areas, including statistical process control (SPC) (e.g., Aggarwal 2018, Breiman 2001, Carvalho et al. 2019, Göb 2006, Hastie et al. 2001). Some control charts based on different machine learning algorithms have been developed in the SPC literature. For instance, the k-nearest neighbors (KNN), random forest (RF) and support vector machines (SVM) have been used in developing SPC control charts. Most of these existing machine learning control charts are based on the assumption that process observations at different observation times are independent of each other. In practice, however, serial data correlation almost always exists in a time series data. It has been well demonstrated in the SPC literature that control charts designed for monitoring independent data would not be reliable to use when serial data correlation exists (e.g. Apley and Tsung 2002, Knoth and Schmid 2004, Lee and Apley 2011, Li and Qiu 2020, Psarakis and Papaleonida 2007, Qiu, Li, and Li 2020, Runger and Willemain 1995, Weiß 2015, Xue and Qiu 2020). Thus, it's necessary to improve these machine learning control charts by overcoming that limitation. This paper aims to address this important issue by suggesting to apply a recursive data de-correlation procedure to the observed data before an existing machine learning control chart is used.

In the SPC literature, there has been some existing discussion about process monitoring of serially correlated data (e.g., Alwan and Roberts 1995, Capizzi and Masarotto 2008, Prajapati and Singh 2012, Psarakis and Papaleonida 2007). Many such existing methods are based on parametric time series modeling of the observed process data and monitoring of the resulting residuals. For instance, Lee and Apley (2011) proposed an exponentially weighted moving average (EWMA) chart for monitoring correlated data by assuming the *in-control* (IC) process observations to follow an ARMA model. In practice, however, the assumed parametric time series models may not be valid, and consequently these control charts may be unreliable to use (e.g., Li and Qiu 2020). Recently, Qiu, Li, and Li (2020) suggested a more flexible data de-correlation method without using a parametric time series model for univariate cases. It only requires the serial data correlation to be stationary and short-range (i.e., the correlation between two observations become weaker when the observation times get farther away). A multivariate extension of that method was discussed in Xue and Qiu (2020). Numerical studies show that such sequential data de-correlation approaches perform well in different cases. In this paper, we suggest improving some existing machine learning control charts by applying such a data de-correlation procedure to the observed process observations in advance. The modified machine learning control charts can handle cases with multiple numerical quality variables, and the quality variables could be continuous numerical or discrete. Numerical studies show that the performance of these modified machine learning control charts is substantially better than their original versions for monitoring processes with serially correlated data in various different cases.

The remaining parts of this paper are organized as follows. In Section 2, the proposed modification for some existing machine learning control charts are described

in detail. Numerical studies for evaluating their performance are presented in Section 3. A real-data example to demonstrate the application of the modified control charts is discussed in Section 4. Finally, some remarks conclude the article in Section 5.

2 Improve Some Machine learning Control Charts for Monitoring Serially Correlated Data

This section is organized in three parts. In Subsection 2.1, some representative existing machine learning control charts are briefly described. In Subsection 2.2, a recursive data de-correlation procedure for the observed sequential data is introduced in detail. Then, the modified machine learning control charts, in which the recursive data de-correlation procedure is applied to the observed data before the original machine learning control charts, are discussed in Subsection 2.3.

2.1 Description of some representative machine learning control charts

Classification is one of the major purposes of supervised machine learning, and many machine learning algorithms like the artificial neural networks, RF and SVM have demonstrated a good performance in accurately classifying input data after learning the data structure from a large training data. Since an SPC problem can be regarded as a binary class classification problem, in which each process observation needs to be classified into either the IC or the *out-of-control* (OC) status during phase II process monitoring, several machine learning algorithms making use of both IC and OC historical data have been employed for process monitoring. For instance, Zhang, Tsung, and Zou (2015) proposed an EWMA control chart based on the probabilistic outputs of a SVM classifier that needs to be built by using both IC and OC historical data. Several other classifiers like the KNN and linear discriminant analysis were also proposed for process monitoring (e.g., Li, Zhang, Tsung, and Mei 2020; Sukchotrat, Kim, Tsui, and Chen 2011). In many SPC applications, however, few OC process observations would be available in advance. For instance, a production process is often properly adjusted during the Phase I SPC, and a set of IC data is routinely collected afterwards for estimating the IC process distribution or some of its parameters (Qiu 2014, Chapter 1). Thus, for such applications, an IC data is usually available before the Phase II SPC, but the OC process observations are often unavailable. To overcome this difficulty, some creative ideas like the *artificial contrast*, *real-time contrast*, and *one class classification* were proposed to develop control charts without assuming the availability of OC process observations during the design stage of the related charts. Several representative machine learning control charts based on these ideas are briefly introduced below.

2.1.1 Control chart based on artificial contrasts

Tuv and Runger (2004) proposed the idea of *artificial contrast* to overcome the difficulty that only IC data are available before the Phase II process monitoring in certain SPC applications. By this idea, an artificial dataset is first generated from a given off-target distribution (e.g., Uniform) and observations in that dataset are regarded as OC observations. Then, a machine learning algorithm (e.g., RF) is applied to the training dataset that consists of the original IC dataset, denoted as $\mathcal{X}_{IC}$, and the artificial contrast dataset, denoted as $\mathcal{X}_{AC}$. The classifier obtained by the RF algorithm is then used for online process monitoring. Hwang et al. (2007) studied the performance of such machine learning control charts by using both the RF and SVM algorithms. These machine learning control charts suffer two major limitations. First, their classification error rates cannot be transferred to the traditional average run length (ARL) metric without the data independence assumption. Second, their decisions at a given time point during phase II process monitoring are made based on the observed data at that time point only, and they have not made use of history data. To overcome these limitations, Hu and Runger (2010) suggested the following modification that consisted of two major steps. i) For process observation $\mathbf{X}_n$ at a given time point n, the log likelihood ratio is first calculated as

$$l_n = \log\left[\hat{p}_1(\mathbf{X}_n)\right] - log\left[\hat{p}_0(\mathbf{X}_n)\right],$$

where $\hat{p}_1(\mathbf{X}_n)$ and $\hat{p}_0(\mathbf{X}_n)$ are the estimated probabilities of $\mathbf{X}_n$ in each class obtained by the RF classifier. ii) A modified EWMA chart is then suggested with the following charting statistic:

$$E_n = \lambda l_n + (1-\lambda)E_{n-1},$$

where $\lambda \in (0, 1]$ is a weighting parameter. This control chart is denoted as AC, representing "artificial contrast". Obviously, like the traditional EWMA charts, the charting statistic E_n of AC is a weighted average of the log likelihood ratios of all available observations up to the time point n.

As suggested by Hu and Runger (2010), the control limit of AC can be determined by the following 10-fold cross-validation (CV) procedure. First, 90% of the IC dataset $\mathcal{X}_{IC}$ and the artificial contrast dataset $\mathcal{X}_{AC}$ is used to train the RF classifier. Then, the E_n with a control limit h is applied to the remaining 10% of the IC dataset $\mathcal{X}_{IC}$ to obtain a run length (RL) value. The above CV procedure is then repeated for $C = 1,000$ times, and the average of the C RL values is used to approximate the ARL_0 value for the given h. Finally, h can be searched by a numerical algorithm (e.g., the bisection searching algorithm) so that the assumed ARL_0 value is reached.

2.1.2 Control chart based on real time contrasts

The artificial contrasts $\mathcal{X}_{AC}$ used in AC are generated from a subjectively chosen off-target distribution (e.g., Uniform), and thus may not represent the actual OC observations well. Consequently, the RF classifier trained using $\mathcal{X}_{IC}$ and $\mathcal{X}_{AC}$ may

not be effective for monitoring certain processes. To improve the chart AC, Deng, Runger, and Tuv (2012) propose a *real time contrast (RTC)* approach, in which the most recent observations within a moving window of the current time point are used as the contrasts. In their proposed approach, the IC dataset is first divided into two parts: a randomly selected N_0 observations from $\mathcal{X}_{IC}$, denoted as $\mathcal{X}_{IC_0}$, is used for training the RF classifier, the remaining IC data, denoted as $\mathcal{X}_{IC_1}$, is used for determining the control limit. The process observations in a window of the current observation time point n are treated as OC data and denoted as $\mathcal{X}_{AC_n} = \{\mathbf{X}_{n-w+1}, \mathbf{X}_{n-w+2}, \ldots, \mathbf{X}_n\}$, where w is the window size. Then, the RF classifier can be re-trained sequentially over time using the training dataset that combines $\mathcal{X}_{IC_0}$ and $\mathcal{X}_{AC_n}$, and the decision rule can be updated accordingly once the new observation $\mathbf{X}_n$ is collected at time n.

Deng et al. (2012) suggested using the following estimated "out-of-bag" correct classification rate for observations in $\mathcal{X}_{IC_0}$ as the charting statistic:

$$P_n = \frac{\sum P_{OOB}(\mathbf{X}_i) I(\mathbf{X}_i \in \mathcal{X}_{IC_0})}{|\mathcal{X}_{IC_0}|},$$

where $|\mathcal{X}_{IC_0}|$ denotes the number of observations in the set $\mathcal{X}_{IC_0}$, and $P_{OOB}(\mathbf{X}_i)$ is the estimated "out-of-bag" correct classification probability for the IC observation $\mathbf{X}_i$ that is obtained from the RF classification. As discussed in Deng et al. (2012), there could be several alternative charting statistics, such as the estimated "out-of-bag" correct classification rate for observations in $\mathcal{X}_{AC_n}$. But, they found that the chart based on the above P_n, denoted as RTC, had some favorable properties.

The control limit of the chart RTC can be determined by the following bootstrap procedure suggested by Deng et al. (2012). First, we draw with replacement a sample from the dataset $\mathcal{X}_{IC_1}$. Then, the chart RTC with control limit h is applied to the bootstrap sample to obtain a RL value. This bootstrap re-sampling procedure is repeated $B = 1,000$ times, and the average of the B RL values is used to approximate the ARL_0 value for the given h. Finally, h can be empirically selected so that assumed ARL_0 is reached. Finally, h be searched by a numerical algorithm so that the assumed ARL_0 value is reached.

2.1.3 Distance based control chart using SVM

The charting statistic of the RTC chart discussed above actually take discrete values, because the estimated "out-of-bag" correct classification probabilities $\{P_{OOB}(\mathbf{X}_i)\}$ are obtained from an ensemble of decision trees (Breiman 2001 and He, Jiang, and Deng 2018). As an alternative, He, Jiang, and Deng (2018) suggested a distance-based control chart under the framework of SVM, which is denoted as DSVM. The DSVM method uses the distance between the support vectors and the process observations in the dataset $\mathcal{X}_{AC_n}$ as a charting statistic. Unlike charting statistic P_n of the RTC chart, this distance-based charting statistic is a continuous variable. Because the distance from a sample of process observations to the boundary surface

defined by the support vectors can be either positive or negative, He, Jiang, and Deng suggested transforming the distance using the standard logistic function

$$g(a) = \frac{1}{1 + \exp(-a)}.$$

Then, the following average value of the transformed distances from individual observations in $\mathcal{X}_{AC_n}$ to the boundary surface can be defined to be the charting statistic:

$$M_n = \frac{\sum g(d(\mathbf{X}_i))I(\mathbf{X}_i \in \mathcal{X}_{AC_n})}{|\mathcal{X}_{AC_n}|},$$

where $d(\mathbf{X}_i)$ is the distance from the observation $\mathbf{X}_i$ to decision boundary determined by the SVM algorithms at time n.

In the above DSVM chart, the kernel function and the penalty parameter need to be selected properly. He, Jiang, and Deng (2018) suggested using the following Gaussian radial basis function (RBF): for any $\mathbf{X}, \mathbf{X}' \in R^p$,

$$K(\mathbf{X}, \mathbf{X}') = \exp\left(\frac{\|\mathbf{X} - \mathbf{X}'\|^2}{\sigma^2}\right)$$

as the kernel function, where p is dimension of the process observations, and the parameter σ^2 was chosen to be larger than 2.8. They also suggested choosing the penalty parameter to be 1. The control limit of the chart DSVM can be determined by a bootstrap procedure, similar to the one described above for the RTC chart.

2.1.4 Control chart based on the KNN classification

Another approach to develop machine learning control charts is to use *one-class classification (OCC)* algorithms. Sun and Tsung (2003) developed a nonparametric control chart based on the so-called support vector data description (SVDD) approach (Tax and Duin 2004), described below. By SVDD, the boundary surface of an IC data can be defined so that the volume within the boundary surface is as small as possible while the Type-I error probability is controled within a given level of α. Then, the boundary surface is used as the decision rule for online process monitoring as follows: a new observation is claimed to be OC if it falls outside of the boundary surface, and IC otherwise. See Camci et al. (2008) for some modifications and generalizations. However, determination of this boundary surface is computationally intensive. To reduce the computation burden, Sukchotrat, Kim and Tsung (2009) suggested a control chart based on the KNN classification, denoted as KNN. In KNN, the average distance between a given observation $\mathbf{X}_i$ and its k nearest neighboring observations in the IC dataset is first calculated as follows:

$$K_i^2 = \frac{\sum_{j=1}^{k} \|\mathbf{X}_i - NN_j(\mathbf{X}_i)\|}{k},$$

where $NN_j(\mathbf{X}_i)$ is the j^{th} nearest neighboring observation of $\mathbf{X}_i$ in the IC dataset, and $\|\cdot\|$ is the Euclidean distance. Then, the $(1-\alpha)$th quantile of all such distances of individual observations in the IC data can be computed. This quantile can be used as the decision rule for online process monitoring as follows. At the current time n, if the average distance from $\mathbf{X}_n$ to its k nearest neighboring observations (i.e., K_n^2) is less than the quantile, then $\mathbf{X}_n$ is claimed as IC. Otherwise, it is claimed as OC.

In the above KNN chart, the control limit (i.e., the $(1-\alpha)$th quantile of $\{K_i^2\}$ of individual observations in the IC data) can be refined by the following bootstrap procedure suggested by Sukchotrat et al (2009). First, a total of $B = 1,000$ bootstrap samples are obtained from the IC dataset by the simple random sampling procedure with replacement. Then, the $(1-\alpha)$th quantile of $\{K_i^2\}$ of individual observations in each bootstrap sample can be computed. Then, the final control limit is chosen to be the mean of the B such quantiles. The KNN chart assumes that process observations at different time points are independent. Thus, its ARL_0 value equals $1/\alpha$.

2.2 Sequential Data De-Correlation

In this subsection, the sequential data de-correlation procedure for multivariate serially correlated data is described in detail. It is assumed that the IC process mean is $\boldsymbol{\mu}$ and the serial data correlation is stationary with the covariances $\boldsymbol{\gamma}(s) = \text{Cov}(\mathbf{X}_i, \mathbf{X}_{i+s})$, for any i and s, that depend only on s.

For the first observation $\mathbf{X}_1$, its covariance matrix is $\boldsymbol{\gamma}(0)$. Then, its standardized vector can be defined to be

$$\mathbf{X}_1^* = \boldsymbol{\gamma}(0)^{-1/2}(\mathbf{X}_1 - \boldsymbol{\mu}).$$

After the second observation $\mathbf{X}_2$ is collected, let us consider the long vector $(\mathbf{X}_1', \mathbf{X}_2')'$. Its covariance matrix can be written as $\Sigma_{2,2} = \begin{pmatrix} \boldsymbol{\gamma}(0) & \boldsymbol{\sigma}_1 \\ \boldsymbol{\sigma}_1' & \boldsymbol{\gamma}(0) \end{pmatrix}$, where $\boldsymbol{\sigma}_1 = \boldsymbol{\gamma}(1)$. The Cholesky decomposition of $\Sigma_{2,2}$ is given by $\Phi_2 \Sigma_{2,2} \Phi_2' = \mathbf{D}_2$, where $\Phi_2 = \begin{pmatrix} \mathbf{I}_p & \mathbf{0} \\ -\boldsymbol{\sigma}_1' \boldsymbol{\gamma}(0)^{-1} & \mathbf{I}_p \end{pmatrix}$, and $\mathbf{D}_2 = \begin{pmatrix} \mathbf{d}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{d}_2 \end{pmatrix} = diag(\mathbf{d}_1, \mathbf{d}_2)$, $\mathbf{d}_1 = \boldsymbol{\gamma}(0)$, and $\mathbf{d}_2 = \boldsymbol{\gamma}(0) - \boldsymbol{\sigma}_1' \boldsymbol{\gamma}(0)^{-1} \boldsymbol{\sigma}_1$. Therefore, we have $\text{Cov}(\Phi_2 \mathbf{e}_2) = \mathbf{D}_2$, where $\mathbf{e}_2 = [(\mathbf{X}_1 - \boldsymbol{\mu})', (\mathbf{X}_2 - \boldsymbol{\mu})']'$. Since $\Phi_2 \mathbf{e}_2 = \begin{pmatrix} \mathbf{I}_p & \mathbf{0} \\ -\boldsymbol{\sigma}_1' \boldsymbol{\gamma}(0)^{-1} & \mathbf{I}_p \end{pmatrix} \begin{pmatrix} (\mathbf{X}_1 - \boldsymbol{\mu})' \\ (\mathbf{X}_2 - \boldsymbol{\mu})' \end{pmatrix} = (\epsilon_1', \epsilon_2')'$, where

$$\begin{aligned} \epsilon_1 &= \mathbf{X}_1 - \boldsymbol{\mu}, \\ \epsilon_2 &= -\boldsymbol{\sigma}_1' \Sigma_{1,1}^{-1} (\mathbf{X}_1 - \boldsymbol{\mu}) + (\mathbf{X}_2 - \boldsymbol{\mu}), \end{aligned}$$

ϵ_1 and ϵ_2 are uncorrelated. Therefore, the de-correlated and standardized vector of $\mathbf{X}_2$ can be defined to be

$$\mathbf{X}_2^* = \mathbf{d}_2^{-1/2} \epsilon_2 = \mathbf{d}_2^{-1/2} \left[-\boldsymbol{\sigma}_1' \Sigma_{1,1}^{-1} (\mathbf{X}_1 - \boldsymbol{\mu}) + (\mathbf{X}_2 - \boldsymbol{\mu}) \right].$$

It is obvious that $\mathbf{X}_1^*$ and $\mathbf{X}_2^*$ are uncorrelated, and both have the identity covariance matrix $\mathbf{I}_p$.

Similarly, for the third observation $\mathbf{X}_3$, which could be correlated with $\mathbf{X}_1$ and $\mathbf{X}_2$, consider the long vector $(\mathbf{X}_1', \mathbf{X}_2', \mathbf{X}_3')'$. Its covariance matrix can be written as $\Sigma_{3,3} = \begin{pmatrix} \Sigma_{2,2} & \boldsymbol{\sigma}_2 \\ \boldsymbol{\sigma}_2' & \gamma(0) \end{pmatrix}$, where $\boldsymbol{\sigma}_2 = ([\gamma(2)]', [\gamma(1)]')'$. If we define $\Phi_3 = \begin{pmatrix} \Phi_2 & \mathbf{0} \\ -\boldsymbol{\sigma}_2'\Sigma_{2,2}^{-1} & \mathbf{I}_p \end{pmatrix}$ and $\mathbf{D}_3 = \begin{pmatrix} \mathbf{d}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{d}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{d}_3 \end{pmatrix} = diag(\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3)$, where $\mathbf{d}_3 = \Sigma_{3,3} - \boldsymbol{\sigma}_2'\Sigma_{2,2}^{-1}\boldsymbol{\sigma}_2$, then we have $\Phi_3\Sigma_{3,3}\Phi_3' = \mathbf{D}_3$. This motivates us to consider $\Phi_3\mathbf{e}_3$, where $\mathbf{e}_3 = [(\mathbf{X}_3 - \boldsymbol{\mu})', (\mathbf{X}_1 - \boldsymbol{\mu})', (\mathbf{X}_2 - \boldsymbol{\mu})']'$. It can be checked that $\Phi_3\mathbf{e}_3 = (\epsilon_1', \epsilon_2', \epsilon_3')'$, where

$$\epsilon_3 = -\boldsymbol{\sigma}_2'\Sigma_{2,2}^{-1}\mathbf{e}_2 + (\mathbf{X}_3 - \boldsymbol{\mu}).$$

Since $\mathrm{Cov}(\Phi_3\mathbf{e}_3) = \mathbf{D}_3$, $\mathbf{e}_3$ is uncorrelated with $\mathbf{e}_1$ and $\mathbf{e}_2$. Therefore, the de-correlated and standardized vector of $\mathbf{X}_3$ is defined to be

$$\mathbf{X}_3^* = \mathbf{d}_3^{-1/2}\epsilon_3 = \mathbf{d}_3^{-1/2}(-\boldsymbol{\sigma}_2'\Sigma_{2,2}^{-1}\mathbf{e}_2 + (\mathbf{X}_3 - \boldsymbol{\mu})),$$

which is uncorrelated with $\mathbf{X}_1^*$ and $\mathbf{X}_2^*$ and has the identity covariance matrix $\mathbf{I}_p$.

Following the above procedure, we can define the de-correlated and standardized vectors sequentially after a new observation is collected. More specifically, at the j-th observation time, the covariance matrix of the long vector $(\mathbf{X}_1', \mathbf{X}_2', \ldots, \mathbf{X}_j')'$ can be written as $\Sigma_{j,j} = \begin{pmatrix} \Sigma_{j-1,j-1} & \boldsymbol{\sigma}_{j-1} \\ \boldsymbol{\sigma}_{j-1}' & \gamma(0) \end{pmatrix}$, where $\boldsymbol{\sigma}_{j-1} = ([\gamma(j-1)]', \ldots, [\gamma(2)]', [\gamma(1)]')'$. It can be checked that $\Phi_j\Sigma_{j,j}\Phi_j' = \mathbf{D}_j$, where $\Phi_j = \begin{pmatrix} \Phi_{j-1} & \mathbf{0} \\ -\boldsymbol{\sigma}_{j-1}'\Sigma_{j-1,j-1}^{-1} & \mathbf{I}_p \end{pmatrix}$, $\mathbf{D}_j = diag(\mathbf{d}_1, \mathbf{d}_2, \ldots \mathbf{d}_j)$, and $\mathbf{d}_j = \Sigma_{j,j} - \boldsymbol{\sigma}_{j-1}'\Sigma_{j-1,j-1}^{-1}\boldsymbol{\sigma}_{j-1}$. Therefore, if we define

$$\epsilon_j = -\boldsymbol{\sigma}_{j-1}'\Sigma_{j-1,j-1}^{-1}\mathbf{e}_{j-1} + (\mathbf{X}_j - \boldsymbol{\mu}),$$

then $\Phi_j\epsilon_j = (\mathbf{e}_1', \mathbf{e}_2', \ldots, \mathbf{e}_j')'$ and $\mathrm{Cov}(\Phi_j\epsilon_j) = D_j$, which implies that $\mathbf{e}_j$ is uncorrelated with $\{\mathbf{e}_1, \ldots, \mathbf{e}_{j-1}\}$. Therefore, the de-correlated and standardized vector of $\mathbf{X}_j$ is defined to be

$$\mathbf{X}_j^* = \mathbf{d}_j^{-1/2}\epsilon_j = \mathbf{d}_j^{-1/2}(-\boldsymbol{\sigma}_{j-1}'\Sigma_{j-1,j-1}^{-1}\mathbf{e}_{j-1} + (\mathbf{X}_j - \boldsymbol{\mu})),$$

which is uncorrelated with $\mathbf{X}_1^*, \ldots, \mathbf{X}_{j-1}^*$ and has the identity covariance matrix $\mathbf{I}_p$.

By the above sequential data de-correlation procedure, we can transform the originally correlated process observations to a sequence of uncorrelated and standardized observations, each of which has the mean $\mathbf{0}$ and the identity covariance matrix $\mathbf{I}_p$. In reality, the IC parameters $\boldsymbol{\mu}$ and $\{\boldsymbol{\gamma}(s)\}$ are usually unknown and should be estimated in advance. To this end, $\boldsymbol{\mu}$ and $\{\boldsymbol{\gamma}(s)\}$ can be estimated from the IC dataset $\mathcal{X}_{IC} = \{\mathbf{X}_{-m_0+1}, \mathbf{X}_{-m_0+2}, \ldots, \mathbf{X}_0\}$ as follows:

$$\widehat{\boldsymbol{\mu}} = \frac{1}{m_0} \sum_{i=-m_0+1}^{0} \mathbf{X}_i \tag{1}$$

$$\widehat{\gamma}(s) = \frac{1}{m_0 - s} \sum_{i=-m_0+1}^{-s} (\mathbf{X}_{i+s} - \widehat{\boldsymbol{\mu}})(\mathbf{X}_i - \widehat{\boldsymbol{\mu}})' .$$

2.3 Machine Learning Control Charts for Monitoring Serially Correlated Data

To monitor a serially correlated process with observations $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n, \ldots$, we can sequentially de-correlate these observations first by using the procedure described in the previous subsection and then apply the machine learning control charts described in Subsection 2.1. However, at the current time point n, to de-correlate $\mathbf{X}_n$ with all its previous observations $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_{n-1}$, will take much computing time, especially when n becomes large. To reduce the computing burden, Qiu et al. (2020) suggested that the observation $\mathbf{X}_n$ only need to be de-correlated with its previous b_{max} observations, based on the assumption that two process observations becomes uncorrelated if their observation times are more than b_{max} apart. This assumption basically says that the serial data correlation is short-ranged, which should be reasonable in many applications. Based on this assumption, a modified machine learning control chart for monitoring serially correlated data is summarized below.

- When $n = 1$, the de-correlated and standardized observation is defined to be $\widehat{\mathbf{X}}_1^* = \widehat{\gamma}(0)^{-1/2}(\mathbf{X}_1 - \widehat{\mu})$. Set an auxiliary parameter b to be 1, and then apply a machine learning control chart to $\widehat{\mathbf{X}}_1^*$.
- When $n > 1$, the estimated covariance matrix of $(\mathbf{X}'_{n-b}, \ldots, \mathbf{X}'_n)'$ is defined to be

$$\widehat{\Sigma}_{n,n} = \begin{pmatrix} \widehat{\gamma}(0) & \cdots & \widehat{\gamma}(b) \\ \vdots & \ddots & \vdots \\ \widehat{\gamma}(b) & \ldots & \widehat{\gamma}(0) \end{pmatrix} =: \begin{pmatrix} \widehat{\Sigma}_{n-1,n-1} & \widehat{\boldsymbol{\sigma}}_{n-1} \\ \widehat{\boldsymbol{\sigma}}'_{n-1} & \widehat{\gamma}(0) \end{pmatrix}.$$

  Then, the de-correlated and standardized observation at time n is defined to be

$$\widehat{\mathbf{X}}_n^* = \widehat{\mathbf{d}}_n^{-1/2} \left[-\widehat{\boldsymbol{\sigma}}'_{n-1} \widehat{\Sigma}_{n-1,n-1}^{-1} \widehat{\mathbf{e}}_{n-1} + (\mathbf{X}_n - \widehat{\boldsymbol{\mu}}) \right],$$

  where $\widehat{\mathbf{d}}_j = \widehat{\Sigma}_{j,j} - \widehat{\boldsymbol{\sigma}}'_{j-1} \widehat{\Sigma}_{n-1,n-1}^{-1} \widehat{\boldsymbol{\sigma}}_{j-1}$, and $\widehat{\mathbf{e}}_{n-1} = [(\mathbf{X}_{n-b} - \widehat{\boldsymbol{\mu}})', (\mathbf{X}_{n-b+1} - \widehat{\boldsymbol{\mu}})', \ldots, (\mathbf{X}_{n-1} - \widehat{\boldsymbol{\mu}})']'$. Apply a machine learning control chart to $\widehat{\mathbf{X}}_n^*$ to see whether a signal is triggered. If not, set $b = min(b + 1, b_{max})$ and $n = n + 1$, and monitor the process at the next time point.

3 Simulation Studies

In this section, we investigate the numerical performance of the four existing machine learning control charts AC, RTC, DSVM and KNN described in Subsection 2.1, in comparison with their modified versions AC-D, RTC-D, DSVM-D and KNN-D discussed in Subsection 2.3, where "-D" indicates that process observations are de-correlated before each method is used for process monitoring. In all simulation examples, the nominal ARL_0 values of all charts are fixed at 200. If there is no further specification, the parameter λ in the chart AC is chosen to be 0.2, as suggested in He et al. (2010), the moving window size w in the charts RTC and DSVM is chosen to be 10, as suggested in Deng et al. (2012) and He et al.(2018), and the number of nearest observations k in the chart KNN is chosen to be 30, as suggested in Sukchotrat et al. (2009). The number of quality variables is fixed at $p = 10$, the parameter b_{max} is chosen to be 20, and the IC sample size is fixed at $m_0 = 2,000$. The following five cases are considered:

- Case I: Process observations $\{\mathbf{X}_n, n \geq 1\}$ are i.i.d. with the IC distribution $N_{10}(\mathbf{0}, I_{10\times 10})$.
- Case II: Process observations $\{\mathbf{X}_n, n \geq 1\}$ are i.i.d. at different observation times, the 10 quality variables are independent of each other, and each of them has the IC distribution χ^2_3, where χ^2_3 denotes the chi-square distribution with the degrees of freedom being 3.
- Case III: Process observations $\mathbf{X}_n = (X_{n1}, X_{n2}, \ldots, X_{n10})'$ are generated as follows: for each i, X_{ni} follows the AR(1) model $X_{ni} = 0.1X_{n-1,i} + \epsilon_{ni}$, where $X_{01} = 0$ and $\{\epsilon_{n1}\}$ are i.i.d. random errors with the $N(0, 1)$ distribution. All 10 quality variables are assumed independent of each other.
- Case IV: Process observations $\mathbf{X}_n = (X_{n1}, X_{n2}, \ldots, X_{n10})'$ are generated as follows: for each i, X_{ni} follows the ARMA(3,1) model $X_{ni} = 0.8X_{n-1,i} - 0.5X_{n-2,i} + 0.4X_{n-3,i} + \epsilon_{ni} - 0.5\epsilon_{n-1,i}$, where $X_{1i} = X_{2i} = X_{3i} = 0$ and $\{\epsilon_{ni}\}$ are i.i.d. random errors with the distribution χ^2_3. All 10 quality variables are assumed independent of each other.
- Case V: Process observations follow the model $\mathbf{X}_n = A\mathbf{X}_{n-1} + \boldsymbol{\epsilon}_n$, where $\{\boldsymbol{\epsilon}_n\}$ are i.i.d. random errors with the $N_{10}(0, B)$ distribution, A is a diagonal matrix with the diagonal elements being 0.5, 0.4, 0.3, 0.2, 0.1, 0.1, 0.2, 0.3, 0.4, 0.5, and B is a 10×10 covariance matrix with all diagonal elements being 1 and all off-diagonal elements being 0.2.

In all five cases described above, each variable is standardized to have mean 0 and variance 1 before process monitoring. Obviously, Case I is the conventional case considered in the SPC literature with i.i.d. process observations and the standard normal IC process distribution. Case II also considers i.i.d. process observations, but the IC process distribution is skewed. Cases III and IV consider serially correlated process observations across different observation times; but the 10 quality variables are independent of each other. In Case V, process observations are serially correlated and different quality variables are correlated among themselves as well.

Evaluation of the IC performance. We first evaluate the IC performance of the related control charts. The control limits of the four control charts AC, RTC, DSVM and KNN are determined as discussed in Subsection 2.1. For each method, its actual ARL_0 value is computed as follows. First, an IC dataset of size $m_0 = 2,000$ is generated, and some IC parameters (e.g. $\boldsymbol{\mu}$ and $\boldsymbol{\gamma}(s)$) are estimated from the IC dataset. Then, each control chart is applied to a sequence of 2,000 IC process observations for online process monitoring, and the RL value is recorded. This simulation of online process monitoring is then repeated for 1,000 times, and the actual conditional ARL_0 value conditional on the given IC data is computed as the average of the 1,000 RL values. Finally, the previous two steps are repeated for 100 times. The average of the 100 actual conditional ARL_0 values is used as the approximated actual ARL_0 value of the related control chart, and the standard error of this approximated actual ARL_0 value can also be computed. For the four modified charts AC-D, RTC-D, DSVM-D and KNN-D, their actual ARL_0 values are computed in a same way, except that process observations are de-correlated before online monitoring.

Table 1 Actual ARL_0 values and their standard errors (in parentheses) of four machine learning control charts and their modified versions when their nominal ARL_0 values are fixed at 200.

Methods	Case I	Case II	Case III	Case IV	Case V
RF	189(3.98)	194(4.20)	105(1.42)	119(2.05)	106(1.33)
RF-D	193(3.22)	182(3.49)	188(3.61)	193(3.70)	194(3.37)
RTC	203(4.66)	207(5.23)	252(5.97)	133(3.02)	269(6.01)
RTC-D	194(3.68)	196(3.64)	201(4.00)	188(3.49)	190(3.96)
DSVM	213(5.20)	195(4.77)	263(6.99)	118(2.87)	277(6.34)
DSVM-D	193(4.33)	198(3.50)	193(4.16)	190(3.72)	188(3.73)
KNN	196(4.77)	188(3.88)	156(3.70)	266(6.02)	134(4.03)
KNN-D	191(4.20)	194(3.69)	194(4.01)	187(3.20)	190(3.18)

From Table 1, we can have the following results. First, the IC performance of the charts AC, RTC, DSVM and KNN all have a reasonably stable performance in Cases I and II when process observations are assumed to be i.i.d. at different observation times and different quality variables are assumed independent as well. Second, in Cases III-V when there is a serial data correlation across different observation times and data correlation among different quality variables, the IC performance of the charts AC, RTC, DSVM and KNN becomes unreliable since their actual ARL_0 values are substantially different from the nominal ARL_0 value of 200. Third, as a comparison, the IC performance of the four modified charts AC-D, RTC-D, DSVM-D and KNN-D is stable in all cases considered. Therefore, this example confirms that the IC performance of the machine learning control charts can be improved in a substantial way by using the suggested modification discussed in Subsection 2.3.

Evaluation of the OC performance. Next, we evaluate the OC performance of the related charts in the five cases discussed above. In each case, a shift is assumed to occur at the beginning of online process monitoring with the size 0.25, 0.5, 0.75 and 1.0 in each quality variable. Other setups are the same as those in Table 1. To

make the comparison among different charts fair, the control limits of the charts have been adjusted properly so that their actual ARL_0 values all equal to the nominal level of 200. The results of the computed ARL_1 values of these charts in Cases I-V are presented in Figure 1.

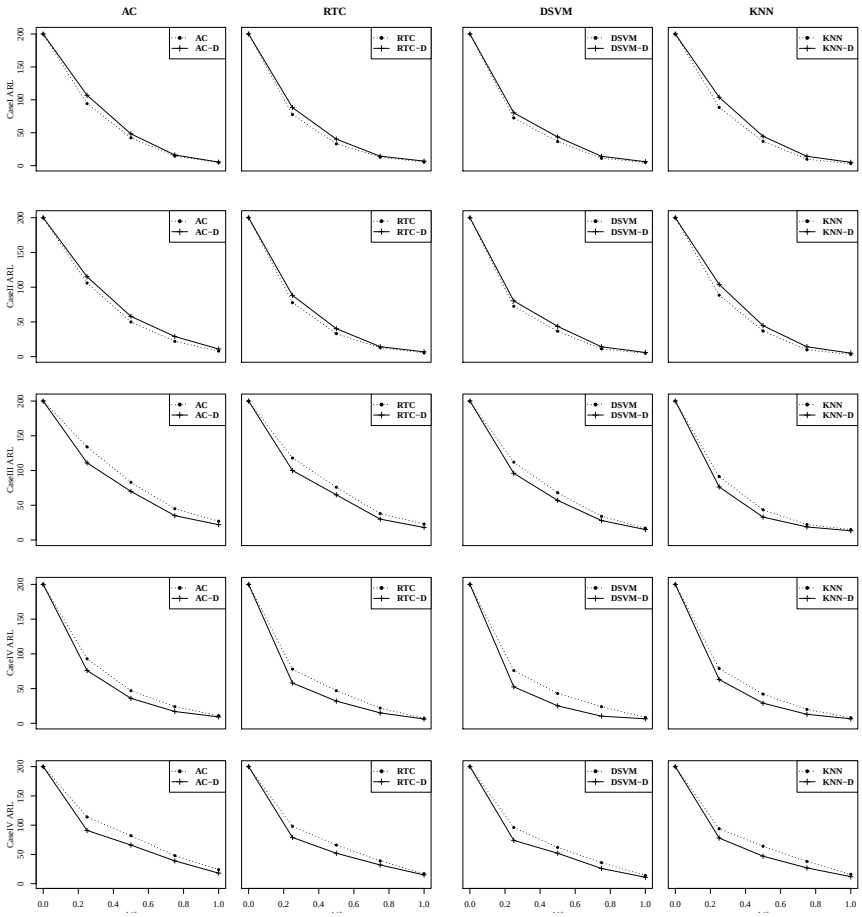

Fig. 1 Computed ARL_1 values of the original and modified versions of the four control charts AC, RTC, DSVM and KNN when their nominal ARL_0 values are fixed at 200, the parameters of the charts are chosen as in the example of Table 1, all quality variables have the same shift, and the shift size changes among 0.25, 0.5, 0.75 and 1.0.

From the Figure 1, it can be seen that the modified versions of the four control charts all have a better OC performance in Cases III-V when the serial data correlation exists. In Cases I and II when process observations are independent at different observation times, the OC performance of the modified versions of the four charts have a slightly worse performance than the original versions of the related charts.

The main reason for the latter conclusion is due to the "masking effect" of data de-correlation, as discussed in You and Qiu (2019). Remember that the de-correlated process observations are linear combinations of the original process observations. Therefore, a shift in the original data would be attenuated during data de-correlation, and consequently the related control charts would be less effective in cases when serial data correlation does not exist.

4 A Real-Data Application

In this section, a dataset from a semiconductor manufacturing process is used to demonstrate the application of the modified machine learning control charts discussed in the previous sections. The dataset is available in the UC Irvine Machine Learning Repository (http://archive.ics.uci.edu /ml/datasets/SECOM). It has a total of 590 quality variables and 1,567 observations of these variables. A total of 600 observations of five specific quality variables are selected here. The original data are shown in Figure 2. From the figure, it seems that the first 500 observations are quite stable, and thus they are used as the IC data. The remaining 100 observations are used for online process monitoring. In Figure 2, the training and testing datasets are separated by the dashed vertical lines.

For the IC data, we first check for existence of serial data correlation. To this end, the p-values of the Durbin-Watson test for the five quality variables are 1.789×10^{-3}, 4.727×10^{-1}, 4.760×10^{-4}, 1.412×10^{-4}, and 9.744×10^{-2}. Thus, there is a significant autocorrelation for the first, third and fourth quality variables. The Augmented Dickey-Fuller (ADF) test for stationality of the autocorrelation gives p-values that are < 0.01 for all quality variables. This result suggests that the stationary assumption is valid in this data. Therefore, the IC data have a significant stationary serial data correlation in this example, and the modification for the machine learning charts discussed in Sections 2 and 3 should be helpful.

Next, we apply the four modified control charts AC-D, RTC-D, DSVM-D and KNN-D to this data for online process monitoring starting from the 501^{st} observation time. In all control charts, the nominal ARL_0 values is fixed at 200, and their control limits are computed in the same way as that in the simulation study of Section 3. All four control charts are shown in Figure 3. From the plots in the figure, the charts AC-D, RTC-D, DSVM-D and KNN-D give their first signals at the 539^{th}, 529^{th}, 525^{th}, and 534^{th} observation times, respectively. In order to determine whether these signals are false alarms or not, the change-point detection approach based on the generalized maximum likelihood estimation (cf., Qiu 2014, Section 7.5) is applied to the test data (i.e., the data between the 501^{st} and 600^{th} observation times). The detected change-point position is at 517. The Hotelling's T^2 test for checking whether the mean difference between the two groups of data with the observation times in [501,516] and [517,600] is significantly different from $\mathbf{0}$ gives the p-value of 4.426×10^{-3}. Thus, there indeed is a significant mean shift at the time point 517. In this example,

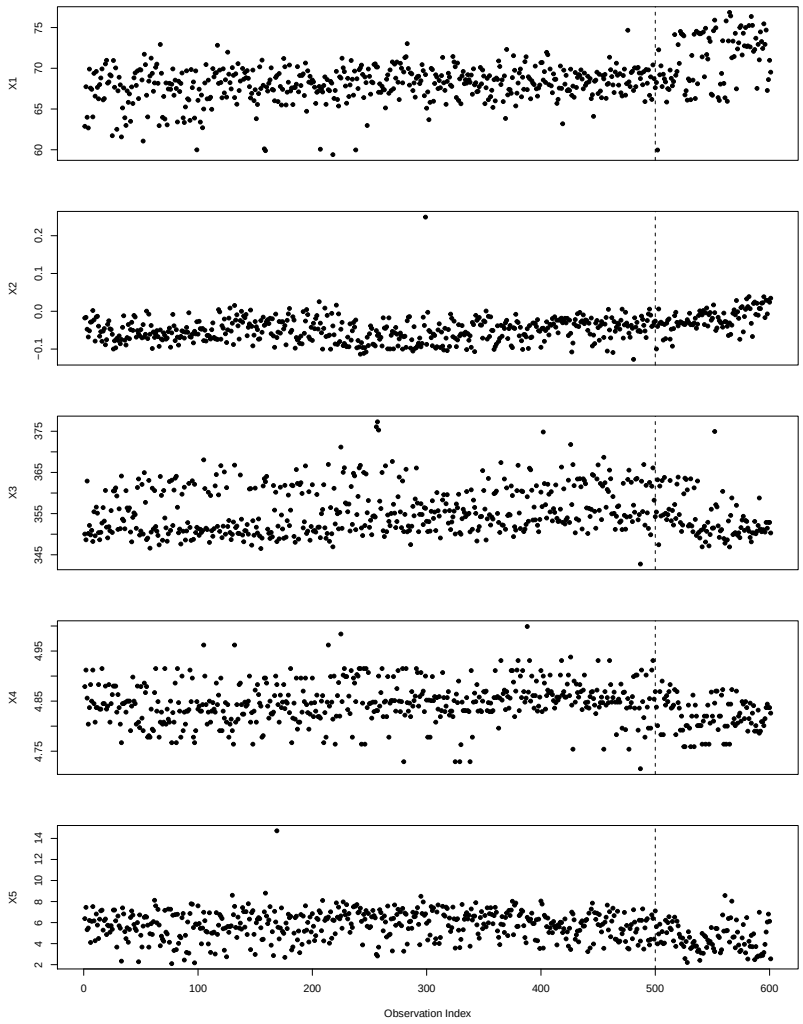

Fig. 2 Original observations of the five quality variables of a semiconductor manufacturing data. The vertical dashed line in each plot separates the IC data from the data for online process monitoring.

it seems that all four charts can detect the shift and the chart DSVM-D can give the earliest signal among them.

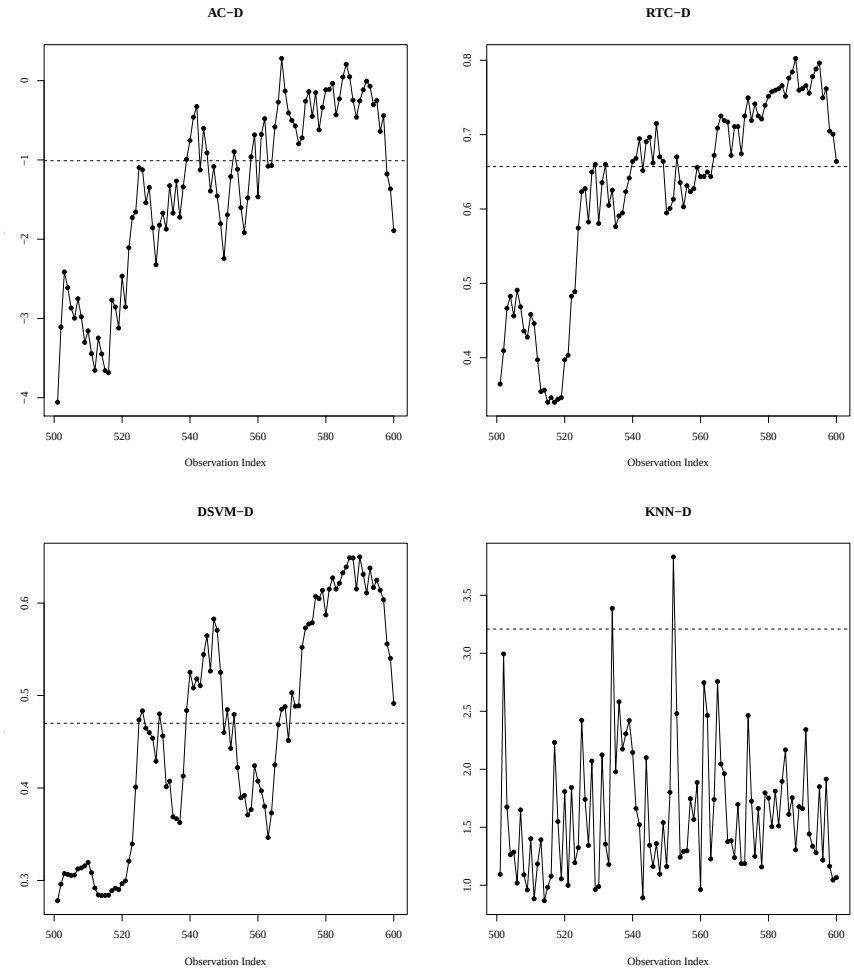

Fig. 3 Control charts AC-D, RTC-D, DSVM-D, and KNN-D for online monitoring of a semiconductor manufacturing data. The horizontal dashed line in each plot denotes the control limit of the related control chart.

5 Concluding Remarks

Recently, several multivariate nonparametric control charts based on different machine learning algorithms have been proposed for online process monitoring. Most existing machine learning control charts are based on the assumption that the multivariate observations are independent of each other. These control charts have a reliable performance when the data independence assumption is valid. However, when the process data are serially correlated, they may not be able to provide a reliable process monitoring. In this paper, we have suggested a modification for these machine learning control charts, by which process observations are first de-correlated before they are used for monitoring serially correlated data. Numerical studies have shown that the modified control charts have a more reliable performance than the original charts in cases when the serial data correlation exists.

There are still some issues to address in the future research. For instance, the "masking effect" of data de-correlation could attenuate the shift information in the de-correlated data. One possible solution is to use the modified data de-correlation procedure discussed in You and Qiu (2017). By this approach, the process observation at the current time point is de-correlated only with a small number of previous process observations within the so-called "spring length" (cf., Chatterjee and Qiu 2009) of the current observation time. Another issue is related to the assumption of short-range stationary serial data correlation that has been used in the proposed modification procedure. In some applications, the serial data correlation could be long-range and non-stationary (cf., Beran 1992). Thus, the proposed modification could be ineffective for such applications.

References

[1] Aggarwal, C.C. (2018), *Neural Networks and Deep Learning*, New Yorker: Springer.

[2] Alwan, L.C., and Roberts, H.V. (1995), "The problem of misplaced control limits," *Journal of the Royal Statistical Society (Series C)*, **44**, 269–278.

[3] Apley D.W., and Tsung, F. (2002), "The autoregressive T^2 chart for monitoring univariate autocorrelated processes," *Journal of Quality Technology*, **34**, 80–96.

[4] Breiman, L. (2001), "Random forests," *Machine Learning*, **45**, 5–32.

[5] Brean, J. (1992), "Statistical Methods for Data with Long-Range Dependence," *Statistical Science*, **4**, 404–416.

[6] Camci, F., Chinnam, R.B., and Ellis, R.D. (2008), "Robust kernel distance multivariate control chart using support vector principles," *International Journal of Production Research*, **46**, 5075–5095.

[7] Capizzi, G., and Masarotto, G. (2008), "Practical design of generalized likelihood ratio control charts for autocorrelated data," *Technometrics*, **50**, 357–370.

[8] Carvalhoa, T.P., Soares, F., Vita, R., Francisco, R., Basto, J.P., Alcalá, S.G.S. (2019), "A systematic literature review of machine learning methods applied to predictive maintenance," *Computers & Industrial Engineering*, **137**, 106024.

[9] Chatterjee, S., and Qiu, P. (2009), "Distribution-free cumulative sum control charts using bootstrap-based control limits," *Annals of Applied Statistics*, **3**, 349–369.

[10] Deng, H., Runger, G., and Tuv, E. (2012), "System monitoring with real-time contrasts," *Journal of Quality Technology*, **44**, 9–27.

[11] Göb, R. (2006), "Data mining and statistical control - a review and some links," In *Frontiers in Statistical Quality Control, vol. 8* (editted by H.J. Lenz, and P.T. Wilrich), Heidelberg: Physica-Verlag, 285–308.

[12] Hastie, T., Tibshirani, R., and Friedman, J. (2001), *The Elements of Statistical Learning - Data Mining, Inference, and Prediction,* Berlin: Springer-Verlag.

[13] He, S., Jiang, W., and Deng, H. (2018), "A distance-based control chart for monitoring multivariate processes using support vector machines," *Annals of Operations Research*, **263**, 191–207.

[14] Hu, J., Runger, G., and Tuv, E. (2007), "Tuned artificial contrasts to detect signals," *International Journal of Production Research*, **45**, 5527–5534.

[15] Hwang, W., Runger, G., and Tuv, E. (2007), "Multivariate statistical process control with artificial contrasts," *IIE Transactions*, **2**, 659–669.

[16] Knoth, S., and Schmid, W. (2004), "Control charts for time series: A review," In *Frontiers in Statistical Quality Control, vol. 7* (editted by H.J. Lenz, and P.T. Wilrich), Heidelberg: Physica-Verlag, 210–236.

[17] Lee, H. C., and Apley, D. W. (2011), "Improved design of robust exponentially weighted moving average control charts for autocorrelated processes," *Quality and Reliability Engineering International*, **27**, 337–352.

[18] Li, W., and Qiu, P. (2020), "A general charting scheme for monitoring serially correlated data with short-memory dependence and nonparametric distributions," *IISE Transactions*, **52**, 61–74.

[19] Li, W., Xiang, D., Tsung, F., and Pu, X. (2020), "A diagnostic procedure for high-dimensional data streams via missed discovery rate control," *Technometrics*, **62**, 84–100.

[20] Li, Zhang, Tsung and Mei(2020), "Nonparametric monitoring of multivariate data via KNN learning, " *International Journal of Production Research*, DOI: 10.1080/00207543.2020.1812750.

[21] Prajapati, D.R., and Singh, S. (2012), "Control charts for monitoring the autocorrelated process parameters: a literature review," *International Journal of Productivity and Quality Management*, **10**, 207–249.

[22] Psarakis, S., and Papaleonida, G.E.A. (2007), "SPC procedures for monitoring autocorrelated processes," *Quality Technology and Quantitative Management*, **4**, 501–540.

[23] Qiu, P. (2014), *Introduction to Statistical Process Control*, Boca Raton, FL: Chapman Hall/CRC.

[24] Qiu, P., Li, W., and Li, J. (2020), "A new process control chart for monitoring short-range serially correlated data," *Technometrics*, **62**, 71–83.

[25] Runger, G.C., and Willemain, T.R. (1995), "Model-based and model-free control of autocorrelated processes," *Journal of Quality Technology*, **27**, 283–292.

[26] Sukchotrat, T. , Kim, S. B. , Tsui, K.-L., and Chen, V. C. P. (2011), "Integration of classification algorithms and control chart techniques for monitoring multivariate processes," *Journal of Statistical Computation and Simulation*, **81**, 1897-1911.

[27] Sukchotrat, T., Kim, S. B., and Tsung, F. (2010), "One-class classification-based control charts for multivariate process monitoring," *IIE Transactions*, **42**, 107–120.

[28] Sun, R., and Tsung, F. (2003), "A kernel-distance-based multivariate control chart using support vector methods," *International Journal of Production Research*, **41**, 2975–2989.

[29] Tax, D.M., and Duin, R.P.W. (2004), "Support vector data description," *Machine Learning*, **54**, 45–66.

[30] Tuv, E., and Runger, G. (2003), "Learning patterns through artificial contrasts with application to process control," *Transactions on Information and Communications Technologies*, **29**, 63–72.

[31] Weiß, C.H. (2015), "SPC methods for time-dependent processes of counts - a literature review," *Cogent Mathematics*, **2**, 1111116.

[32] You, L., and Qiu, P. (2019), "Fast computing for dynamic screening systems when analyzing correlated data," *Journal of Statistical Computation and Simulation*, **89**, 379–394.

[33] Xue, L., and Qiu, P. (2020), "A nonparametric CUSUM chart for monitoring multivariate serially correlated processes," *Journal of Quality Technology*, DOI: 10.1080/00224065.2020.1778430.

[34] Zhang, C., Tsung, F., and Zou, C. (2015), "A general framework for monitoring complex processes with both in-control and out-of-control information," *Computers & Industrial Engineering,* **85**, 157–168.