International Journal of Production Research,
Volume 46, Issue 18, 2008, pp 5075-5095

A robust kernel-distance multivariate control chart using support vector principles

F. Camci[†], R. B. Chinnam[*‡], and R. D. Ellis[‡]

[†]Impact Technologies, LLC, 200 Canal View Boulevard, Rochester, NY 14623

[‡]Department of Industrial & Manufacturing Engineering, Wayne State University, Detroit, MI 48201, USA

[*] Corresponding Author: Tel: +313-577-4846; Fax: +313-577-8833; E-mail: r_chinnam@wayne.edu

Abstract

It is important to monitor manufacturing processes in order to improve product quality and reduce production cost. Statistical Process Control (SPC) is the most commonly used method for process monitoring, in particular, making distinctions between variations attributed to normal process variability to those caused by 'special causes'. Most SPC and multivariate SPC (MSPC) methods are *parametric* in that they make assumptions about the distributional properties and auto-correlation structure of in-control process parameters, and if satisfied, are effective in managing false alarms/positives and false negatives. However, when processes do not satisfy these assumptions, the effectiveness of SPC methods is compromised. Several *non-parametric* control charts based on sequential ranks of data depth measures have been proposed in the literature, but their development and implementation have been rather slow in industrial process control. Several non-parametric control charts based on *machine learning* principles have also been proposed in the literature to overcome some of these limitations. However, unlike conventional SPC methods, these non-parametric methods require event data from each out-of-control process state for effective model building. This paper presents a new non-parametric multivariate control chart based on *kernel-distance* that overcomes these limitations by employing the notion of one-class classification based on support vector principles. The chart is non-parametric in that it makes no assumptions regarding the data probability density and only requires 'normal' or in-control data for effective representation of an in-control process. It does however make an explicit provision to incorporate any available data from out-of-control process states. Experimental evaluation on a variety of benchmarking datasets suggests that the proposed chart is effective for process monitoring.

Keywords: Control chart; Support vector machines; Kernel-distance

I. Introduction

In order to improve product quality and reduce production cost, it is necessary to detect equipment malfunctions, failures, or other special events as early as possible. For example, according to the survey conducted by (Nimmo 1995; Chen, Kruger *et al.* 2004), the US-based petrochemical industry could save up to $10 billion annually if abnormal process behavior could be detected, diagnosed and appropriately dealt with. By monitoring the performance of a process over time, statistical process control (SPC) attempts to distinguish process variation attributed to *common causes* from variation attributed to *special causes*, and hence, forms a basis for process monitoring and equipment malfunction detection (Martin, Morris *et al.* 1996). It is also the most commonly used tool to analyze and monitor processes (Eickelmann and Anant 2003).

Most SPC methods are *parametric* in that they make assumptions about the distributional properties and auto-correlation structure of in-control process parameters, and if satisfied, are very effective in managing false alarms/positives and false negatives. Among others, they offer the following statistical and practical advantages: 1) only data from an in-control state is necessary to initialize the control chart, 2) data from a relatively limited number of sub-groups (say 20 to 30) is adequate for accurate initialization, and 3) tradeoffs between false alarms/positives (Type-I error) and false negatives (Type-II errors) can be managed (by changing the width of the control limits and the sub-group sample size, respectively). While conventional SPC charts (such as the Shewhart type control charts (Shewhart 1926), the cumulative sum (CUSUM) control charts, and the exponentially weighted family control charts) are developed for univariate processes (Manabu, Shinji *et al.* 2004), multivariate statistical process control (MSPC) is employed to monitor processes that have correlated multi-variables (Montgomery 2001). One type of MSPC is multivariate charts extended from univariate SPC methods, including Hotelling's T^2 chart (Hotelling 1947), multivariate EWMA (Lowry, Woodall *et al.* 1992); (Runger and Prabhu 1996), (Testik and Borror 2004) and multivariate CUSUM charts (Ngai and

Zhang 2001); (Runger and Testik 2004). Another type of MSPC is based on latent variable projection, such as Principal Component Analysis (PCA) and Partial Least Squares (PLS) (MacGregor and Kourti 1995) (Raich and Cinar 1996) (Yoon and MacGregor 2004). An alternative approach to account for the dynamic aspects of the data in MSPC is to use MRA (multi-resolution analysis) (Bakshi 1998; Teppola and Minkkinen 2000). For a good review of MSPC charts, see (Lowry and Montgomery 1995). Among these, Hotelling's T^2 chart is widely used in practice (Sun and Tsung 2003).

The standard assumption behind majority of SPC and MSPC methods mentioned above is that the process variables follow a Gaussian distribution (Rose 1991), a questionable assumption in several industrial processes (Polansky 2001), and in particular, highly automated processes (Chinnam and Kolarik 1992). (Schilling and Nelson 1976) and many other researchers have investigated the effects of non-normality on the control limits and charting performance. To alleviate such effects, some distribution-free or *non-parametric* control charts have been proposed based on sequential ranks of data depth measures (Liu and Singh 1993; Aradhye, Bakshi *et al.* 2001; Stoumbos and Reynolds 2001; Chakraborti, Van der Laan *et al.* 2003; Messaoud, Weihs *et al.* 2004), but their development and implementation have been rather slow in industrial process control (Chakraborti, Van der Laan *et al.* 2001).

Several non-parametric control charts based on *machine learning* and pattern recognition principles have also been proposed in the literature. For example, (Cook and Chiu 1998) proposed *radial basis function* (RBF) networks to recognize shifts in correlated manufacturing processes, (Chinnam 2002) proposed *support vector machines* (SVMs) for recognizing shifts in correlated and other manufacturing processes, and (Smith 1994) and (Pugh 1991) considered *multi-layer perceptron* (MLP) networks for implementing Shewhart type control charts, all relaxing the Gaussian assumption. While these methods have shown success in relaxing the Gaussian assumption, the fundamental limitation with these and most other machine learning methods proposed in the literature for process control is that they cast the problem as that of

'classification' or 'pattern recognition', and hence, strictly require example data from all out-of-control states of interest. This is a critical limitation for obtaining example cases from all such states might be difficult, expensive, or even impossible. The second limitation is that they make no explicit provision to make tradeoffs between Type-I errors (false alarms) and Type-II errors (inability to detect shifts in process condition). Most of these machine learning methods also necessitate modeling and training for each specific failure type. A model that is developed for a specific type abnormal event (out-of-control state) cannot necessarily give good classification accuracy for another type of abnormal event.

This paper presents a new non-parametric *kernel-distance* control chart that employs the notion of *one-class classification* or *novelty detection* to overcome these limitations and adopts *support vector machine* (SVM) principles for doing so. There are several reasons for basing the proposed control chart on SVM principles: 1) They are a system for *efficiently* training linear learning machines in the kernel-induced feature space, 2) they successfully *control* the flexibility of kernel-induced feature space through generalization theory, and 3) they exploit existing *optimization* theory in doing so. An important feature of SVM systems is that, while enforcing the learning biases suggested by generalization theory, they also produce 'sparse' dual representations of the hypothesis, resulting in extremely efficient algorithms (Cristianini and Taylor 2000). This is due to the Karush-Kuhn-Tucker conditions (Kuhn and Tucker 1951) (Mangasarian 1994), which hold for the solution and play a crucial role in the practical implementation and analysis of these machines. Another important feature of the support vector approach is that due to Mercer's conditions on the kernels (Mercer 1909), the corresponding optimization problems are convex and hence have no local minima. This fact, and the reduced number of non-zero parameters, marks a clear distinction between these systems and other machine learning algorithms, such as neural networks (Cristianini and Taylor 2000). The end result is that the proposed kernel-distance control chart is non-parametric, only requires data from an in-control process state, makes provision to utilize any available data from out-of-control

states, and allows some tradeoff between Type-I and Type-II errors. The proposed kernel-distance control chart supports both univariate and multivariate processes and can monitor both process location and dispersion aspects through a single control chart.

A notable exception in the literature that also offers several similar features is another *kernel-distance* control chart (called *k*-chart) independently proposed by (Sun and Tsung 2003), also based on a one-class classifier, Support Vector Data Description (SVDD), developed by David Tax (Tax 2001; Tax and Duin 2004). However, a significant limitation with SVDD, and hence the *k*-chart by Sun and Tsung, is that it lacks any ability to make a good distinction between outliers and normal data within the training set. In addition, they make no provision to utilize any available data from out-of-control states, and lastly, they offer no structured method for making tradeoffs between Type-I and Type-II errors. This can also result in poor representation of the in-control process state if the data available for initializing the control chart is not pre-processed for elimination of outliers. Our proposed kernel-distance control chart overcomes these limitations by integrating the SVDD method with principles borrowed from Support Vector Representation and Discrimination Machine (SVRDM) by Yuan and Casasent (2003). Many of these positive attributes of our proposed kernel distance control chart are illustrated in Figure 1. Here, we particularly work with the same multivariate process control dataset employed by (Sun and Tsung 2003). Figure 1(a) illustrates a plot of 'normal' process data projected onto the space of the two most dominant principal components (identical to Figures 11-12 of (Sun and Tsung 2003)), along with a known 'outlier' point (from Figure 13 of (Sun and Tsung 2003)). If one were to use the data from Figure 1(a) to initialize the *k*-chart, there is no guarantee that the outlier point will be recognized to be an outlier (see for example Figure 9 of (Sun and Tsung 2003)). On the contrary, the kernel distance control chart proposed here gives us several options. Supposing that the outlier data point is not so labeled and presented for chart initialization, the proposed method recognizes the point to be an outlier (as shown in Figure 1(a)). If the outlier point is labeled prior to chart initialization as an 'outlier', the proposed method recognizes this label and puts it outside

the normal boundary (as shown in Figure 1(b)). If for some reason, one chooses to treat this outlier point as a 'normal' data point, the proposed method will accept this constraint and treats the point as normal and defines normal process boundary with the point as a boundary point (as is evident from Figure 1(c)). Thus, the proposed kernel distance control chart is very robust and can take advantage of any available data/knowledge regarding process faults. In addition, our proposed method also offers an effective heuristic for optimizing the kernel parameters, something missing from SVDD and the *k*-Chart from Sun and Tsung. For all these reasons, we label the proposed method the *robust* kernel-distance control chart or *rk*-Chart in short.

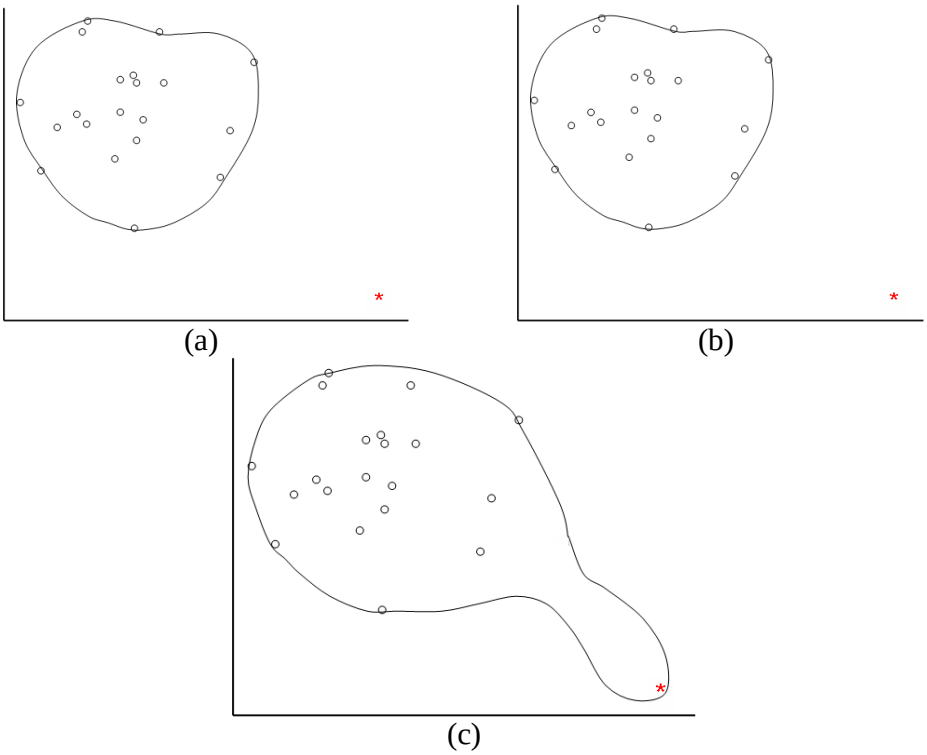


Figure 1. Flexibility of *rk*-Chart. In Panel a) the point well outside the normal data is "unlabeled" but is recognized and declared to be an outlier by *rk*-Chart. In panel (b) the point is "labeled" as an outlier, and the *rk*-Chart again declares it an outlier making use of the labeling. The results are identical to those without labeling. Finally, in panel c) the point is "labeled" as normal, and *rk*-Chart accepts this constraint and treats the point as normal and defines normal process boundary with the point as a boundary point. In all these cases, the *rk*-Chart was tuned with typical parameters (specified later).

The rest of this paper is organized as follows: Section II provides background information on support vector machines, the theory behind *rk*-Chart is presented in Section III, experimental results in Section IV, and concluding remarks in Section V.

II. Support Vector Machines

This section provides a brief background on support vector machines (SVMs) rooted in *Statistical Learning Theory*, a notion first introduced by (Vapnik 1998). We first explain the basics of SVMs for binary classification and then discuss how the technique can be extended to deal with the problem of one-class classification for developing *rk*-Chart.

II.1. Binary classification – Linear case

SVMs belong to the class of maximum margin classifiers. They perform pattern recognition between two classes by finding a decision surface that has maximum distance to the closest points in the training set, which are termed *support vectors*. We start with a training set of points $\mathbf{x}_i \in \Re^d$, $i = 1, 2..., N$ where each point $\mathbf{x}_i$ belongs to one of two classes identified by the label $y_i \in \{1, -1\}$ and d is the dimensionality of the points. Let the training set be denoted by $\Im = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$. Assuming *linearly separable* data, the goal of maximum margin classification is to separate the two classes by a hyperplane such that the distance to the support vectors is maximized. This is achieved by minimizing $\|w\|^2$ subject to the constraint $y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 \geq 0$, where $\mathbf{w}$ is normal to the hyper-plane. Figure 2 illustrates these concepts for the separable case. In order to provide for *non-separable* cases, the formulation is modified as follows:

$$\text{Minimize } \|w\|^2 + C\sum \xi_i \tag{1}$$

$$\text{Subject to: } y_i\left(\mathbf{x}_i \cdot \mathbf{w} + b\right) - 1 + \xi_i \geq 0 \tag{2}$$

This quadratic optimization problem can be solved efficiently using the following Lagrangian dual formulation:

$$\text{Maximize } \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \tag{3}$$

$$\text{Subject to: } 0 \le \alpha_i \le C \text{ and } \sum_i \alpha_i . y_i = 0 \tag{4}$$

where α_i denote the Lagrange multipliers. The Lagrangian formulation of the problem offers the advantage of having Lagrange multipliers in the constraints and training data in the form of inner products between data vectors (Müller, Mika *et al.* 2001). In the solution, non-zero α_i values represent support vectors that are on the separating hyper-plane (satisfying the equation $\mathbf{w} \cdot \mathbf{x} + b = 0$).

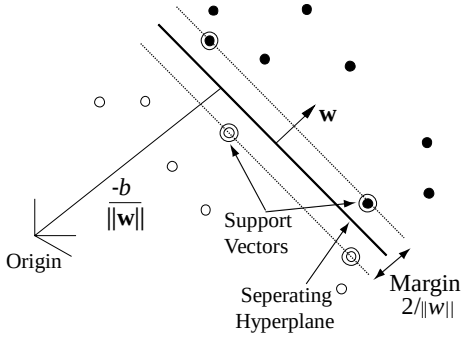


Figure 2: Separation of classes by hyper-plane.

II.2. Binary classification – Nonlinear case

In many cases, classes are not linearly separable. In order to learn non-linear relations with a linear machine, we need to select a set of non-linear features and to rewrite the data in the new representation. This is equivalent to applying a fixed non-linear mapping of the data to a feature space, in which the linear machine can be used. Thus, the non-linear separable case could be handled in two steps: first a fixed non-linear mapping transforms the data into a feature space F, and then a linear machine is used to classify them in the feature space. One important property of linear learning machine is that it can be expressed in a dual representation. That is, the hypothesis can be expressed as a linear combination of the training points so that the decision rule can be evaluated using just inner products between the test point and the training points. If one could

compute the inner product $\langle \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}) \rangle$ in feature space directly as a function of the original input points, it becomes possible to merge the two steps needed to build a non-linear learning machine. We call such a direct computation method a *kernel* function. A kernel is a function K, such that for all $\mathbf{x}, \mathbf{z} \in X$

$$K(\mathbf{x}, \mathbf{z}) = \langle \phi(\mathbf{x}) \cdot \phi(\mathbf{z}) \rangle \tag{5}$$

where ϕ is a mapping from X to an inner product feature space F.

By replacing the inner product with an appropriately chosen 'kernel' function, one can implicitly perform a non-linear mapping to a high dimensional feature space without increasing the number of tunable parameters, provided the kernel computes the inner product of the feature vectors corresponding to the two inputs. Thus, the use of kernels makes it possible to map the data implicitly into a feature space and to train a linear machine in such a space, potentially side-stepping the computational problems inherent in evaluating the feature map.

In practice, the approach taken is to define a kernel function directly, hence implicitly defining the feature space. In this way, we avoid the feature space not only in the computation of inner products, but also in the design of the learning machine itself. Mercer's theorem (Mercer 1909) provides a characterization of when a function $K(\mathbf{x}, \mathbf{z})$ is a kernel (i. e. what properties are necessary to ensure that it is a kernel for some feature space). An important family of kernel functions is the polynomial kernel:

$$K(\mathbf{x}, \mathbf{z}) = (1 + \mathbf{x} \cdot \mathbf{z})^d \tag{6}$$

where d is the degree of the polynomial. In this case the components of the mapping are all the possible monomials of input components up to the degree d. An even more popular kernel in the literature is the Gaussian (Tax and Duin 1999; Yuan and Casasent 2003):

$$K(\mathbf{x}, \mathbf{z}) = e^{-\|\mathbf{x}-\mathbf{z}\|^2/\sigma^2} \tag{7}$$

For a good discussion on making kernels, see (Cristianini and Taylor 2000). For more detailed information on the broader topic of support vector machines see (Müller, Mika et al. 2001).

III. Robust Kernel-Distance Control Chart (*rk*-Chart)

III.1. *rk*-Chart as a one-class classifier

As stated earlier, the proposed robust kernel-distance control chart (*rk*-Chart) employs the notion of one-class classification for process monitoring, and in doing so, models the boundary of process data from an in-control state and declares the process to be in control or out of control depending on where the new observation lies with respect to the boundary that exists in the feature space.

In the statistical process control literature, the two most important measures that are of particular interest in the context of process monitoring are measures of process *central tendency* and process *dispersion*. As originally proposed by W. Shewhart (Shewhart 1926), the sample *arithmetic mean* is the most employed measure for central tendency (among other measures such as mode and median). While the simplest statistical parameter that describes variability in observed data is the *sample range*, the *sample standard deviation* is a better estimate of variability for it considers every observation. Not unlike many multivariate statistical process control (MSPC) methods, the proposed *rk*-Chart jointly models these measures of central tendency as well as dispersion in a multi-dimensional space. While we recommend the sample mean and standard deviation as statistical measures for process monitoring (resulting in a 2-dimensional space), the proposed *rk*-Chart method is a general method and can incorporate any other type of location and dispersion measures, promising application specific measures, as well as other measures such as higher-order statistical moments (e. g., skewness and kurtosis). The only requirement for *rk*-Chart is that the vector of sample statistical measures is real, $\mathbf{x}_i \in \mathbb{R}^d$, where d denotes the dimensionality of the vector. However, note that the number of support

vectors (as well as the size of the training set) necessary for adequate representation of normal or in-control process state will increase as a function of d. In the case of multivariate processes, the feature space could simply be the process variable space or a transformation thereof (such as the space of dominant principal components). While it is also possible to introduce features from the co-variance matrix of the subgroup sample to monitor process dispersion, care should be exercised to manage d. Given that *rk*-Chart models the process data in two or higher dimensional spaces, in the case of univariate processes, it is necessary that the subgroup sample size be greater than one to facilitate extraction of at least two features (such as mean and standard deviation). While *rk*-Chart can in theory deal with both univariate and multivariate processes, and jointly monitors both location and dispersion measures, it does not necessarily have the ability to identify the type of process fault (such as a process location shift or process variance shift).

Thus, in developing the proposed control chart, we deviate from classical SVM designed for binary classification to representation of boundary from a single class (i. e. $\mathbf{x}_i \in \square^{d}$). *rk*-Chart is particularly inspired from Support Vector Data Description (SVDD) (Tax and Duin 1999) and Support Vector Representation Machine (SVRM) (Yuan and Casasent 2003) and gives the minimum volume closed spherical boundary around the in-control process data, represented by center $\mathbf{c}$ and radius r. Minimization of the volume is achieved by minimizing r^2, which represents *structural error* (Müller, Mika et al. 2001):

$$\text{Min} \quad r^2 \tag{8}$$

$$\text{Subject to: } \left\|\mathbf{x}_i - \mathbf{c}\right\|^2 \le r^2 \quad \forall i\,,\ \mathbf{x}_i : \ i^{th} \text{ data point} \tag{9}$$

The formulation above does not allow any data to fall outside of the sphere. In order to make provision within the model for potential outliers within the training set, a penalty cost function is introduced as follows (for data that lie outside of the sphere):

$$\text{Min} \ r^2 + C\sum_i \xi_i \tag{10}$$

$$\text{Subject to: } \left\|\mathbf{x}_i - \mathbf{c}\right\|^2 \le r^2 + \xi_i \,,\ \xi_i \ge 0 \quad \forall i \tag{11}$$

where C is the coefficient of penalty for each outlier (also referred to as the *regularization parameter*) and ξ_i is the distance between the i^{th} data point and the hyper-sphere. Once again, this is a quadratic optimization problem and can be solved efficiently by introducing Lagrange multipliers for constraints (Vapnik 1998):

$$\begin{aligned} L(r, \mathbf{c}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\gamma}) = r^2 + C\sum_i \xi_i - \\ \sum_i \alpha_i \left\{ r^2 + \xi_i - (\mathbf{x}_i \cdot \mathbf{x}_i - 2\mathbf{c} \cdot \mathbf{x}_i + \mathbf{c} \cdot \mathbf{c}) \right\} - \sum_i \gamma_i \xi_i \end{aligned} \tag{12}$$

where γ_i and α_i are Lagrange multipliers, $\gamma_i \ge 0$, $\alpha_i \ge 0$, and $\mathbf{x}_i \cdot \mathbf{x}_i$ is inner product of $\mathbf{x}_i$ and $\mathbf{x}_i$. Note that for each training data point $\mathbf{x}_i$, a corresponding α_i and γ_i are defined. L is minimized with respect to r, $\mathbf{c}$, and $\boldsymbol{\xi}$, and maximized with respect to $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$. Taking the derivatives of (12) with respect to r, $\mathbf{c}$, $\boldsymbol{\xi}$, and equating them to zero, we obtain the following constraints:

$$\mathbf{c} = \sum_i \alpha_i \mathbf{x}_i \tag{13}$$

$$C - \alpha_i - \gamma_i = 0 \quad \forall i \tag{14}$$

$$\sum_i \alpha_i = 1 \tag{15}$$

Given that $\gamma_i \ge 0$, $\alpha_i \ge 0$, constraint (14) can be rewritten as:

$$0 \le \alpha_i \le C \quad \forall i \tag{16}$$

The following quadratic programming equations can be obtained by substituting (13), (14), (15), and (16) in (12).

$$\text{Max } \sum_i \alpha_i (\mathbf{x}_i \cdot \mathbf{x}_i) - \sum_{i,j} \alpha_i \alpha_j (\mathbf{x}_i \cdot \mathbf{x}_j) \tag{17}$$

$$\text{Subject to: } 0 \le \alpha_i \le C \quad \forall i \,,\ \sum_i \alpha_i = 1 \tag{18}$$

Standard algorithms exist for solving this problem (Tax 2001). The above Lagrange formulation also allows further interpretation of the values of $\boldsymbol{\alpha}$. If necessary, the Lagrange

multipliers (α_i, γ_i) will take a value of zero in order to make the corresponding constraint term zero in (12). Thus, the *rk*-Chart formulation satisfies the Karush-Kuhn-Tucker (KKT) conditions for achieving a global optimal solution. Noting that $C = \alpha_i + \gamma_i$, if one of the multipliers becomes zero, the other takes on a value of C. When a data point $\mathbf{x}_i$ is inside the sphere, the corresponding α_i will be equal to zero. If it is outside of the sphere, i.e. $\xi_i > 0$, γ_i will be zero resulting in α_i to be C. When the data point is at the boundary, α_i and γ_i will be between zero and C to satisfy (15). The quadratic programming solution often yields a 'few' data points with a non-zero α_i value, or *support vectors*. What is of particular interest is that support vectors can effectively represent the data while remaining sparse. Let $\mathbf{S}^{SV} = \{x_i : \alpha_i \neq 0\}$ denote the set of support vectors.

In general, it is highly unlikely that a hyper-sphere can offer a good representation for the boundary of in-control process data in the 'original input space'. Hence, data ought to be transformed to a 'higher dimensional feature space' where it can be effectively represented using a hyper-sphere. Not unlike SVMs, *rk*-Chart also employs kernels to achieve this transformation without compromising computational complexity. Thus, the dot product in (17) is replaced by a Kernel function, leading us once again to the following quadratic programming problem:

$$\text{Max} \sum_i \alpha_i K(\mathbf{x}_i, \mathbf{x}_i) - \sum_{i,j} \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{19}$$

$$\text{Subject to } 0 \leq \alpha_i \leq C \quad \forall i \,, \sum_i \alpha_i = 1 \tag{20}$$

III.2. Gaussian kernel optimization within *rk*-Chart

The proposed robust kernel-distance control chart employs the one-class classification formulation from above along with a Gaussian kernel. The Gaussian kernel has been shown to particularly offer better performance over other kernels for one-class classification problems (see (Tax 2001) for more discussion on this) and hence the motivation for using it. The issue is

optimization of the scale parameter σ of (7). While σ could be specified by the user, *rk*-Chart employs the procedure outlined in Table 1 for choosing σ.

Table 1: Heuristic procedure for choosing σ for the Gaussian kernel.

Step 1:	Calculate the average 'nearest neighbor distance', denoted n_d, between all the data points in the dataset (i. e. $n_d = E_{\Im}(x_i^{nnd})$ where $x_i^{nnd} = \| \mathbf{x}_j - \mathbf{x}_i \| \leq \| \mathbf{x}_k - \mathbf{x}_i \| \ \forall i \neq k, j \neq i$ $E_{\Im}$ denotes average operator over the training set $\Im$).
Step 2:	For each data point $\mathbf{x}_i$:
2.1:	Construct a *local* robust kernel-distance 'boundary', denoted $rk\text{-}B_i^L$, utilizing the set $\mathbf{S}_i = \{\mathbf{x}_j : \| \mathbf{x}_j - \mathbf{x}_i \|^2 \leq (2n_d)^2\}$ (i. e. data within a sphere of radius $2n_d$).§ In building $rk\text{-}B_i^L$, set $\sigma_i^L = E_{\mathbf{S}_i} \| \mathbf{x}_i - E_{\mathbf{S}_i}(\mathbf{x}_i) \|$ for the Gaussian kernel (i. e. the average distance of data within $\mathbf{S}_i$ to the mean of $\mathbf{S}_i$). Let r_i^* denote the optimal radius of $rk\text{-}B_i^L$, i. e. the quadratic programming solution.
2.2:	If $\mathbf{x}_i \notin rk\text{-}B_i^L(\beta r_i^*)$, i. e. the data point is rejected by a scaled or *inner* $rk\text{-}B_i^L$ of radius βr_i^* where $0 < \beta < 1$ (a parameter pre-specified by the user, typically around 0.95), it is added into the *boundary list*, denoted $\mathbf{S}^{BL}$. It is also necessary that for any data point to be part of $\mathbf{S}^{BL}$, it cannot be accepted by any other *inner* $rk\text{-}B_j^L$. In addition, if $\mathbf{S}_i \equiv \mathbf{x}_i$, $\mathbf{x}_i$ is excluded from $\mathbf{S}^{BL}$ for it might be an outlier. Fig. 3 illustrates this procedure for determination of $\mathbf{S}^{BL}$.
Step 3:	The optimal *global* $rk\text{-}B^G$ (i. e. the *rk*-Chart boundary that represents the complete training set $\Im$) is then constructed using the following Gaussian kernel parameter: $\sigma^* = \underset{\min(x_i^{nnd}) \leq \sigma \leq \max(x_i^{nnd})}{\arg\max} \{\mathbf{S}^{BL} \approx \mathbf{S}^{SV}(\sigma)\}$.

§ Tax and Duin (Tax and Duin 1999) show that this setting of the local sphere radius to $2n_d$ results in good performance.

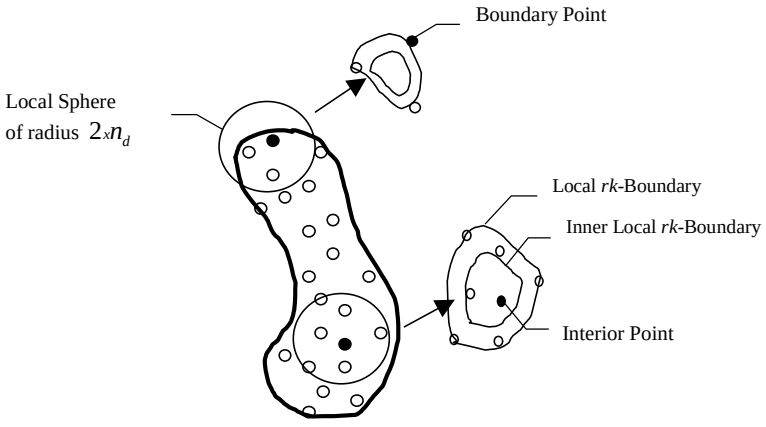


Figure 3: Determination of class boundary list: Data points on the boundary will be rejected by *inner local rk*-Boundaries.

In the context of a global *rk*-Chart, smaller σ values yield more representing points (i. e. support vectors) and a tighter hyper-sphere, whereas larger values give fewer support vectors and result in a bigger hyper-sphere. The goal is to identify a value for σ that results in good agreement between the 'support vector list' $\mathbf{S}^{SV}$ of the global *rk*-Chart boundary and the 'boundary list' $\mathbf{S}^{BL}$ resulting from local *rk*-Boundaries (see Table 1 for precise definitions of these terms, and Figure 3 for a depiction of the class boundary list):

$$\sigma^* = \underset{\min(x_i^{nnd}) \le \sigma \le \max(x_i^{nnd})}{\arg\max} \{\mathbf{S}^{BL} \approx \mathbf{S}^{SV}(\sigma)\} \tag{21}$$

In general, smaller σ values result in a global support vector list that is a superset of the boundary list, with some points that are not part of the boundary list. On the contrary, larger σ values result in a global support vector list that is a subset of the boundary list. In assessing this agreement, *rk*-Chart computes the fitness of a σ value by employing a two-part strategy: *effective representation* and *compactness*. Effective representation is achieved by ensuring that the global support vector list 'best matches' the boundary list. Compactness on the contrary emphasizes a smaller support vector list, which improves generalization. Compactness is managed through a user-defined parameter $0 \le \varsigma \le 1$. The higher the value of ς the more compact the support vector list and the higher the Type II error (i. e. inability to detect novel conditions), resulting in a larger hyper-sphere.

There is typically a σ value, denoted by σ_c, that results in near perfect agreement between the support vector list and the boundary list. As σ exceeds σ_c, the support vector list gets smaller. The actual value of σ employed in constructing the proposed global *rk*-Chart is:

$$\sigma = \sigma_c + \varsigma(\sigma_{max} - \sigma_c) \tag{22}$$

Figure 4 illustrates the influence of different compactness levels on the quality of representation, using an example dataset. The innermost *rk*-Chart boundary provides effective representation but with 20 support vectors all of which are in $\mathbf{S}^{BL}$ $(\varsigma = 0)$, whereas the outermost *rk*-Chart boundary

achieves compactness with just 2 support vectors $(\varsigma = 1)$. While the parameters $\sigma_{\min} = \min(x_i^{nnd})$, $\sigma_{\max} = \max(x_i^{nnd})$, σ_c are calculated empirically, the compactness parameter ς and the *rk*-Chart boundary scaling parameter β (employed for constructing the scaled inner *rk*-B_i^L) need to be pre-specified by the user or require repeated trials (typically set around 0.95).

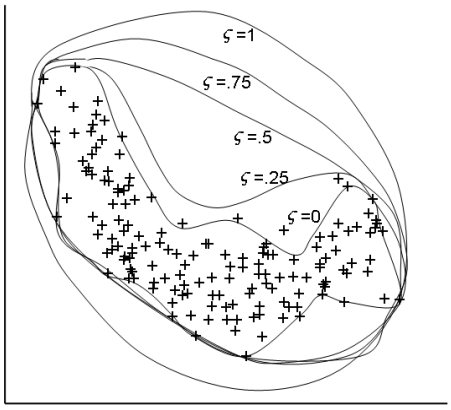


Figure 4: Influence of ς on *rk*-Chart representation.

Once the optimal σ value is calculated based on the desired degree of compactness, one can construct '*inner*' and '*outer*' boundary representations by correspondingly changing the radius of the *rk*-Chart hyper-sphere. Figure 5 illustrates this procedure for the same dataset from Figure 4. It is clear that as the radius is changed the overall geometric shape is maintained while the scale changes.

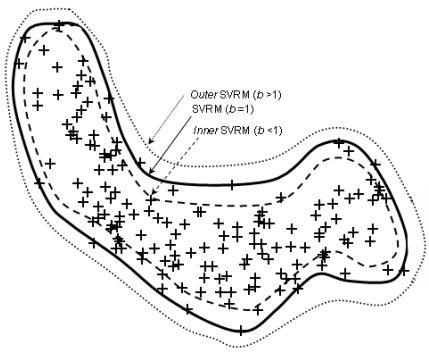


Figure 5: Influence of scaling *rk*-Chart hyper-sphere radius (using parameter β) on boundary representation.

III.3. Learning from abnormal data

There are occasions when observation samples from abnormal or out-of-control process states are available in the training set. Not exploiting this information can have a major detrimental effect on the performance of the control chart (in terms of Type-I and Type-II errors). The earlier *rk*-Chart formulation can be modified as follows to exploit any available examples from out-of-control states:

$$\text{Min } r^2 + C\sum_i \xi_i + C_o \sum_j \xi_j \tag{23}$$

$$\text{Subject to: } \left\|\mathbf{x}_i - \mathbf{c}\right\| \le r^2 + \xi_i \text{ and } \left\|\mathbf{x}_j - \mathbf{c}\right\| \ge r^2 - \xi_j \tag{24}$$

where C_o is the penalty value for an out-of-control state data point falling inside the representation boundary.

The Lagrange formulation of this new problem is as follows:

$$\begin{aligned} L(r, \mathbf{c}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\gamma}) &= r^2 + C\sum_i \xi_i + C_o \sum_j \xi_j - \sum_i \gamma_i \xi_i \\ &- \sum_j \gamma_j \xi_j - \sum_i \alpha_i \left\{ r^2 + \xi_i - (\mathbf{x}_i \cdot \mathbf{x}_i - 2\mathbf{c} \cdot \mathbf{x}_i + \mathbf{c} \cdot \mathbf{c}) \right\} \end{aligned} \tag{25}$$

To satisfy KKT condition, we once again take derivates of the cost function w.r.t. the parameters and equate them to zero, leading to the following equations:

$$\sum_i \alpha_i - \sum_i \alpha_i = 1 \tag{26}$$

$$\mathbf{c} = \sum_i \alpha_i \mathbf{x}_i - \sum_j \alpha_j \mathbf{x}_j \tag{27}$$

Substituting (26) and (27) in (25), leads us to following Lagrange formulation:

$$\begin{aligned} L(r, \mathbf{c}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\gamma}) = &\sum_{i \in I} \alpha_i (\mathbf{x}_i \cdot \mathbf{x}_i) - \sum_{i \in J} \alpha_i (\mathbf{x}_i \cdot \mathbf{x}_i) - \sum_{i,j \in I} \alpha_i \alpha_j (\mathbf{x}_i \cdot \mathbf{x}_j) \\ &+ \sum_{i \in I, j \in J} \alpha_i \alpha_j (\mathbf{x}_i \cdot \mathbf{x}_j) + \sum_{i \in J, j \in I} \alpha_i \alpha_j (\mathbf{x}_i \cdot \mathbf{x}_j) \\ &- \sum_{i,j \in J} \alpha_i \alpha_j (\mathbf{x}_i \cdot \mathbf{x}_j) \end{aligned} \tag{28}$$

where, I is the class of in-control state examples and J is the class of out-of-control state examples. This formulation can be simplified by labeling out-of-state classes as '-1' and in-control class as '+1'.

$$y_i = \begin{cases} 1 & x_i \in I \\ -1 & x_i \in J \end{cases} \tag{29}$$

$$\alpha_i^{'} = y_i \alpha_i \tag{30}$$

Substituting (29) and (30) in (28) results in the following:

$$\text{Max} \sum_{i \in \{I,J\}} \alpha_i^{'} (\mathbf{x}_i \cdot \mathbf{x}_i) - \sum_{i,j \in \{I,J\}} \alpha_i^{'} \alpha_j^{'} (\mathbf{x}_i \cdot \mathbf{x}_j) \tag{31}$$

$$0 \le \alpha_i \le C \tag{32}$$

$$0 \le \alpha_j \le C_o \tag{33}$$

$$\sum_i \alpha_i - \sum_i \alpha_i = 1 \tag{34}$$

As can be seen from equations (31-34), the formulation essentially remains the same with an additional constraint for the corresponding Lagrange multiplier value of the out-of-control state example.

IV. Experimental results

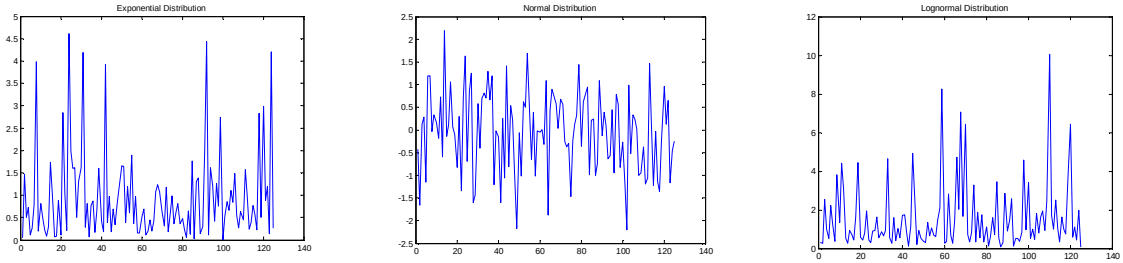
The application of *rk*-Chart will be discussed in two subsections: First, *rk*-Chart is evaluated using datasets that follow common probability distributions (i. e. normal, lognormal and exponential). We then evaluate *rk*-Chart using a benchmarking dataset (i. e. the Smith dataset) (Smith 1994) that has been used extensively for methodological development and evaluation in the process control literature (Chinnam 2002).

First, normal, lognormal and exponential distributions are employed to generate data. Parameters for the distributions and a sample of dataset are given in Table 2 and Figure 6, respectively. For a typical $\bar{X}$ chart, it is recommended to have at least 20-25 patterns (Woodall

and Montgomery 1999). In our experiment, 250 samples are generated for each distribution. Samples are grouped with size of 10 and two features, the mean and the standard deviation, are calculated for each sample group, resulting in only 25 two-dimensional data points.

Table 2: Parameters of distributions for evaluation experiments.

		Normal	Lognormal	Exponential
Normal Behavior (Nor)	μ	0	0	1
	σ	1	1	-
Small Mean Shift (SM)	μ	1	1	2
	σ	1	1	-
Large Mean Shift (LM)	μ	3	3	4
	σ	1	1	-
Small Variance Shift (SV)	μ	0	0	-
	σ	2	2	-
Large Variance Shift (LV)	μ	0	0	-
	σ	3	3	-



(a) $\mu = 1$ (b) $\mu = 0, \sigma = 1$ (c) $\mu = 0, \sigma = 1$

Figure 6: Example time Series. a) IID Exponential, b) IID Normal, c) IID Lognormal.

We trained *rk*-Chart in two ways: First, with only in-control data, and second, with in-control and limited out-of-control data as well. In the latter case, we used 10 abnormal patterns. As mentioned before, *rk*-Chart does not require out-of-control data, but having some helps to improve the accuracy of the method. Type-I and Type-II errors are defined as rejecting a true hypothesis and accepting a false hypothesis, respectively. In a statistical process control context, Type-I error refers to rejecting an in-control process as if it is out-of-control and Type-II error refers to accepting an out-of-control process as if it is in-control. The format of reporting Type-I and Type-II errors is given in the Table 3. The results of *rk*-Chart implementation on processes that follow normal, lognormal, and exponential distributions are summarized in Table 3.

Table 3: Type-I and Type-II errors defined.

Hypothesis Test		**Estimated**	
		In-control	Out-of-control
Actual	*In-control*	Correct	Type-I error
	Out-of-control	Type-II error	Correct

Table 4: Classification accuracy for: 1) (Left side of the table) Training only with 25 in-control patterns or sub-groups, each of which sub-group sample size of 10. 2) (Right side of the table) Training with 25 in-control and 10 out-of-control patterns, each with a sub-group sample size of 10; a) Normal b) Lognormal c) Exponential distributions; SM: Small Mean Shift, SV: Small Variance Shift, LM: Large mean shift, LV: Large variance shift.

1) Training only with in-control data

a.1			Estimated	
			In-control	Out-of-control
Actual	In-control		88.3%	11.7%
	Out-of-control	SM	7.4%	92.6%
		LM	0.0%	100.0%
		SV	6.4%	93.6%
		LV	0.2%	99.8%

2) Training with in-control and limited out-of-control data

a.2			Estimated	
			In-control	Out-of-control
Actual	In-control		86.6%	13.4%
	Out-of-control	SM	9.0%	91.0%
		LM	0.0%	100.0%
		SV	4.1%	95.9%
		LV	0.1%	99.9%

a) Normal Distribution

b.1			Estimated	
			In-control	Out-of-control
Actual	In-control		73.2%	26.8%
	Out-of-control	SM	4.5%	95.5%
		LM	0.0%	100.0%
		SV	14.8%	85.2%
		LV	4.1%	95.9%

b.2			Estimated	
			In-control	Out-of-control
Actual	In-control		81.3%	18.7%
	Out-of-control	SM	9.5%	90.5%
		LM	0.0%	100.0%
		SV	16.9%	83.1%
		LV	3.0%	97.0%

b) Lognormal Distribution

c.1			Estimated	
			In-control	Out-of-control
Actual	In-control		88.8%	11.2%
	Out-of-control	SM	23.9%	76.1%
		LM	0.7%	99.3%

c.2			Estimated	
			In-control	Out-of-control
Actual	In-control		80.8%	19.2%
	Out-of-control	SM	16.0%	84.0%
		LM	0.4%	99.6%

c) Exponential Distribution

As seen from Table 4, the non-parametric *rk*-Chart technique is able to detect out-of-control processes with no available out-of-control data with Type-I errors ranging from a highest of 26.8% to a low of 11.2%. Type-II errors ranged from 23.9% down to 0% in the case of no available out-of-control data. When limited training data from faulty process states are available,

Type-I errors improve to a high of 19.2% and a low of 13.4%. Type-II errors also improve with limited faulty process training data, to a high of 16.9% and a low of 0%.

We will also demonstrate the effectiveness of *rk*-Chart for detecting mean and variance shifts in using the benchmarking dataset generated by Smith (Smith 1994) and compare the results from a general support vector machine (SVM) method proposed by (Chinnam 2002) and a multi-layer-perceptron (MLP) neural network model (Smith 1994). The dataset has 300 samples from in-control state and out-of-control states of large mean shift (LM), small mean shift (SM), large variance shift (LV), and small variance shift (SV). The parameters are given in Table 5. The results will be reported in three different categories. In the first category, the results of *rk*-Chart will be compared with results of Shewhart chart, MLP and SVM. Note that MLP and SVM create a distinct model for each abnormality type (e.g. small mean shift, large variance shift). In the second category, two different *rk*-Chart results are reported: *rk*-Chart$^{(1)}$, which is trained with only in-control data and *rk*-Chart$^{(2)}$, which is trained with in-control and limited out-of-control data. Note that MLP and SVM cannot be implemented in case of absence of out-of-control samples. In the third category, the results of *rk*-Chart that is trained with very limited in-control (i.e. 25 samples) and out-of-control data (i.e. 10 samples) are reported.

Table 5: Parameters of testing datasets.

State Label	μ	σ
Normal Behavior (Nor)	0	1
Small Mean Shift (SM)	1	1
Large Mean Shift (LM)	3	1
Small Variance Shift (SV)	0	2
Large Variance Shift (LV)	0	3

Table 6: Classification Accuracy of *rk*-Chart versus MLP, Shewhart Charts and SVM charts using the benchmarking dataset from Smith (1994). *in: In-control data, out: Out-of-control data.

	rk-Chart		MLP	Shewhart Control Charts	SVM	
	Test	Train	Test	Test	Test	Train
Small Shift	91%	92%	72%	73%	93%	91%
Large Shift	96%	N/A	100%	100%	100%	99%

In the first category, classification accuracy is reported in methods such as MLP, Shewhart chart and SVM instead of Type-I and Type-II errors. Thus, the classification accuracy of *rk*-Chart is calculated in order to compare the results with these methods. The weighted average of Type-I and Type-II errors are calculated, considering the number of patterns used for in-control and out-of-control data. As seen from Table 6, *rk*-Chart performs better than MLP and Shewhart charts. SVM is only 1% to 4% better than *rk*-Chart. Even though SVM method gives better results than *rk*-Chart, there are two difficulties of implementing SVM and MLP in real world settings:

1. A distinct model is developed for each of the out-of-control state for SVM and MLP. There are four out-of-control states (SM, LM, SV, LV) resulting in four SVM and MLP models. Even though each model works well for developed out-of-control states, they cannot effectively work for other out-of-control states that are yet to have a model. In addition, SVM and MLP may be sensitive to an undefined out-of-control state. In contrast, *rk*-Chart characterizes the in-control state of the process and it has the ability to detect any undefined and unseen type of out-of-control state.
2. SVM and MLP are trained with 300 samples from in-control state and 300 samples from each modeled out-of-control state resulting in 4 models using a total of 2400 samples. On the other hand, *rk*-Chart uses only 300 samples from in-control state and 200 samples from out-of-control states.

Table 7: Type-I and Type-II errors for the benchmarking dataset from Smith (1994) using *rk*-Chart. In*: in-control state, Out*: Out-of-control state. *rk*-Chart$^{(1)}$: Training with 300 normal samples. *rk*-Chart$^{(2)}$: Training with 300 normal and 200 abnormal samples (100 small mean shift, 100 small variance shift)

			Estimated							
			rk-Chart$^{(1)}$				*rk*-Chart$^{(2)}$			
			In*	Out*	In*	Out*	In*	Out*	In*	Out*
			Train		Test		Train		Test	
Actual	In-cont.		100%	0%	84%	16%	88%	12%	90%	10%
	Out*	SM	N/A	N/A	5%	95%	1%	99%	7%	93%
		LM	N/A	N/A	0%	100%	N/A	N/A	0%	100%
		SV	N/A	N/A	15%	85%	4%	96%	9%	91%
		LV	N/A	N/A	15%	85%	N/A	N/A	8%	92%

In the second category, we will report *rk*-Chart results with no and limited out-of-control data. In the former case (i.e. *rk*-Chart$^{(1)}$), only 300 samples of in-control data are used for training, in the latter case (i.e. *rk*-Chart$^{(2)}$) 300 samples of in-control and 100 samples of small mean shift and 100 samples of small variance shift are employed for training. The Type-I and Type-II errors as given in Table 7 are promising with 16% Type-I error and highest 15% Type-II error in *rk*-Chart$^{(1)}$ and 12% Type-I error and highest 9% Type-II error with *rk*-Chart$^{(2)}$.

In the third category, *rk*-Chart is implemented with very limited in-control data. We implemented *rk*-Chart with 25 training patterns from the benchmarking Smith (Smith 1994) dataset and results are shown in Table 8.

Table 8: Process state classification for non-correlated Smith data with limited number of in-control and out-of-control samples (i. e. 25 in-control and 10 out-of-control patterns)

Data size of 25			*rk*-Chart	
			in*	out*
Actual	in		88%	12%
	out	SM	9%	91%
		LM	1%	99%
		SV	8%	92%
		LV	4%	96%

As seen from Table 8, in the worst of the cases tested, *rk*-Chart had the power to effectively detect out-of-control conditions 91% of the time (9% Type-II error), with false alarm rate of only 12% (Type-I error) even with limited data size.

As for parameter selection in initializing *rk*-Charts, when the control chart is initialized with just in-control data, two parameters are required: a penalty value for misclassification of in-control data (C) and compactness (ς). The recommended default parameter values are 1.0 and 0.2, for C and ς, respectively. An additional penalty parameter (C_0) is required for cases with any available out-of-control data for initialization. The recommended default parameter value is once again 1.0. All the experimental results reported here are based on these default settings. Extensive experimentation with these parameters suggests that the *rk*-Chart is relatively robust

with respect to parameter value selection and that the recommended default values work well in most cases. However, individual applications can call for different types of tradeoffs between Type-I and Type-II errors, which call for appropriate selection or tuning of *rk*-Chart parameters.

V. CONCLUSION

A new process control technique based on support vector machine principles, called robust kernel-distance control chart (*rk*-Chart), is proposed for both univariate and multivariate processes. *rk*-Chart has several advantages over the conventional SPC techniques and pattern recognition methods in the literature. *rk*-Chart does not make any assumption about the data distribution, which is a fundamental restriction for conventional SPC techniques. In addition, conventional SPC techniques cannot benefit from available out-of-control data, whereas *rk*-Chart can learn from out-of-control samples, where applicable. Pattern recognition methods used for process control in the literature based on other support vector machine (SVM) principles, radial-basis function (RBF) networks, and multi-layer-perceptron (MLP) neural networks require excessive amount of in-control data as well as out-of-control data. Furthermore, a distinct model for each type of out-of-control process needs to be created and trained using both in-control and out-of-control data with SVM, RBF and MLP. These models are not sensitive to out-of-control conditions other than those on which they are trained. In contrary, *rk*-Chart characterizes in-control-processes and requires only in-control data. However, it makes an explicit provision to accommodate any available out-of-control data. Thus, it is sensitive to all types of out-of-control processes. It is also shown that *rk*-Chart is able to give very reasonable results with limited in-control and out-of-control data when tested using a variety of datasets.

ACKNOWLEDGEMENTS

This research is partially funded by National Science Foundation under grant DMI-0300132.

REFERENCES

Aradhye, H. B., Bakshi, B. R., Strauss, R. A. and Davis, J. F. (2001). Multiscale statistical process control using wavelets: Theoretical analysis and properties. Columbus, OH, Ohio State University.

Bakshi, B., Multiscale PCA with application to multivariate statistical process monitoring. *Journal of the American Institute of Chemical Engineers*, 1998, **44**, 1596-1610.

Chakraborti, S., Van der Laan, P. and Bakir, S. T., Nonparametric control charts: An overview and some results. *Journal of Quality Technology*, 2001, **33**, 304-315.

Chakraborti, S., Van der Laan, P. and Van de Wiel, M., A class of distribution-free control charts. *Journal of the Royal Statistical Society, Series C*, 2004, 55(3), 443-462

Chen, Q., Kruger, U., Meronk, M. and Leung, A. Y. T., Synthetic of t2 and q statistics for process monitoring, control engineering practice. *Control Engineering Practice*, 2004, **12**, 745-755.

Chinnam, R. B., Support vector machines for recognizing shifts in correlated and other manufacturing processes. *International Journal of Production Research*, 2002, **40**, 4449-4466.

Chinnam, R. B. and Kolarik, W. J., Automation and the total quality paradigm, in *Proceedings of the 1st IERC*, 1992, Chicago, IL, IIE.

Cook, D. F. and Chiu, C., Using radial basis function neural networks to recognize shifts in correlated manufacturing process parameters. *IIE Transactions*, 1998, **30**, 227-234.

Cristianini, N. and Taylor, J. S., *An introduction to support vector machines and other kernel-based learning methods*. (Cambridge: Cambridge University Press).

Eickelmann, N. and Anant, A., Statistical process control: What you don't measure can hurt you! *Software, IEEE*, 2003, **20**, 49-51.

Hotelling, H. (1947). Multivariate quality control-illustrated by the air testing of sample bombsights. Techniques of statistical analysis. C. Eisenhart, M. W. Hastay and W. A. Wallis. New York, McGraw-Hill**:** 111-184.

Kuhn, H. and Tucker, A., Nonlinear programming, in *Proceedings of 2nd Berkeley Symposium on Mathematical Statistics and Probabilistics*, 1951, Berkely, CA, University of California Press, 481-492.

Liu, R. Y. and Singh, K., A quality index based on data depth and multivariate rank tests. *Journal of the American Statistical Association (JASA)*, 1993, **88**, 252-260.

Lowry, C. A. and Montgomery, D. C., A review of multivariate control charts. *IIE Transactions*, 1995, **27**, 800-810.

Lowry, C. A., Woodall, W. H., Champ, C. W. and Rigdon, S. E., A multivariate exponentially weighted moving average control chart. *Technometrics*, 1992, **34**, 46-53.

MacGregor, J. F. and Kourti, T., Statistical process control of multivariate processes. *Control Engineering Practice*, 1995, **3**, 403-414.

Manabu, K., Shinji, H., Iori, H. and Hiromu, O., Evolution of multivariate statistical process control: Application of independent component analysis and external analysis. *Computers & Chemical Engineering*, 2004, **28**, 1157-1166.

Mangasarian, O. L., *Nonlinear programming*. (Philadelphia, PA: Society for Industrial and Applied Mathematics).

Martin, E. B., Morris, A. J. and Zhang, J., Process performance monitoring using multivariate statistical process control. *IEE Proceedings*, 1996, **143**, 132-144.

Mercer, J., Functions of positive and negative type and their connection with the theory of integral equations. *Philosophical Transactions of the Royal Society of London, Series A*, 1909, **209**, 415-446.

Messaoud, A., Weihs, C. and Hering, F. (2004). A nonparametric multivariate control chart based on data depth. Dortmund, Germany, Department of Statistics, University of Dortmund.

Montgomery, D. C., *Introduction to statistical quality control*. (New York: Wiley).

Müller, K. R., Mika, S., Rätsch, G., Tsuda, K. and Schölkopf, B., An introduction to kernel-based learning algorithms. *IEEE Neural Networks*, 2001, **12**, 181-201.

Ngai, H. and Zhang, J., Multivariate cumulative sum control charts based on projection pursuit. *Statistica Sinica*, 2001, **11**, 747-766.

Nimmo, I., Adequately address abnormal situation operations. *Chemical Engineering Progress*, 1995, **91**, 36-45.

Polansky, A. M., A smooth nonparametric approach to multivariate process capability. *Technometrics*, 2001, **43**, 199-211.

Pugh, G. A., A comparison of neural networks to spc charts. *Computers and Industrial Engineering*, 1991, **21**, 253-255

Raich, A. and Cinar, A., Statistical process monitoring and disturbance diagnosis in multivariate continuous processes. *Journal of the American Institute of Chemical Engineers*, 1996, **42**, 995-1009.

Rose, K., Mathematics of success and failure. *Circuits and Devices, IEEE*, 1991, **7**, 26-30.

Runger, G. C. and Prabhu, S. S., A markov chain model for the multivariate exponentially weighted moving averages control chart. *Journal of the American Statistical Association (JASA)*, 1996, **91**, 1701-1706.

Runger, G. C. and Testik, M. C., Multivariate extensions to cumulative sum control charts. *Quality and Reliability Engineering International*, 2004, **20**, 587 - 606.

Schilling, E. G. and Nelson, P.R., The effect of non-normality on the control limits of $\bar{X}$ charts. *Journal of Quality Technology*, 1976, **8**, 183-187.

Shewhart, W. A., Quality control charts. *Bell System Technical Journal*, 1926, **22**, 593-603.

Smith, A. E., X-bar and r control chart interpretation using neural computing. *International Journal of Production Research*, 1994, **32**, 309-320.

Stoumbos, Z. G. and Reynolds, M. R., On shewhart-type nonparametric multivariate control charts based on data depth. *Frontiers in Statistical Quality Control*, 2001, **6**, 207-227.

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

Tax, D. M. (2001). One class classification. Delft, The Netherlands, Delft Technical University.

Tax, D. M. J. and Duin, R. P. W., Support vector domain description. *Pattern Recognition Letters*, 1999, **20**, 1191-1199.

Tax, D. M. J. and Duin, R. P. W., Support vector data description. *Machine Learning*, 2004, **54**, 45-66.

Teppola, P. and Minkkinen, P., Wavelet-pls regression models for both exploratory data analysis and process monitoring. *Journal of Chemometrics*, 2000, **14**, 383-399.

Testik, M. C. and Borror, C. M., Design strategies for the multivariate exponentially weighted moving average control chart. *Quality and Reliability Engineering International*, 2004, **20**, 571-577.

Vapnik, V., *Statistical learning theory*. (New York: Wiley).

Woodall, W. H. and Montgomery, D. C., Research issues and ideas in statistical process control. *Journal of Quality Technology*, 1999, **31**, 376-386.

Yoon, S. and MacGregor, J. F., Principal-component analysis of multiscale data for process monitoring and fault diagnosis. *Journal of the American Institute of Chemical Engineers*, 2004, **50**, 2891-2903.

Yuan, C. and Casasent, D., Support vector machines for class representation and discrimination, in *International Joint Conference on Neural Networks*, 2003, Portland, OR, 1610-1615.

https://dspace.org/

Robust kernel distance multivariate control chart using support vector principles

Camci, Fatih; Chinnam, R. B.; Ellis, R. D.

2012-01-24
Taylor & Francis
http://dx.doi.org/10.1080/00207540500543265

Downloaded from a DSpace repository. https://dspace.org/