Spatial Rank-based High-dimensional Monitoring Through Random Projection

Chen Zhang[1], Nan Chen[2], and Jianguo Wu*[3]

[1] Department of Industrial Engineering, Tsinghua University, Beijing, China, 100084

[2] Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore, 117576

*[3] Department of Industrial Engineering and Management, Peking University, Beijing, China, 100871

Abstract

High-dimensional process monitoring becomes increasingly important in many applications, where the joint distribution of process variables is usually unknown and not normal, and requires nonparametric methods for analysis and monitoring. However, when the process dimension is much larger than the reference sample size, most traditional nonparametric multivariate control charts fail due to the curse of dimensionality. Furthermore, when the process goes out of control, only a few (sparse) dimensions will be influenced, which increases the difficulty for both detection and diagnosis. To address these problems, this paper proposes a new nonparametric monitoring scheme for high-dimensional processes. This scheme first projects the high-dimensional process into several sub-processes using ensemble random projections for dimension reduction. Then for each sub-process a local nonparametric control chart is constructed based on the spatial rank test. Finally all the local charts are fused together for decision-making. Furthermore, after an out-of-control alarm is triggered, a diagnostic framework is proposed based on the square-root

*Corresponding author: j.wu@pku.edu.cn

LASSO algorithm. Numerical studies together with real-data examples demonstrate the efficacy and applicability of the proposed methodology.

Keywords: Change detection, Diagnosis, High-dimensional monitoring, Random projection, Spatial rank test, Square-root LASSO, Statistical process control

1 Introduction

Multivariate statistical process control (SPC) plays an important role for online monitoring of sequential data in many applications, such as fraudulent record detection (Tsung et al. 2007), multiple sensor network detection (Guerriero et al. 2009), health care monitoring (Spiegelhalter et al. 2012), image monitoring (Megahed et al. 2011), social network monitoring (McCulloh et al. 2012), *etc.*

One trend is, as the sensing technology advances and an increasing number of data streams are generated, the number of process variables to be monitored is growing tremendously to magnitude of hundreds or thousands. For example, in semiconductor industry, the manufacturing involves hundreds of stages, and at each stage, hundreds of sensors are allocated in the chamber with in total thousands of process variables (Lee et al. 2011) to be measured and analyzed. In service industry, thousands of customer variables from demographical, financial and geographic respects are collected by financial companies to detect frauds in credit cards or insurance claims (Jiang et al. 2012). In video monitoring, every video includes hundreds of frames, and each frame contains thousands of pixels to be monitored (Liu et al. 2015). These high dimensional data streams usually lead to many process parameters to be estimated, and consequently require a larger number of reference samples in Phase I analysis, which is often infeasible in practice. This poses tremendous challenges for conventional multivariate SPC methods, and triggers high demands for new SPC methods.

- Dr. Zhang is an Assistant Professor in the Department of Industrial Engineering, Tsinghua University. Her email address is zhangchen01@tsinghua.edu.cn.
- Dr. Chen is an Associate Professor in the Department of Industrial Systems Engineering and Management, National University of Singapore. His email is isecn@nus.edu.sg.
- Dr. Wu is an Assistant Professor in the Department of Industrial Engineering and Management, Peking University. His email is j.wu@pku.edu.cn. He is the corresponding author.

Suppose we aim at monitoring a p-dimensional process with m_0 independent and identically distributed ($i.i.d$) historical (reference) samples $\mathbf{X}_{-m_0+1}, \ldots, \mathbf{X}_0 \in \mathbb{R}^p$ on hand. The t^{th} future sample $\mathbf{X}_t$ is collected over time following the conventional change-point model, i.e.,

$$\mathbf{X}_t \overset{\text{i.i.d.}}{\sim} \begin{cases} F_0(\mathbf{X} - \boldsymbol{\mu}_0) & \text{for} \quad t = 1 \ldots, \tau, \\ F_1(\mathbf{X} - \boldsymbol{\mu}_1), & \text{for} \quad t = \tau + 1, \ldots, \end{cases} \tag{1}$$

where τ is the unknown change point. $F_0(\mathbf{X} - \boldsymbol{\mu}_0)$ and $F_1(\mathbf{X} - \boldsymbol{\mu}_1)$ are the in-control (IC) and out-of-control (OC) distributions, respectively, and are assumed to be continuous. In practice, they can be the same or different types of distributions, but their location parameters $\boldsymbol{\mu}_0$ and $\boldsymbol{\mu}_1$ are assumed to be unequal. This paper focuses on the "high-dimensional" case with $p \geq m_0$. In this case, the performance of the traditional T^2-type charts will degenerate (Bai and Saranadasa 1996) in the sense that the uncertainty in the estimated covariance matrix of $\mathbf{X}_t$, i.e., $\hat{\boldsymbol{\Sigma}}$, grows rapidly with p. Consequently, this uncertainty will distort the IC run-length distribution, making these methods unreliable (Champ et al. 2005). To address this drawback, a bunch of works have been done, and they can be majorly divided into two categories.

The first category is to modify $\hat{\boldsymbol{\Sigma}}$ to make it stable. Some pioneer works include replacing $\hat{\boldsymbol{\Sigma}}$ with either an identity matrix or a diagonal matrix (Chen et al. 2010; Mei 2010). These methods are equivalent to monitoring each dimension separately, with the assumption that the process variables are uncorrelated. Though in this way a good estimation of the marginal distribution of $\mathbf{X}_t$ can be achieved and a satisfactory IC performance can be guaranteed, these methods sacrifice the OC detection power due to their ignorance of the correlation structure. Chen et al. (2011) proposed to modify $\hat{\boldsymbol{\Sigma}}$ by adding a regularization term to ensure a well-posed estimation. However, this method can only alleviate the requirement of m_0 to some degree. It is still hard to be applied when p is large.

The second category avoids direct estimation of $\hat{\boldsymbol{\Sigma}}$ by reducing the dimension into a smaller set that is sufficient for process monitoring. The most seminal method in this category is to use principal component analysis (PCA) to extract features from the original process, and then use the PCA scores for monitoring (Ranger and Alt 1996). Later, Ding et al. (2006) proposed to use independent component analysis (ICA) instead for a better detection performance. Since ICA projects the original data into a subspace that maximizes the separation of any clustering structure existing in the data, this method can potentially better separate the OC samples from the IC ones than PCA. Both PCA and ICA can be regarded as unsupervised learning algorithms. When features of OC patterns are known, supervised learning algorithms can be used to further increase the detection power. Sukchotrat et al. (2011)

proposed to use linear discriminant analysis (LDA) to project the data into the subspace that best separated the IC and OC distributions. Ngai and Zhang (2001) proposed to use projection pursuit to find the most possible mean-shift direction. However, these methods share one common limitation that they aim at pursuing projections according to certain "interesting" criteria. While for a given data set with $p \geq m_0$, correct pursuance may be quite difficult. For example, for PCA, when p is much bigger than m_0, the first few estimated principal component directions based on $\hat{\mathbf{\Sigma}}$ are not consistent or converged to the appropriate subspaces, unless the first few eigenvalues of the population covariance matrix $\mathbf{\Sigma}$ are large enough compared with others (Jung et al. 2009). This criterion is not yet always satisfied in practice. Furthermore, one assumption of these methods is that the OC directions must lie in the low dimensions formulated by the extracted features. Otherwise, these methods will almost have no detection power. For example, monitoring of only the first several principal components will miss process shifts in other principal components.

Most of the methods mentioned above are based on a fundamental assumption that the process data follow multivariate normal distributions. However, in high-dimensional data streams, the underlying process distribution is usually not normal. When applied to non-normal data, these methods may have significantly diminished performance or even fail completely. This motivates the use of nonparametric charts. For cases with $p < m_0$, nonparametric methods have been widely used to achieve robust performance in non-normal distributions. These methods can be generally classified into two categories. The first one is to apply traditional nonparametric tests to SPC, such as the spatial sign test (Zou and Tsung 2011), the spatial rank test (Zou et al. 2012), and the directional rank test (Holland and Hawkins 2014). However, these methods are still built upon the T^2-type statistics and require sufficient reference samples to estimate the covariance matrix of the spatial signs or ranks. If the requirement is unsatisfied, they still suffer the same problem as the traditional T^2 charts. Qiu and Hawkins (2003) constructed a nonparametric CUSUM chart based on the anti-rank test. This chart can be applied in the case of $p \geq m_0$, since it does not consider correlations of anti-ranks of different variables. However, its performance is not very efficient, which will be shown in Section 3. Recently, Chen et al. (2016) constructed a monitoring scheme based on the Wilcoxon test to monitor each data stream separately. In this method, a data-driven method is proposed to set the control limit dynamically, which guarantees that the chart is able to achieve a constant prescribed false alarm rate at each time step. Later Zhang et al. (2016) extended Chen et al. (2016)'s work with a "divide-and-conquer" method. Its essence is to divide the original high-dimensional process into several low-dimensional processes (i.e., 2-dimensional processes in their paper). Then it constructs a nonparametric chart for every low-dimensional subprocess based on the goodness-of-fit test, and finally combines all these local charts together. This strategy has a trade-off between the feasibility of direct monitoring of a high-dimensional process and the poor power of totally ignoring the correlation structure. However, this method still faces with the problem that, usually for a high-dimensional process, the correlation structure is too complicated to be described by a pairwise approximation. Furthermore, how to select the pairs significantly influences the detection power and is still an open problem to be solved. In addition, the computational complexity to calculate the data-driven control limit sequence is quite heavy and grows exponentially with p, hindering its direct application to high-dimensional cases. The second type is based on machine learning or pattern recognition methods. The basic idea is to classify an online testing sample according to its distance from the IC distribution or a pre-specified OC distribution. So far several distance metrics have been adopted in SPC schemes, such as the Euclidean distance with the random forest algorithm (Deng et al. 2012), the k-nearest neighbor algorithm (Sukchotrat et al. 2011), and the kernel mean discrepancy (Huang et al. 2014). However, all the distance metrics considered in these methods do not use the data correlation structure at all, and consequently lead to their poor detection power when data are strongly correlated.

In summary, so far there is no satisfactory monitoring scheme for high-dimensional, correlated, and non-normal data streams with $p \geq m_0$. This paper targets at this research gap by developing a new nonparametric control chart. The core of our method is to decompose the high-dimensional process space into several subspaces using ensemble projections, and then construct nonparametric control charts for each subspace separately. Finally all the local results are combined for decision-making. However, as mentioned previously, due to the curse of dimensionality, traditional projection pursuit methods, such as PCA, LDA, or ICA cannot extract correct features using very limited reference samples. To solve this problem, enlightened by compressed sensing (Baraniuk 2007), we propose to use random projection. It can on the one hand guarantee that the geometric structure of original data is preserved after projection with high probability, and on the other hand does not require large reference sample size for implementation. However, unlike only using one projection in compressed sensing, here we propose to use ensemble random projections, and consequently can detect OC changes in different projected subspaces. Then for each subspace we construct a local monitoring scheme based on the spatial rank test. This test not only takes correlations of different variables into account, but also has robust detection power for general continuous multivariate distributions. Last but not the least, we propose a diagnostic framework to identify the potential OC directions using the square-root LASSO. The diagnostic framework also has satisfactory performance for various distributions.

The remainder of this paper is organized as follows. Section 2 introduces the spatial rank-based high-dimensional monitoring scheme with ensemble random projections, and the diagnostic procedure for root cause identification. Section 3 shows the thorough numerical studies of the proposed methodology. Section 4 applies the proposal to real-data examples. Section 5 concludes the paper with remarks. Some technical details are provided in the Appendices.

2 Spatial Rank-based High-Dimensional Monitoring Scheme

2.1 A review of spatial rank-based EWMA control chart

Spatial rank-based method is often used to construct robust tests for location testing problems (Oja 2010). It has robust detection power for general continuous multivariate distributions and is especially efficient for the elliptical distribution family. Define the spatial sign function for a multivariate variable $\mathbf{X}$ as $U(\mathbf{X}) = \mathbf{X}/||\mathbf{X}||I(\mathbf{X} \neq \mathbf{0})$. The empirical spatial rank for the t^{th} online sample $\mathbf{X}_t$ can be defined as the average of the spatial signs of pairwise differences, i.e.,

$$R(\mathbf{X}_t) = \frac{1}{m_0 + t - 1} \sum_{i=-m_0+1}^{t-1} U(\mathbf{X}_t - \mathbf{X}_i). \tag{2}$$

Given the IC distribution $\mathbf{X} \sim F_0(\mathbf{X} - \boldsymbol{\mu}_0)$, we want to test the null hypothesis, $H_0 : \boldsymbol{\mu} = \boldsymbol{\mu}_0$ against the alternative hypothesis $H_1 : \boldsymbol{\mu} \neq \boldsymbol{\mu}_0$. Under H_0, we have $\mathrm{E}(R(\mathbf{X}_t)) = \mathbf{0}$. Otherwise under H_1, $\mathrm{E}(R(\mathbf{X}_t))$ is generally not zero (see Chapter 4 in Oja (2010) for more discussions). Thus, the T^2-type test statistic $R'(\mathbf{X}_t)\{\mathrm{Cov}\,[R(\mathbf{X}_t)]\}^{-1}R(\mathbf{X}_t)$ is a reasonable candidate test statistic for the hypothesis. When $\mathbf{X}_t$ is still in control, the test statistic should be small, while a large value of the test statistic should reject the null hypothesis. Since the spatial rank releases the heavy tails' influence on the T^2-type test, the spatial rank test has efficient detection power for elliptical distributions. However, this test statistic is not affine invariant. It means that if the sample $\mathbf{X}_t$ is transformed to $\mathbf{D}\mathbf{X}_t$ by any full-rank matrix $\mathbf{D}$, the statistic value will be changed. Consequently the control limit should be adjusted when different coordinate systems are used, which is rather unappealing. To rectify this issue, we make an affine-invariant modification by invariantly transforming $\mathbf{X}_t$ to $\mathbf{M}\mathbf{X}_t$, in which $\boldsymbol{\Sigma} = (\mathbf{M}'\mathbf{M})^{-1}$ and $\boldsymbol{\Sigma}$ is the covariance matrix of $\mathbf{X}_t$. Then we construct the spatial rank test as

$$Q_t = R'(\mathbf{M}\mathbf{X}_t)\{\mathrm{Cov}\,[R(\mathbf{M}\mathbf{X}_t)]\}^{-1}R(\mathbf{M}\mathbf{X}_t).$$

According to Zou et al. (2012), for $\mathbf{X}_t$ of elliptical distributions, when the process is IC, we have $\mathrm{Cov}(R(\mathbf{M}\mathbf{X}_t)) = \mathrm{E}\left[||R(\mathbf{M}\mathbf{X}_t)||^2\right]\mathbf{I}_p/p$. Furthermore, to increase the detection power of Q_t for small changes, we can incorporate the EMWA technique and construct the charting statistic for $\mathbf{X}_t$ as

$$Q_t = \frac{(2-\lambda)p}{\lambda\xi}||\mathbf{v}_t||^2, \tag{3}$$

$$\mathbf{v}_t = (1-\lambda)\mathbf{v}_{t-1} + \lambda R(\mathbf{M}\mathbf{X}_t),$$

with $\xi = E[||R(\mathbf{M}\mathbf{X}_t)||^2]$. This spatial rank test also performs robustly for general continuous distributions, as demonstrated by numerical studies in Section 3.

2.2 Random projection for dimension reduction

Usually $\boldsymbol{\Sigma}$ is unknown and needs to be estimated from the m_0 reference samples. However, when $p > m_0$, the estimated matrix $\hat{\boldsymbol{\Sigma}}$ is singular, and the chart cannot be started. Even when $p \leq m_0$, the test statistic will still perform poorly for small t if p is nearly as large as m_0. One possible solution is dimension reduction, i.e., to reduce p into a much smaller $k < m_0$. Then these m_0 samples are enough to learn the k-dimensional distribution well. Consider projecting the original p-dimensional samples $\{\mathbf{X}_{-m_0+1}, \ldots, \mathbf{X}_t\}$ to a k-dimensional ($k \ll p$) subspace as samples $\{\mathbf{Y}_{-m_0+1}, \ldots, \mathbf{Y}_t\}$ using a matrix $\mathbf{P} \in \mathbb{R}^{p\times k}$, i.e., $\mathbf{Y}_i = \mathbf{P}'\mathbf{X}_i(i = -m_0+1, \ldots, t)$. Then the statistical distance between H_0 and H_1 in the projected space equals $d_k = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)'\mathbf{P}(\mathbf{P}'\boldsymbol{\Sigma}\mathbf{P})^{-1}\mathbf{P}'(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)$ for elliptical distributions. As we know, generally a larger k leads to a larger d_k and separates H_0 and H_1 "further", indicating less information loss. However, on the other hand, when $\boldsymbol{\Sigma}$ is unknown and substituted by $\hat{\boldsymbol{\Sigma}}$, a larger k will lead to a larger accumulated uncertainty in $(\mathbf{P}'\hat{\boldsymbol{\Sigma}}\mathbf{P})^{-1}$, which masks the true distance to some degree and makes it difficult to discriminate between the hypotheses. Given a certain k, traditional dimension reduction methods, such as LDA, aim at finding a projection matrix $\mathbf{P}$ based on $\hat{\boldsymbol{\Sigma}}$ to maximize (preserve) d_k as much as possible. However, unless k is very small ($k \ll m_0$), the uncertainty in $\hat{\boldsymbol{\Sigma}}$ will influence the estimation of $\mathbf{P}$ and make the optimal d_k unattainable. In these cases, one possible solution is to construct a projection which does not depend on $\boldsymbol{\Sigma}$. With in mind that d_k is also highly dependent on the Euclidean distance between $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_0$, we can alternatively seek for a projection $\mathbf{P}$ that preserves the Euclidean distance of $\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0$ mostly. In other words, we hope to preserve the geometric structure of $\mathbf{X}_t$ after projection. This brings us the idea of random projection.

In random projection, $\mathbf{P}$ does not depend on the data, but is randomly generated by drawing its each element from the normal distribution with mean 0 and variance $1/k$. This strategy, on the one hand, guarantees that $\mathbf{P}$ can be constructed rapidly with much smaller computation time than PCA or ICA. On the other hand, it can also preserve the geometry structure of the data with high probability. Specifically, according to the Johnson-Lindenstrauss lemma (Johnson and Lindenstrauss 1984), as long as $k > C\epsilon^{-2}\log(\rho^{-1})$, where $\epsilon \in (0,1)$ is the disturbing level and $C > 0$ is a large constant, the random projection $\mathbf{P}$ guarantees that $\forall i, j \in [-m_0+1, t]$ and $i \neq j$,

$$\Pr\{(1-\epsilon)||\mathbf{X}_i - \mathbf{X}_j||^2 \leq ||\mathbf{Y}_i - \mathbf{Y}_j||^2 \leq (1+\epsilon)||\mathbf{X}_i - \mathbf{X}_j||^2\} \geq 1-\rho.$$

It means that the distance between any two points, $\mathbf{X}_i$ and $\mathbf{X}_j$, is approximately preserved in the subspace points $\mathbf{Y}_i$ and $\mathbf{Y}_j$. Strictly speaking, $\mathbf{P}$ is not necessary to be a projection since each column of $\mathbf{P}$ is not restricted to be orthogonal with each other. However, orthogonalizing $\mathbf{P}$ is computationally expensive and hence undesirable. Fortunately, according to Hecht-Nielsen (1994), in a high-dimensional space, there exists a much larger number of near orthogonal directions than orthogonal ones. This indicates that these random directions can be sufficiently close to orthogonal, and consequently $\mathbf{P}'\mathbf{P}$ would approximate to an identity matrix. Another point to be noted is that the elements of $\mathbf{P}$ are not necessarily to be drawn from the normal distribution. Any zero-mean Sub-Gaussian distribution is applicable. Some popular variances can be referred to Matoušek (2008).

Of course, the Johnson-Lindenstrauss lemma can only guarantee that with a probability higher than $1-\rho$, the projected distance is preserved with ϵ-level disturbance. There is still probability that the projected distance is not preserved, with disturbance bigger than ϵ. In this case, the dimension reduction will loss certain information and consequently affect the following data analysis. Actually, information loss is common in various dimension reduction methods. For example, PCA can only preserve data information in the selected principal components, while loses the information in unselected ones. Of course, we can alleviate the information loss by decreasing ρ, which can be achieved by increasing k. For example, for m_0 samples $\mathbf{X}_i, i = 1, \ldots, m_0$, if we want all pairwise distances to have ϵ-level disturbance, ρ should be around $1/m_0^2$ (or $1/m_0(m_0-1)$ more accurately). Consequently we need to set $k = O(\epsilon^{-2}\log m_0)$.

2.3 Spatial rank-based EWMA chart with ensemble random projections

Now we can first project $\{\mathbf{X}_{-m_0+1}, \ldots, \mathbf{X}_t\}$ into k-dimensional $\{\mathbf{Y}_{-m_0}, \ldots, \mathbf{Y}_t\}$ using a random projection $\mathbf{P}$ with $k \ll m_0$, and then construct the spatial rank-based EWMA chart on $\{\mathbf{Y}_{-m_0+1}, \ldots, \mathbf{Y}_t\}$

as

$$Q_t = \frac{(2-\lambda)k}{\lambda\xi}||\mathbf{v}_t||^2,$$
$$\mathbf{v}_t = (1-\lambda)\mathbf{v}_{t-1} + \lambda R(\hat{\mathbf{M}}\mathbf{Y}_t),$$

with $\mathbf{v}_0 = \mathbf{0}$ and $\xi = \mathrm{E}\left[||R(\hat{\mathbf{M}}\mathbf{Y}_t)||^2\right]$. $\hat{\mathbf{M}}$ can be derived from the sample covariance matrix $\hat{\mathbf{\Sigma}}$ of $\{\mathbf{Y}_{-m_0+1}, \ldots, \mathbf{Y}_0\}$ according to $\hat{\mathbf{\Sigma}} = (\hat{\mathbf{M}}^T\hat{\mathbf{M}})^{-1}$. Though in this way we make the monitoring scheme applicable, as mentioned previously, the dimension reduction is accompanied with information loss. Specifically, for a certain projection, it only preserves the data structure in this subspace, but loses data information in other subspaces which are orthogonal to the current one. One mitigation strategy is to use ensemble projections, i.e., projecting $\mathbf{X}_t$ into different subspaces using different $\mathbf{P}$ and combining their monitoring results together. By setting different $\mathbf{P}$ orthogonal with each other, different projections reveal the data structure from different perspectives (subspaces) and consequently complement each other. To be more specific, consider S orthogonal random projections, $\mathbf{P}^s = [\mathbf{p}_1^s, \ldots, \mathbf{p}_k^s]\,(s = 1, \ldots, S)$, where $\mathbf{p}_l^s$ is its l^{th} column vector for $l = 1, \ldots, k$. In particular, $\mathbf{p}_l^s \perp \mathbf{p}_m^j, \forall s, j = 1, \ldots, S, s \neq j; l, m = 1, \ldots, k$. However, $\mathbf{p}_l^s$ and $\mathbf{p}_m^s$ do not need to be orthogonal with each other. The detailed generation procedure of $\mathbf{P}^s(s = 1, \ldots, S)$ and its property are shown in the Appendix A. Then we construct the test statistic for each projection as $Q_t^1, \ldots, Q_t^s$, and aggregate these statistics together, i.e.,

$$Q_t^0 = \sum_{s=1}^{S} Q_t^s, \tag{4}$$

as the final monitoring statistic. Correspondingly, we chose a control limit $L > 0$ for (4) and define if $Q_t^0 > L$, the monitoring scheme triggers an OC alarm at sample t. Henceforth we denote this monitoring scheme as the RPSR chart for abbreviation.

2.4 Additional remarks

1. The choices of k and S are crucial for the detection power of RPSR. For k, on the one hand, according to the Johnson-Lindenstrauss lemma, k determines the disturbing level ϵ and the accuracy level ρ. A higher k leads to a lower disturbing level and a higher accuracy level. However, on the other hand, with limited m_0 reference samples, increasing k too much leads

to a more noisy estimation of the covariance matrix of $\mathbf{Y}_t^s = \mathbf{P}^{s\prime}\mathbf{X}_t$, and thus deteriorates the monitoring results. Therefore, the optimal choice of k should balance these two sides. As analyzed earlier, for m_0 samples, if we want all pairwise distances have ϵ-level disturbance, we recommend $k = O(\epsilon^{-2}\log m_0)$ with $\epsilon \in (0, 0.5)$. For S, we recommend to set $S = \lfloor p/k \rceil$ to ensure that the data structure in the original space can be discovered using these S complementary subspaces. Of course, it is also possible to set $S \geq \lfloor p/k \rceil$ (in this case, these S projections may no longer be orthogonal with other). Then we can get better detection results for RPSR, since we can observe the original high dimensional space with more information. However, the performance improvement is not very cost-efficient compared with the additionally brought computational cost (this will be demonstrated in Section 3).

2. When the chosen $k \geq m_0/2$, the estimated $\hat{\boldsymbol{\Sigma}}^s$ for each subspace may not be accurate enough (Zou et al. 2012). As such, we may use the self-starting technique to update $\hat{\boldsymbol{\Sigma}}^s$ together with $\hat{\mathbf{M}}^s$ when more online samples are gathered. In particular, when a new online sample $\mathbf{X}_t$ is available, we recalculate (or update) the sample covariance matrix $\hat{\boldsymbol{\Sigma}}_{t-1}^s$ using $\{\mathbf{Y}_{-m_0+1}^s, \ldots, \mathbf{Y}_{t-1}^s\}$ for every subspace, and get $\hat{\boldsymbol{\Sigma}}_{t-1}^s = (\hat{\mathbf{M}}_{t-1}^{sT}\hat{\mathbf{M}}_{t-1}^s)^{-1}$. Then we have the monitoring statistic as

$$Q_t^s = \frac{(2-\lambda)k}{\lambda\xi_t}||\mathbf{v}_t||^2,$$
$$\mathbf{v}_t = (1-\lambda)\mathbf{v}_{t-1} + \lambda R(\hat{\mathbf{M}}_{t-1}^s \mathbf{Y}_t^s),$$

with $\mathbf{v}_0 = \mathbf{0}$ and $\xi_t = \mathrm{E}\left[||R(\hat{\mathbf{M}}_{t-1}^s \mathbf{Y}_t^s)||^2\right]$. The corresponding formula can be further derived as

$$\mathrm{E}\left[||R(\hat{\mathbf{M}}_{t-1}^s \mathbf{Y}_t^s)||^2\right] = \frac{1}{m_0+t-1}\left[\sum_{i=-m_0+1}^{0}||R(\hat{\mathbf{M}}_0 \mathbf{Y}_i^s)||^2 + \sum_{i=1}^{t-1}||R(\hat{\mathbf{M}}_{i-1}^s \mathbf{Y}_i^s)||^2\right],$$

with $R(\hat{\mathbf{M}}_0^s \mathbf{Y}_i^s) = \frac{1}{m_0}\sum_{k=-m_0+1}^{0} U(\hat{\mathbf{M}}_0^s(\mathbf{Y}_i^s - \mathbf{Y}_k^s))$ for $k = -m_0+1, \ldots, 0$.

However, it should be noted that the self-starting technique may bring in another problem: the probability that each pairwise distance of the m_0+t points is not disturbed than ϵ, which equals $\alpha_t = 1 - \binom{m_0+t}{2}\rho$, is decreasing as t increases. In other words, if we want $\alpha_t \geq 0.5$, we need to set $\rho = 1/(m_0+t)^2$, with $k \in O(\epsilon^{-2}\log(m_0+t))$, which is unpleasant. Fortunately, the increase speed of the required k is very small as t increases. Then we may set $k \in O(\epsilon^{-2}\log(m_0+T_0))$, where T_0 is the interested time horizon. In this way we can keep $\alpha_t \geq \alpha_{T_0} \geq 0.5$ for all $t = 1, \ldots, T_0$.

Furthermore, when t becomes large enough, we no longer need to update $\hat{\mathbf{\Sigma}}_t^s$ or $\hat{\mathbf{M}}_t^s$, since they are already sufficiently close to the true values. Without updating, the sample space to be projected is always $m_0 + t_1$ where t_1 is the time point stopping updating. Since the sample space no longer increases with t, α_t will always keep consistent after t_1.

3. To ensure fast implementation of the monitoring scheme with the self-starting technique, we need to consider its computational complexity. To construct Q_t^s, for every subspace, we need to calculate $\hat{\mathbf{M}}_{t-1}^s$ from $\hat{\mathbf{\Sigma}}_{t-1}^s$. Although any matrix that satisfies $\hat{\mathbf{\Sigma}}_{t-1}^s = (\hat{\mathbf{M}}_{t-1}^{s\prime}\hat{\mathbf{M}}_{t-1}^s)^{-1}$ will suffice for transforming $\mathbf{Y}_t^s$ to achieve the affine-invariant property, similar to Zou et al. (2012), we choose a particularly attractive one, the triangular Cholesterol inverse root of $\hat{\mathbf{\Sigma}}_{t-1}^s$, which takes $\hat{\mathbf{M}}_{t-1}^s$ as an upper triangular matrix. Then we estimate $\hat{\mathbf{M}}_{t-1}^s$ using the ranking-one downdating Cholesky factorization with a computational effort of $O(k^2)$. Ranking $\mathbf{Y}_t^s$ further takes $O(m_0+t)$ computational effort. Therefore, the total computational effort for Q_t^s at t is $O(Sk^2 + Sk(m_0 + t))$, which is not very computationally expensive in front of modern computing resources. For example, for a experiment setting with $p = 100, m_0 = 100, k = 20$ and $S = 5$, the calculation of Q_{100}^s takes 0.006 seconds on a personal single-core desktop. For calculating Q_t^0 without self-starting, the computational effort is $Q(Sk(m_0 + t))$.

4. Now we talk about how to set the control limit L for RPSR. When m_0 is large enough, $\hat{\mathbf{\Sigma}}^s$ and $\hat{\mathbf{M}}^s$ can be well estimated. We do not need to estimate or update $\hat{\mathbf{M}}_t^s$ at every t. In this case, Q_t^0 is a Markov chain, then the run length as well as L can be approximately calibrated through the Markov chain approach (Runger and Prabhu 1996). However, When m_0 is small and the self-starting technique is adopted for small t, the conditional distribution of Q_t^0 is considerably different from the steady-state one. In such case, L depends not only on λ and the prescribed IC ARL, i.e., ARL_0, but also on m_0. Therefore we suggest calculating L based on simulation. Given a process setting (with a certain p, k and S, the reference sample size m_0, the control chart setting λ, and ARL_0), the IC run length distribution of the monitoring statistic is very robust under various process distributions, even including very skew distributions (Zou et al. 2012). This means that for a certain application process in practice with limited samples, we may use a multivariate normal distribution instead to generate "fake" samples and use them to calculate the control limit for this application (Zou et al. 2012). In this way, we can guarantee that the control chart can still start under limited samples in practice. The detailed simulation procedure is shown in Algorithm 1. Specifically, given a certain m_0, we simulate the monitoring scheme for in total N replications (say $N = 50,000$ used in this paper) by drawing $\mathbf{X}$ from the standard

multivariate normal distribution. Then we use the bisection algorithm (Qiu et al. 2010) to search the control limit L so that the IC ARL of these N replications equals the prescribed one.

Algorithm 1 Procedure to find the control limit

```
Input: the prescribed $ARL_0$, the number of simulation replications $N$, the maximum run length $T_0$ where $T_0 \gg ARL_0$, and the control chart parameters including $p$, $k$, $S$, $m_0$, $\lambda$.
Output: The calculated control limit $L$ which ensures the chart has IC ARL equal to $ARL_0$.
for $b = 1$ to $N$ do
    Generate samples $\mathbf{X}_{-m_0+1}, \ldots, \mathbf{X}_0, \mathbf{X}_1, \ldots, \mathbf{X}_{T_0}$ from the standard multivariate normal distribution for this replication $b$.
    Run the control chart with $L = \infty$, and get the charting statistic $\{Q_{bt}, t = 1, \ldots, T_0\}$ for this replication $b$.
end for
Set the $L_u = \min_{b=1,\ldots,N} \max_{t=1,\ldots,T_0} Q_{bt}$ and $L_l = 0$.
Calculate the $ARL$ given $L = L_u$ using the $N$ simulation replications $\{Q_{bt}, t = 1, \ldots, T_0\}, b = 1, \ldots, N$.
if $ARL < ARL_0$ then
    Increase $T_0$, and go back to regenerate $N$ sequences.
end if
while $|ARL - ARL_0| \geq \epsilon$ do
    if $ARL > ARL_0$ then
        $L = (L + L_l)/2$
    else
        $L = (L + L_u)/2$
    end if
    Calculate the $ARL$ given the new $L$ using the $N$ simulation replications $\{Q_{bt}, t = 1, \ldots, T_0\}, b = 1, \ldots, N$.
end while
```

2.5 Post-signal diagnostic procedure

Now we consider how to identify the process mean change $\boldsymbol{\delta} = \boldsymbol{\mu}_1 - \boldsymbol{\mu}_0$ after an OC signal is triggered at time t. A natural diagnostic scheme can be designed as

$$\hat{\boldsymbol{\delta}} = \sum_{i=1}^{t} w_i \mathbf{X}_i - \frac{1}{m_0} \sum_{i=-m_0+1}^{0} \mathbf{X}_i, \tag{5}$$

with the tuning parameter $w_i = \kappa(1-\kappa)^{t-i} / \sum_{j=1}^{t} \kappa(1-\kappa)^{t-j}$ where $0 < \kappa \leq 1$. It means that we estimate $\boldsymbol{\mu}_1$ by weighted averaging the online samples before t in an exponentially decaying way using the tuning parameter κ, so that more emphases are put on the recent data with fewer emphases on the old ones. However, (5) may include IC noise. Actually, in a high-dimensional process, the probability that all parameters shift simultaneously is rather low. Therefore, it is more reasonable to assume that most components of $\mathbf{X}_t$ are still in control, and a fault is more likely to be caused by unknown changes that only reflect one or a small set of variables. This is so-called sparsity characteristic. Jiang et al. (2012) and Zou and Qiu (2009) addressed this problem by forcing L_1 penalty on $\boldsymbol{\mu}_1$, and estimating $\boldsymbol{\mu}_1$ by either forward variable selection or adaptive LASSO algorithm. Though their diagnostic methods have quite satisfactory performance for MSPC with the normal distribution assumption, LASSO is very sensitive to outliers and hence cannot achieve robust estimation results for non-normal distributions with heavy tails. One revised shrinkage algorithm which dispenses with this problem is the square-root LASSO (Belloni et al. 2011). It can achieve near-oracle performance without assuming the process is normal. With these in mind, we propose a diagnostic framework based on the square-root LASSO. In particular, we estimate $\boldsymbol{\delta}$ by

$$\hat{\boldsymbol{\delta}} \in \arg\min_{\boldsymbol{\delta}\in\mathbb{R}^p} \{Q(\boldsymbol{\delta})\}^{1/2} + \frac{\gamma}{k}||\boldsymbol{\delta}||_1, \tag{6}$$

where $Q(\boldsymbol{\delta}) = \sum_{i=1}^{t}\sum_{s=1}^{S} w_i \left[\hat{\mathbf{M}}^s\mathbf{P}^{s\prime}(\mathbf{X}_i - \boldsymbol{\mu}_0 - \boldsymbol{\delta})\right]' \left[\hat{\mathbf{M}}^s\mathbf{P}^{s\prime}(\mathbf{X}_i - \boldsymbol{\mu}_0 - \boldsymbol{\delta})\right]$; γ is the penalty level and $||\boldsymbol{\delta}||_1$ is the l_1 norm of $\boldsymbol{\delta}$. In (6), the part $Q(\boldsymbol{\delta})^{1/2}$ provides an accurate estimation of $\boldsymbol{\delta}$, the part $||\boldsymbol{\delta}||_1$ restricts the number of changed components. The key of the square-root LASSO is the score, i.e., the magnitude of the gradient of $Q^{1/2}(\boldsymbol{\delta})$ evaluated at the true parameter value $\boldsymbol{\delta}_0$. The square-root LASSO ensures that, with a probability larger than $1-\alpha$, the score is smaller than the penalty of $\boldsymbol{\delta}$, i.e., γ/S. In this case, when $\boldsymbol{\delta}_0 = \mathbf{0}$, the probability of miss-identifying a nonzero $\boldsymbol{\delta}$ would be smaller than α. With this in mind, for multivariate normal distributions, we need to set $\gamma \geq \sqrt{kS}\Phi^{-1}(1-\alpha/2p)/\sigma_{min}$ where $\sigma_{\min} = \min_{j=1,\ldots,p}\text{std}(X_{tj})$ and Φ^{-1} is the inverse CDF of the standard normal distribution. As such, we set $\gamma = c\sqrt{kS}\Phi^{-1}(1-\alpha/2p)/\sigma_{min}$ where $c = 1.1$ is a control constant and $\alpha = 0.05$. For non-normal distributions, Belloni et al. (2011) demonstrates that the above γ with Gaussian CDF Φ is still a valid asymptotic choice. Therefore, the square-root LASSO has more robust performance than LASSO-based diagnostic methods for non-Gaussian processes. (6) can be formulated as a convex conic programming problem and solved using efficient algorithmic methods. More discussions about its properties and solving procedures can be referred in Belloni et al. (2011). When $t \to \infty$, $Q(\boldsymbol{\delta})$ can be approximated in a recursive way as $Q(\boldsymbol{\delta}) = \sum_{s=1}^{S}\left[\hat{\mathbf{M}}^s(\mathbf{z}_t^s - \boldsymbol{\delta})\right]'\left[\hat{\mathbf{M}}^s(\mathbf{z}_t^s - \boldsymbol{\delta})\right]$ where $\mathbf{z}_t^s = (1-\kappa)\mathbf{z}_{t-1}^s + \kappa\mathbf{P}^{s\prime}(\mathbf{X}_i - \hat{\boldsymbol{\mu}}_0)$, with $\mathbf{z}_t^s = \mathbf{0}(s = 1,\ldots,S)$. In this way solving (6) will save a lot of computer memory when p and t are huge. In the next section, we use some numerical studies to demonstrate the efficiency of this diagnostic framework.

3 Numerical Results

In this section, we evaluate the performance of RPSR through some numerical studies. In particular, in 3.1, we study the performance of RPSR for different OC scenarios. In 3.2 we study the performance of the post-signal diagnostic framework based on the square-root LASSO.

We consider the following process distributions in our numerical studies: (i) p-dimensional normal distribution, denoted as $\mathcal{N}_p$; (ii) p-dimensional t distribution with 5 degrees of freedom, denoted as $t_{p,5}$; (iii) p-dimensional gamma distribution with shape parameter 3 and scale parameter 1, denoted as $\text{Gam}_{p,3}$. Here $\mathcal{N}_p$ and $t_{p,5}$ belong to the elliptical distribution family, while $t_{p,5}$ has heavier tails than $\mathcal{N}_p$. In contrast, $\text{Gam}_{p,3}$ is not elliptical. These distributions are commonly used in the literature to study the robustness of nonparametric charting performance (Zou et al. 2012; Chen et al. 2016; Zhang et al. 2016). In the simulation, we consider $p = 100$ and 500, representing high-dimensional and ultra high-dimensional cases respectively. We set $m_0 = 100$. Clearly, such few reference samples are not sufficient to provide any meaningful estimation of the distributional parameters for such a large p.

3.1 Out-of-control performances

Without loss of generality, for each distribution, the mean vector $\boldsymbol{\mu}_0$ is set to be $\mathbf{0}$. The covariance matrix $\boldsymbol{\Sigma}$ is chosen to be a block matrix as $\boldsymbol{\Sigma} = \text{diag}(\sigma_1\boldsymbol{\Phi}, \sigma_2\boldsymbol{\Phi}, \ldots, \sigma_5\boldsymbol{\Phi})$ where $\boldsymbol{\Phi} = (\phi)_{p/5\times p/5}$ is set to be $\phi_{ij} = 0.5^{|i-j|}, \forall i, j = 1, \ldots, p/5$, and $\sigma_r = 1.5^{r-1}$ for $r = 1, \ldots, 5$. In this way the components in the same block have strong correlations with each other, while components between different blocks have no correlation at all. Since it is impossible to enumerate all the change patterns to allow a full-scale study of the charting performance, following similar studies in Zou and Tsung (2011); Zou et al. (2012), here we consider shifts in the first $0.06p$ components of the process mean vector with size δ, i.e., $\boldsymbol{\mu}_1 = \boldsymbol{\mu}_0 + \delta\mathbf{e}$ with $\mathbf{e} = (\underbrace{1, \ldots, 1}_{1,\ldots,0.06p}, 0, \ldots, 0)^T$. For RPSR, we set $k = 20$, and $S = 5$ for $p = 100$ and $S = 25$ for $p = 500$ respectively.

We first demonstrate the robustness of RPSR by comparing it with its counterparts. In particular, we consider the random projection based chart without the spatial rank test. Specifically, for each subspace with $\mathbf{Y}_t^s (s = 1, \ldots, S)$, we construct its T^2 chart, i.e., $T_t^s = \mathbf{Y}_t^{s\prime}\hat{\boldsymbol{\Sigma}}_{t-1}^s\mathbf{Y}_t^s$, and sum all the $T_t^s (s = 1, \ldots, S)$ together for monitoring, denoted as RPT2. We also consider the chart that directly applies the spatial rank test (Zou et al. 2012), denoted as SR. Furthermore, we consider the chart that first applies PCA for data and then uses the first k PCA scores in the spatial rank test for monitoring, denoted as PCASR. To make a fair comparison, we incorporate the EMWA technique for all the charts. We fix $m_0 = 100$, $\tau = 50$ and set $\lambda = 0.1$ in all the cases. Note that for the RPSR and RPT2 charts, we do not incorporate the self-starting technique, i.e., not update $\hat{\mathbf{\Sigma}}_t^s$ on line but only use $\hat{\mathbf{\Sigma}}_0^s$ all the time, since $m_0 = 100$ is large enough to estimate a distribution with $k = 20$. However, for the other PCASR and SR charts, since $m_0 = 100$ is not enough to estimate a distribution or a correlation matrix with $p = 100$, we incorporate the self-starting technique and update $\hat{\mathbf{\Sigma}}_t^s$. We find the control limits for these charts to ensure their IC ARL= 200. In particular, for RPT2 and RPSR, their control limits are calculated using the Markov Chain approach (Runger and Prabhu 1996). For PCASR and SR, their control limits are calculated using Algorithm 1 with samples generated from their process distributions (Note that this is impossible in practice, since we do not know the real distributions in advance). Table 1 illustrates the OC performance of RPSR, RPT2, SR, PCASR with $k = 10$, and PCASR with $k = 99$ for these three process distributions with $p = 100$ in detecting mean shifts of size $\delta = 0.25, 0.5, 1, 2, 4$. We use the steady state OC ARL for performance evaluation. This means that any series where an OC alarm is triggered before the true change point τ is discarded. All the ARLs shown in this paper are based on 10,000 replications, and the standard deviations of the run length (SDRL) are also shown in parentheses. We can see that RPT2 performs better than RPSR for normal distributions, this makes sense since the T^2 chart is designed particularly for multivariate normal processes. However, when the process becomes non-normal, RPT2 loses its efficiency. Yet RPSR still has satisfactory detection power, demonstrating its robust efficiency for general process distributions. This result is consistent with Zou et al. (2012). Furthermore, it is clear that RPSR has the best performance among these methods. For the other three charts, even with the self-starting technique, the accumulated $m_0 + \tau = 150$ IC samples are still not enough for estimating $\mathbf{\Sigma}$ at all. For the two PCASR charts, the PCASR with $k = 10$ performs better than the PCASR with $k = 99$. At the first glance this seems surprising since more principal components mean more potential change directions to be detected. However, this is reasonable, since these principal components are so poorly estimated that they do not increase the detection power, but add more noise to the chart and consequently deteriorate the detection power. For the SR chart, it performs better than PCASR with $k = 10$ for small shifts, but worse for large shifts. This indicates that PCA is actually not a good choice to improve the performance of SR when $p \geq m_0$.

To evaluate the influence of S, here we further consider two other RPSRs with $S = 10$ (denoted as RPSR(10)) and $S = 2$ (denoted as RPSR(2)). These two are only for illustration but not recommended. For RPSR(10), we construct the random projections by running Algorithm 2 twice. We set $\mathbf{P}^{1:10}$ using the projections of the first run and set $\mathbf{P}^{11:20}$ using the projections of the second run. As shown in Table 1, as S increases, the detection power of RPSR increases. However, the increase magnitude is not very cost-efficient from $S = 5$ to $S = 10$, at the price of double computational effort. Therefore we still recommend $S = \lfloor p/k \rceil$.

Table 1: OC ARL comparison in detecting mean shifts with $p = 100$ and $\lambda = 0.1$ (numbers in parentheses are SDRL values).

	δ	RPSR	RPSR(10)	RPSR(2)	RPT2	PCASR(10)	PCASR(99)	SR
$\mathcal{N}_{100}$	0	200(215)	200(212)	200(224)	200(195)	200(172)	200(351)	200(374)
	0.25	167(193)	162(186)	174(200)	139(118)	196(169)	195(341)	183(347)
	0.5	84.2(122)	79.6(117)	123(168)	61.4(47.4)	168(157)	185(353)	128(295)
	1	17.8(8.64)	16.2(7.99)	29.8(64.2)	18.1(6.12)	59.2(56.8)	168(332)	86.7(240)
	2	7.66(1.21)	7.26(0.61)	9.04(2.23)	8.56(1.45)	13.4(4.19)	145(309)	49.6(173)
	4	4.44(0.49)	4.21(0.42)	4.85(0.65)	4.91(0.45)	6.12(1.15)	87.5(238)	27.7(108)
$t_{100,5}$	0	200(226)	200(219)	200(218)	200(196)	200(186)	200(355)	200(345)
	0.25	176(200)	169(199)	184(213)	184(156)	193(185)	193(351)	185(348)
	0.5	125(132.7)	115(148)	143(175)	157(117)	176(168)	178(340)	133(301)
	1	28.7(40.1)	27.3(34.5)	42.8(67.0)	60.5(49.4)	112(131)	175(337)	90.9(247)
	2	9.00(1.77)	8.67(1.67)	10.9(3.94)	12.7(6.42)	19.7(9.67)	169(333)	43.3(156)
	4	4.97(0.54)	4.92(0.52)	5.51(0.89)	5.23(1.32)	7.63(1.79)	114(273)	38.6(144)
$\text{Gam}_{100,3}$	0	200(214)	200(213)	200(224)	200(197)	200(164)	200(383)	200(372)
	0.25	191(205)	190(201)	195(209)	194(189)	195(163)	196(371)	187(364)
	0.5	170(196)	167(194)	187(201)	174(182)	192(160)	193(360)	178(342)
	1	112(159)	109(142)	121(157)	111(121)	190(154)	190(353)	148(315)
	2	20.4(29.9)	19.4(28.4)	37.8(78.5)	63.2(86.7)	165(142)	189(353)	63.7(200)
	4	6.71(1.29)	6.54(1.05)	9.66(2.87)	34.9(25.1)	23.1(8.92)	152(315)	28.8(114)

Furthermore, to better illustrate the performance of RPSR, we also compare it with three other state-of-the-arts methods, the CUSUM chart by Mei (2010) (shorted as MCUSUM), the anti-rank chart by Qiu and Hawkins (2003) (shorted as AnRank), and the Kernel Hibert chart by Huang et al. (2014) (shorted as RKHS). For MCUSUM, we assume the shift direction is known and consequently use the one-side chart for comparison. We set its parameter k, i.e., the target shift size, as 0.25, because it is the smallest shift size we are interested in. For the other two EWMA-type charts, here two values of λ, 0.1 and 0.025, are considered separately. As Table 2 shows, for either $\lambda = 0.1$ or $\lambda = 0.025$, RPSR performs consistently better than MCUSUM, RKHS, and AnRank, because they do not consider correlations between different variables at all. This illustrates the efficiency and superiority of the proposed method. It should be noted that for $\text{Gam}_{p,3}$, all the charts lose their detection power to some degree. This performance deterioration is caused by the severe skewness of the gamma distribution.

We further consider the influence of m_0 on the monitoring performance. In particular, for the multivariate normal process with $p = 100$, we consider $m_0 = 60, 80, 100, 120, 140, 210$. First, for different m_0, we consider fixing $k = 20$ and $S = 5$. The performance of the RPSR chart is shown in Table 3. As m_0 increases, the RPSR chart has better detection power generally, since it can estimate the IC parameters more accurately. However, as m_0 exceeds 100, the performance increase becomes slower, especially for larger shifts with $\delta > 0.5$. This is because when $m_0 = 100$, the IC parameters can be estimated well enough, hence the contribution of a larger m_0 will become negligible. Furthermore, we also consider changing k and the corresponding S for different m_0, as shown in Table 4. Similar to Table 3, as m_0 increases, the RPSR chart becomes more efficient. This is reasonable since we have more samples for the IC process. However, it should be noted that for the same m_0, the RPSR in Table 4 performs better than that in Table 3. This indicates that adjusting k and S according to different m_0 in practice is necessary. For smaller m_0, a smaller k is preferable, since it can guarantee the parameters to be well estimated in Phase I. On the contrary, for larger m_0, a larger k is more beneficial, because it can guarantee less loss caused by the projection, while a smaller k is a bit wasteful.

3.2 Efficiency of the diagnostic procedure

The above study demonstrates the detection efficiency of RPSR. We now investigate the capability of the diagnostic algorithm in identifying the OC variables after an OC alarm is triggered. Suppose s_0 is the subset of the truly changed variables, and $\hat{s}$ is a subset of $\{1, ..., p\}$ determined by the diagnostic algorithm. We use the true positive percentage (TPP) (Jiang et al. 2012), i.e.,

$$P_c = \mathrm{E}(\frac{|s_0 \cap \hat{s}|}{|s_0|}),$$

where "$|\cdot|$" denotes the number of components in the set, to evaluate the overall quality of the diagnostic algorithm.

We also compare the proposed square-root LASSO based diagnostic framework (denoted as SRLS) with the commonly used LASSO based diagnostic framework (denoted as LASSO) (Wang and Jiang 2009; Zou and Tsung 2011). However, the LASSO algorithm is sensitive to outliers, and consequently has degenerated performance for non-normal distributions with heavy tails. Tables 5 and 6 report the diagnostic results of SRLS and LASSO for the RPSR chart with $p = 100$, $k = 20$, $S = 5$, and $\lambda = 0.1$. Here two values of κ, 0.1 and 0.025, are considered. As shown in Tables 5 and 6, SRLS performs

Table 2: OC ARL comparison in detecting mean shifts when $m_0 = 100$ (numbers in parentheses are SDRL values).

			$\lambda = 0.1$		$\lambda = 0.025$		Other methods	
	p	δ	RPSR	RKHS	RPSR	RKHS	MCUSUM	AnRank
$\mathcal{N}_p$	100	0	200(215)	200(220)	197(177)	200(221)	201(275)	200(338)
		0.25	167(193)	194(214)	151(152)	180(211)	127(106)	179(313)
		0.5	84.2(122)	147(159)	67.0(50.9)	121(137)	85.1(41.6)	152(276)
		1	17.8(8.64)	66.7(75.4)	26.6(5.60)	41.5(28.1)	51.5(17.3)	129(243)
		2	7.66(1.20)	3.13(2.95)	13.7(1.43)	17.2(6.97)	31.3(7.04)	119(225)
		4	4.40(0.49)	1.00(0.67)	7.99(0.49)	8.27(2.49)	18.2(3.17)	110(220)
	500	0	201(225)	200(226)	200(218)	201(239)	200(260)	200(324)
		0.25	176(214)	200(221)	147(146)	174(203)	122(88.9)	197(310)
		0.5	107(171)	172(187)	56.9(30.0)	115(143)	84.3(40.5)	178(291)
		1	15.3(4.65)	83.3(83.8)	24.9(3.40)	42.7(28.1)	51.7(15.8)	161(261)
		2	7.07(0.78)	18.8(8.69)	13.2(0.93)	16.9(6.34)	31.8(7.06)	156(257)
		4	4.11(0.31)	6.60(1.61)	7.82(0.39)	8.25(2.44)	18.3(3.00)	153(251)
$t_{p,5}$	100	0	200(226)	198(219)	200(163)	201(230)	199(226)	200(346)
		0.25	176(200)	189(207)	155(153)	177(202)	148(143)	186(320)
		0.5	125(162)	164(185)	80.8(77.9)	126(154)	103(67.4)	158(282)
		1	28.7(40.1)	98.4(101)	30.9(7.86)	51.8(39.5)	65.6(26.9)	135(250)
		2	9.00(1.77)	24.7(14.7)	15.3(1.93)	19.7(8.98)	38.7(11.3)	115(218)
		4	4.97(0.54)	7.86(2.18)	8.74(0.75)	9.37(3.04)	22.7(5.03)	110(212)
	500	0	199(226)	201(209)	200(187)	200(225)	200(211)	200(310)
		0.25	177(206)	193(196)	149(149)	215(235)	142(142)	198(302)
		0.5	123(164)	174(183)	67.9(42.1)	147(178)	99.5(57.1)	188(298)
		1	23.4(25.9)	104(112)	27.7(4.34)	55.4(40.8)	63.5(25.1)	161(268)
		2	8.26(1.17)	25.8(15.8)	14.6(1.26)	20.4(8.30)	38.1(11.1)	158(256)
		4	4.80(0.45)	7.98(2.08)	8.44(0.56)	9.46(2.86)	22.5(4.84)	155(254)
$\text{Gam}_{p,3}$	100	0	200(214)	202(223)	200(200)	201(224)	200(221)	200(7.51)
		0.25	191(205)	201(229)	181(163)	187(214)	155(109)	197(7.12)
		0.5	170(196)	195(211)	146(139)	152(185)	118(65.1)	195(7.12)
		1	112(159)	162(172)	73.9(62.8)	105(110)	78.6(27.8)	193(7.11)
		2	21.9(29.9)	84.1(85.4)	28.1(5.38)	37.3(21.8)	47.6(12.3)	190(7.02)
		4	8.12(1.29)	17.5(7.09)	14.3(1.36)	15.5(5.68)	28.4(5.45)	186(6.91)
	500	0	200(224)	200(177)	200(184)	200(221)	200(176)	200(5.42)
		0.25	193(221)	196(191)	182(177)	196(186)	150(94.2)	198(5.43)
		0.5	178(203)	185(186)	143(147)	188(212)	114(54.2)	199(4.45)
		1	122(170)	136(136)	59.0(30.4)	110(121)	78.5(26.4)	195(4.12)
		2	17.3(6.33)	72.7(74.2)	26.5(3.32)	37.1(19.9)	48.3(11.9)	193(4.21)
		4	7.62(0.79)	15.6(6.00)	14.0(1.45)	15.2(5.04)	28.5(5.00)	190(4.23)

Table 3: Detection power for different m_0 with fixed $k = 20$ and $S = 5$ for $\mathcal{N}_{100}$.

δ	$m_0 = 60$	$m_0 = 80$	$m_0 = 100$	$m_0 = 120$	$m_0 = 140$	$m_0 = 210$
0	200(258)	200(234)	200(215)	200(206)	202(210)	200(200)
0.25	180(238)	169(199)	167(193)	163(179)	161(191)	156(165)
0.5	129(198)	119(134)	84.2(122)	81.1(125)	79.3(121	76.9(97.2)
1	22.5(41.8)	19.7(10.4)	17.8(8.64)	17.7(8.12)	17.3(6.61)	17.1(6.86)
2	7.82(1.37)	7.69(1.24)	7.66(1.22)	7.64(1.18)	7.62(1.18)	7.57(1.11)
4	4.45(0.51)	4.39(0.51)	4.41(0.49)	4.39(0.49)	4.36(0.48)	4.36(0.48)

Table 4: Detection power for different m_0 with various k and S for $\mathcal{N}_{100}$.

	$m_0 = 60$	$m_0 = 80$	$m_0 = 100$	$m_0 = 140$	$m_0 = 210$
δ	$k = 17, S = 6$	$k = 20, S = 5$	$k = 25, S = 4$	$k = 33, S = 3$	$k = 50, S = 2$
0	200(209)	200(202)	200(229)	205(244)	200(221)
0.25	162(209)	169(195)	156(204)	143(208)	135(178)
0.5	125(197)	119(176)	83.5(151)	67.6(113)	45.9(59.9)
1	26.6(49.9)	19.7(46.7)	16.5(13.4)	15.1(5.51)	14.3(4.84)
2	7.93(1.53)	7.62(1.78)	7.34(1.14)	7.15(1.05)	6.83(0.95)
4	4.52(0.51)	4.39(0.48)	4.28(0.45)	4.19(0.37)	4.11(0.31)

better than LASSO, especially for t_5 and Gam_3 distributions. Furthermore, we can see a clear pattern that when the shift size increases, P_c increases accordingly. Although this result might be intuitive, it is not trivial. Though larger shifts are easier for identification, they yet have smaller OC ARLs and consequently fewer OC samples available for diagnosis. Therefore, it is reasonable but not obvious to see that the identification probability actually improves (Zou and Qiu 2009). Second, for different shift sizes, the optimal κ^* is different. κ^* is generally small for small shifts and large for large shifts. This is because that for small shifts, a smaller κ indicates a slower weight decay and consequently is more helpful in accumulating the OC information. In contrast, for large shifts, since their corresponding OC run lengths are usually short, a larger value of κ can put more weight on the most recent OC samples, and make the diagnostic procedure more sensitive and accurate to capture the OC shifts in a short period.

4 Case Studies

In this section, we use real-world data sets to illustrate the application of RPSR. The first set is from the handwritten numerical identification as shown in Section 4.1, and the second data set is from the

Table 5: TPP of the diagnostic frameworks with $\kappa = 0.025$.

		$\mathcal{N}_{100}$		$t_{100,5}$		$\mathrm{Gam}_{100,3}$	
	κ	SRLS	LASSO	SRLS	LASSO	SRLS	LASSO
δ	0.250	0.250	0.144	0.189	0.105	0.187	0.084
	0.5	0.446	0.431	0.383	0.361	0.193	0.142
	1	0.569	0.528	0.517	0.485	0.287	0.188
	2	0.660	0.589	0.568	0.555	0.385	0.295
	4	0.893	0.657	0.705	0.623	0.565	0.300

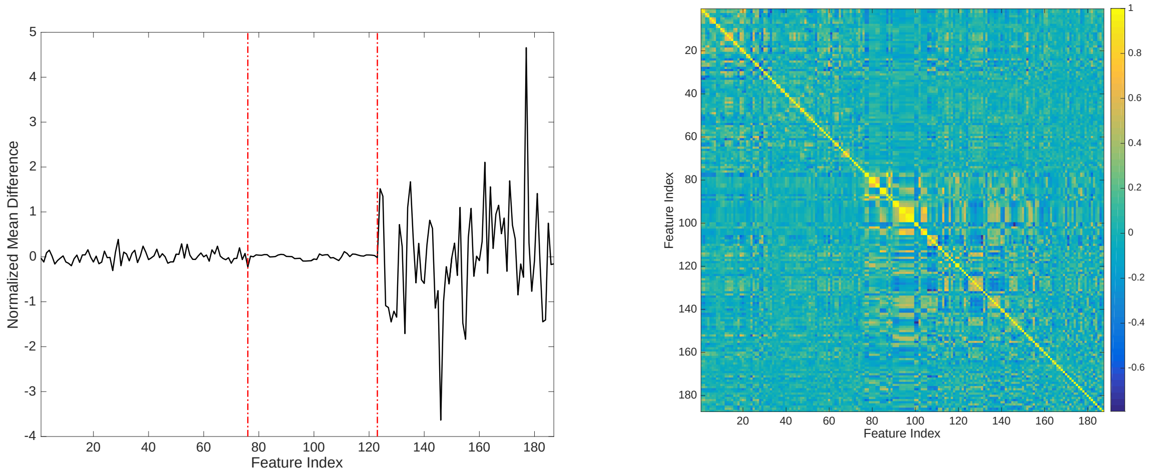
Table 6: TPP of the diagnostic frameworks with $\kappa = 0.1$.

		$\mathcal{N}_{100}$		$t_{100,5}$		$\mathrm{Gam}_{100,3}$	
	κ	SRLS	LASSO	SRLS	LASSO	SRLS	LASSO
δ	0.25	0.146	0.127	0.105	0.109	0.070	0.035
	0.5	0.326	0.291	0.210	0.199	0.105	0.102
	1	0.602	0.502	0.493	0.445	0.210	0.183
	2	0.741	0.652	0.690	0.605	0.410	0.412
	4	0.932	0.757	0.891	0.751	0.650	0.464

semiconductor manufacturing process as shown in Section 4.2.

4.1 Handwritten numeral identification

The data set consists of hundreds of features for handwritten numerals, “0” -“9”, extracted from a collection of Dutch utility maps (https://archive.ics.uci.edu/ml/datasets/Multiple+Features). Here we select 187 features for monitoring, including 76 features of Fourier coefficients of the character shapes, 47 features of Zemike moments, and 64 features of Karhunen-Love coefficients. Among these ten numerals, we find that “6” and “9” have high similarity in terms of Fourier coefficients and Zemike moments, but have certain difference in terms of Karhunen-Love coefficients. To better illustrate this point, for each feature, we calculate its standardized average distance between “6” and “9” based on 200 samples. As Figure 1a shows, only the third block of the features, which correspond to Karhunen-Love coefficients, have significant nonzero standardized average distances. With this information, we treat “6” as the IC state and “9” as the OC state, and design an online monitoring scenario using these 187 features (process variables) to detect the numeral change from “6” to “9”, i.e., to detect the change of the 64 Karhunen-Love coefficients in the process. We first analyze the statistical properties of these features. Figure 1b shows the correlation structure of features of “6”. It is clear that there

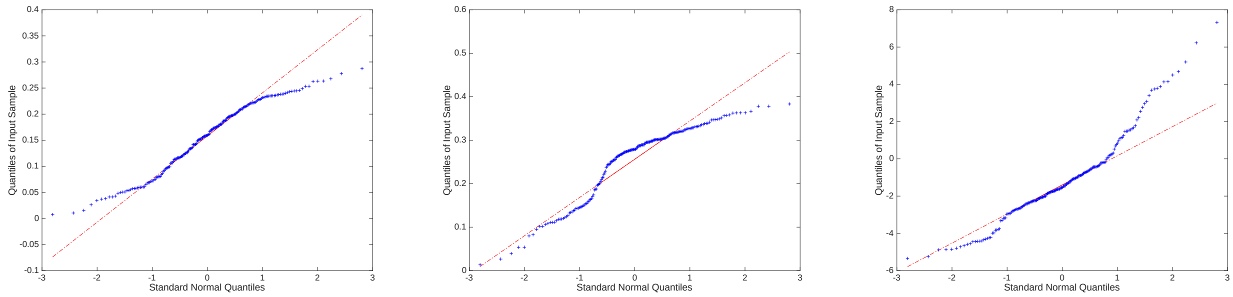


;

(a) (b)

Figure 1: (a). The difference between numeral "6" and "9" in terms of the 187 features; (b) The correlation structure of the total 187 features of "6".

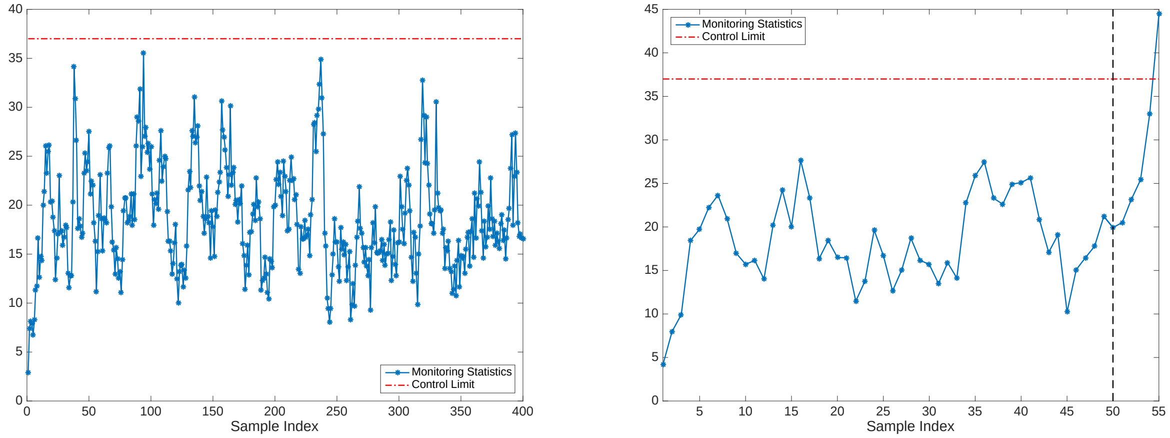
exist block-wise correlations between different features. Furthermore, the normal Q-Q plots in Figure 2 show that these variables do not have marginal normal distributions, indicating nonparametric charts might perform more robustly for this data set.



(a) $V30$ (b) $V60$ (c) $V124$

Figure 2: The normal Q-Q plots for V30, V60 and V124.

To demonstrate the application of RPSR, we design the sequential monitoring in this way: in each simulation replication, $m_0 = 100$ IC samples are randomly drawn with replacement as reference samples from the 200 IC samples of "6". Then the subsequent samples are randomly drawn with replacement from either these IC samples (for $t \leq \tau$) or the OC samples of "9" (for $t > \tau$) sequentially as online testing samples. We set RPSR parameters $k = 20$ and $S = \lfloor 187/20 \rceil = 9$, and consider two values of λ, 0.1 and 0.025, respectively. We adopt the self-starting technique for RPSR. For each λ, we use Algorithm 1 with samples generated from the multivariate normal distribution to set its control limit and ensure its IC ARL$= 200$. Figure 3 shows the resulting RPSR monitoring statistics with $\lambda = 0.1$ in one simulation replication for either the IC or OC scenario, where the dash red line represents the control limit and the blue curve connecting with stars represents the monitoring statistics. When the process is IC as Figure 3a shows, the monitoring statistics are always below the control limit, confirming the stability of the chart. When the process is OC with $\tau = 50$ as Figure 3b shows, RPSR has a quick response to the shift with a timely increase in the monitoring statistics, and finally exceeds the control limit at $t = 55$, signaling an OC alarm with run length 5. To better test the performance of RPSR, similar to Section 3, we compare its performance with the other four charts in terms of OC ARL. In particular, for MCUSUM, since the OC directions are various for these 187 variables as Figure 1a shows, we adopt the two-side CUSUM chart with the minimum interesting shift size $k = \pm 0.5$ for monitoring. The control limits of all the charts are tuned to ensure that their IC ARLs equal 200, and their OC performances are shown in Table 7. We can see that for these four charts, RPSR performs best. As to MCUSUM, RKHS and AnRank, since they fail to consider the correlation structure at all, their performances are much worse than the former two, which is consistent with the performance in Section 3.



(a) IC process

(b) OC process

Figure 3: One replication of RPSR for monitoring the handwritten numerals with $k = 20$, $S = 9$, and $\lambda = 0.1$. (a) is for the IC process; and (b) is for the OC process with change occurring at $\tau = 50$.

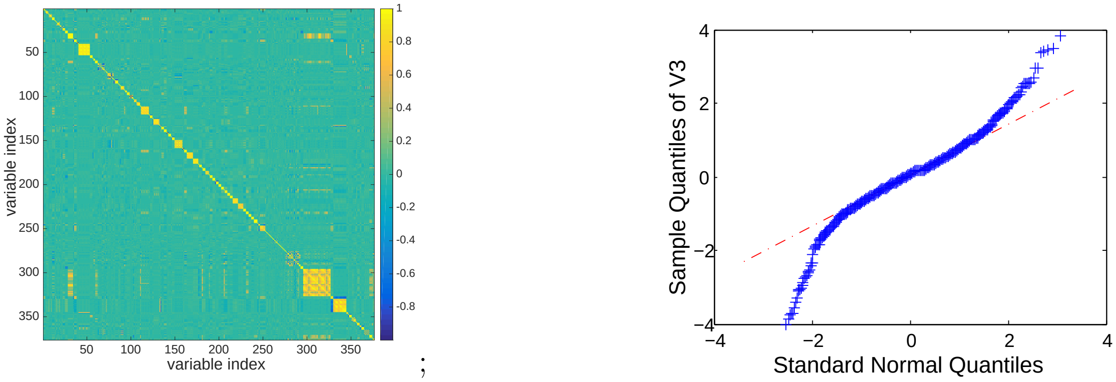
Table 7: The ARLs of different charts for monitoring the handwritten numerals with $p = 187$, $m_0 = 200$ and $\tau = 50$ (numbers in parentheses are SDRL values).

	$\lambda = 0.1$		$\lambda = 0.025$		Other methods	
	RPSR	RKHS	RPSR	RKHS	MCUSUM	AnRank
IC	196(206)	197(214)	198(204)	200(211)	197(204)	199(51.4)
OC	5.55(1.41)	95.0(95.2)	7.09(2.14)	44.5(63.1)	18.2(2.76)	69.5(12.3)

4.2 Semiconductor manufacturing

The data set contains 1,567 samples in total from a semiconductor manufacturing process (http://archive.ics.uci.edu/ml/datasets/SECOM). Each sample is a vector of 591 dimensions, consisting of 591 continuous measurements during the fabrication in producing each sample. Among them, 1,463 samples are classified as conforming ones (IC samples), while the remaining 104 samples are classified as nonconforming ones (OC samples). The goal of this section is to use this data set to illustrate the online process quality control using RPSR.

As a preprocessing step, we remove 215 variables with constant values or too many missing data from the 591 variables, in all the 1,567 samples, and remain the left 376 variables for analysis. Figure 4a shows the correlation matrix of the 376 variables, which has a block-wise structure. Some variables have quite strong correlations with each other, such as variable 300 to variable 330, while some others have no correlations. Figure 4b draws the normal Q-Q plot of one selected variable, i.e., variable 3. It is clear to see that the variable does not have marginal normal distributions. Q-Q plots of other variables also have similar features.



(a)

(b)

Figure 4: (a). The correlation structure of the 376 variables, and (b) the Q-Q plot of one selected variable in the semiconductor manufacturing process.

To demonstrate the application of RPSR, we monitor the samples sequentially in the same way as Section 4.1. In particular, in each simulation replication, m_0 samples are randomly drawn with replacement as reference samples from the 1463 IC samples, then the subsequent samples are randomly drawn with replacement from either the IC samples (for $t \leq \tau$) or the OC samples (for $t > \tau$) sequentially as online testing samples. Here we set $m_0 = 250$ and $\tau = 50$. We set RPSR parameters $k = 47$ and $S = \lfloor 376/47 \rceil = 8$. Similar to Section 4.1, we consider two values of λ, 0.1 and 0.025, respectively. We adopt the self-starting technique for RPSR. For each λ, we use Algorithm 1 with samples generated by bootstrapping the available IC data to set its control limit and ensure its IC ARL= 200. We also compare our method with RKHS, MCUSUM and AnRank, whose settings are the same as Section 4.1. The monitoring results are shown in Table 8. It is clear to see that RPSR performs the best.

Table 8: The ARLs of different charts for monitoring the semiconductor manufacturing samples with $p = 376$, $m_0 = 250$ and $\tau = 50$ (numbers in parentheses are SDRL values).

	$\lambda = 0.1$		$\lambda = 0.025$		Other methods	
	RPSR	RKHS	RPSR	RKHS	MCUSUM	AnRank
IC	200(214)	200(170)	202(181)	200(210)	200(180)	200(30.5)
OC	19.4(7.54)	36.3(7.41)	24.5(4.84)	31.1(15.7)	30.5(18.1)	90.4 (18.9)

5 Conclusions

High-dimensional data streams with dimension larger than the number of IC reference samples are very common in many applications. Usually these data streams do not follow normal distributions, and show strong between-stream correlations. These bring a lot of challenges for traditional MSPC schemes when they are applied in such high-dimensional cases. To address these challenges, this paper presents a new monitoring scheme. First, we propose to decompose the high-dimensional space into several subspaces using ensemble random projections. Since these subspaces have dimensions much smaller than the reference sample size, they can be well estimated by the reference samples. Then for every subspace, we construct a local monitoring scheme based on the spatial rank test to detect the local changes, and finally we combine the monitoring results of all the subspaces together for final decision-making. This monitoring scheme has efficient detection power for sparse process changes and robust performance for non-normal distributions. Furthermore, we propose a diagnostic framework to identify the sparse OC variables after an OC alarm is triggered. Numerical studies as well as real-data examples demonstrate the efficacy and applicability of the proposed methodology.

Acknowledgements

The authors thank the editor and two anonymous referees for their many helpful comments that have resulted in significant improvements in the article. Jianguo Wu is partially supported by the Natural Science Foundation of China under Grant 51875003. Nan Chen is partially supported by Singapore AcRF R-266-000-123-114.

References

Bai, Z. and Saranadasa, H. (1996), "Effect of high dimension: by an example of a two sample problem," *Statistica Sinica*, 311–329.

Baraniuk, R. G. (2007), "Compressive sensing [lecture notes]," *IEEE signal processing magazine*, 24, 118–121.

Belloni, A., Chernozhukov, V., and Wang, L. (2011), "Square-root lasso: pivotal recovery of sparse signals via conic programming," *Biometrika*, 98, 791–806.

Champ, C. W., Jones-Farmer, L. A., and Rigdon, S. E. (2005), "Properties of the T 2 control chart when parameters are estimated," *Technometrics*, 47, 437–445.

Chen, L. S., Paul, D., Prentice, R. L., and Wang, P. (2011), "A regularized Hotelling's T 2 test for pathway analysis in proteomic studies," *Journal of the American Statistical Association*, 106, 1345–1360.

Chen, N., Zi, X., and Zou, C. (2016), "A distribution-free multivariate control chart," *Technometrics*, 58, 448–459.

Chen, S. X., Qin, Y.-L., et al. (2010), "A two-sample test for high-dimensional data with applications to gene-set testing," *The Annals of Statistics*, 38, 808–835.

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

Ding, Y., Zeng, L., and Zhou, S. (2006), "Phase I analysis for monitoring nonlinear profiles in manufacturing processes," *Journal of Quality Technology*, 38, 199.

Guerriero, M., Willett, P., and Glaz, J. (2009), "Distributed target detection in sensor networks using scan statistics," *Signal Processing, IEEE Transactions on*, 57, 2629–2639.

Hecht-Nielsen, R. (1994), "Context vectors: general purpose approximate meaning representations self-organized from raw data," *Computational Intelligence: Imitating Life*, 43–56.

Holland, M. D. and Hawkins, D. M. (2014), "A control chart based on a nonparametric multivariate change-point model," *Journal of Quality Technology*, 46, 63.

Huang, S., Kong, Z., and Huang, W. (2014), "High-dimensional process monitoring and change point detection using embedding distributions in reproducing kernel Hilbert space," *IIE Transactions*, 46, 999–1016.

Jiang, W., Wang, K., and Tsung, F. (2012), “A variable-selection-based multivariate EWMA chart for process monitoring and diagnosis,” *Journal of Quality Technology*, 44, 209–230.

Johnson, W. B. and Lindenstrauss, J. (1984), “Extensions of Lipschitz mappings into a Hilbert space,” *Contemporary Mathematics*, 26, 1.

Jung, S., Marron, J., et al. (2009), “PCA consistency in high dimension, low sample size context,” *The Annals of Statistics*, 37, 4104–4130.

Lee, S.-P., Chao, A.-K., Tsung, F., Wong, D. S. H., Tseng, S.-T., and Jang, S.-S. (2011), “Monitoring batch processes with multiple on-off steps in semiconductor manufacturing,” *Journal of Quality Technology*, 43, 142–157.

Liu, K., Mei, Y., and Shi, J. (2015), “An Adaptive Sampling Strategy for Online High-Dimensional Process Monitoring,” *Technometrics*, 57, 305–319.

Matoušek, J. (2008), “On variants of the Johnson–Lindenstrauss lemma,” *Random Structures & Algorithms*, 33, 142–156.

McCulloh, I. A., Johnson, A. N., and Carley, K. M. (2012), “Spectral analysis of social networks to identify periodicity,” *The Journal of Mathematical Sociology*, 36, 80–96.

Megahed, F. M., Woodall, W. H., and Camelio, J. A. (2011), “A review and perspective on control charting with image data,” *Journal of Quality Technology*, 43, 83.

Mei, Y. (2010), “Efficient scalable schemes for monitoring a large number of data streams,” *Biometrika*, 97, 419–433.

Ngai, H.-M. and Zhang, J. (2001), “Multivariate cumulative sum control charts based on projection pursuit,” *Statistica Sinica*, 747–766.

Oja, H. (2010), *Multivariate nonparametric methods with R: an approach based on spatial signs and ranks*, Springer Science & Business Media.

Qiu, P. and Hawkins, D. (2003), “A nonparametric multivariate cumulative sum procedure for detecting shifts in all directions,” *Journal of the Royal Statistical Society: Series D (The Statistician)*, 52, 151–164.

Qiu, P., Zou, C., and Wang, Z. (2010), “Nonparametric profile monitoring by mixed effects modeling,” *Technometrics*, 52, 265–277.

Ranger, G. C. and Alt, F. B. (1996), “Choosing principal components for multivariate statistical process control,” *Communications in Statistics-Theory and Methods*, 25, 909–922.

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

Spiegelhalter, D., Sherlaw-Johnson, C., Bardsley, M., Blunt, I., Wood, C., and Grigg, O. (2012), “Statistical methods for healthcare regulation: rating, screening and surveillance,” *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 175, 1–47.

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

Tsung, F., Zhou, Z., and Jiang, W. (2007), "Applying manufacturing batch techniques to fraud detection with incomplete customer information," *IIE transactions*, 39, 671–680.

Wang, K. and Jiang, W. (2009), "High-dimensional process monitoring and fault isolation via variable selection," *Journal of Quality Technology*, 41, 247.

Zhang, C., Chen, N., and Zou, C. (2016), "Robust multivariate control chart based on goodness-of-fit test," *Journal of Quality Technology*, 48, 139.

Zou, C. and Qiu, P. (2009), "Multivariate statistical process control using LASSO," *Journal of the American Statistical Association*, 104, 1586–1596.

Zou, C. and Tsung, F. (2011), "A Multivariate Sign EWMA Control Chart," *Technometrics*, 53, 84–97.

Zou, C., Wang, Z., and Tsung, F. (2012), "A spatial rank-based multivariate EWMA control chart," *Naval Research Logistics*, 59, 91–110.

Appendices

Ensemble random projection generation

Here we introduce the generation procedure of the ensemble random projections $\mathbf{P}^s(s = 1, \ldots, S)$ in Algorithm 2.

Algorithm 2 Ensemble random projection matrix generation

```
For $s = 1$, generate each element of $\mathbf{P}^1 = [\mathbf{p}_1^1, \ldots, \mathbf{p}_k^1]$ from the normal distribution with mean 0 and variance $1/k$.
for $s = 2$ to $S$ do
    Find the null space of $\{\mathbf{P}^1, \ldots, \mathbf{P}^{s-1}\}$ as $\mathbf{R} \in \mathcal{R}^{p \times V}$ whose every column $\mathbf{r}_v$ is one direction (unit vector) orthogonal to any column in $\{\mathbf{P}^1, \ldots, \mathbf{P}^{s-1}\}$, with totally $V$ directions found.
    For the next projection matrix $\mathbf{P}^s = [\mathbf{p}_1^s, \ldots, \mathbf{p}_k^s]$, construct its each column as $\mathbf{p}_l^s = \sum_{v=1}^V a_{lv}\mathbf{r}_v$ for $l = 1, \ldots, k$, where $a_{lv}$ is generated from the normal distribution with mean 0 and variance $1/k$.
end for
```

In particular, here $\mathbf{P}^s(s = 1, \ldots, S)$ are generated a bit differently from the traditional algorithms. We generate S matrices sequentially by assuming $\mathbf{P}^s$ is orthogonal from the previous $\mathbf{P}^{s-1}, \ldots, \mathbf{P}^1$, while traditional generation methods do not have this constraint. However, we can still show that our generated S "not so random" random matrices can still preserve the distances after projection with high probability in the following proposition.

Proposition 1. *For the constructed ensemble projection matrices $\mathbf{P}^s(s = 1, \ldots, S)$ generated by Algorithm 2, they satisfy the Johnson and Lindenstrauss lemma. In particular, for $\epsilon \in (0, 1)$, let $k \geq$*

$C\epsilon^{-2}\log(\rho^{-1})$ where C is a large absolute constant. Then for any $\mathbf{X}_i$ and $\mathbf{X}_j(i, j = -m_0+1, \ldots, t, i \neq j)$, we have

$$\Pr\{(1-\epsilon)||\mathbf{X}_i - \mathbf{X}_j||^2 \leq ||\mathbf{P}^{s\prime}(\mathbf{X}_i - \mathbf{X}_j)||^2 \leq (1+\epsilon)||\mathbf{X}_i - \mathbf{X}_j||^2\} \geq 1-\rho. \tag{7}$$

Proof. Consider projecting a vector of the original space $\mathbf{X} \in \mathcal{R}^{p\times 1}$ to the subspace based on $\mathbf{P}^s$. Then we have $\mathbf{Y} = \mathbf{P}^{s\prime}\mathbf{X}$. With $p^s_{lj} = \sum_{v=1}^V a^s_{lv} r_{vj} (j = 1, \ldots, p)$, we have

$$Y_l = \sum_{j=1}^p p^s_{lj} X_j = \sum_j^p \sum_{v=1}^V a^s_{lv} r_{vj} X_j,$$

for $l = 1, \ldots, k$. Since $a^s_{lv} \sim \mathcal{N}(0, 1/k)$, we have

$$\begin{aligned}\mathrm{Var}[Y_l] &= \mathrm{E}[(\sum_{j=1}^p \sum_{v=1}^V a^s_{lv} r_{vj} X_j)^2] \\ &= \sum_{j=1}^p \sum_{v=1}^V \mathrm{E}[(a^s_{lv} r_{vj} X_j)^2] + \sum_{j=1}^p \sum_{v=1}^V \sum_{u=1,u\neq v}^V \mathrm{E}[(a^s_{lv} r_{vj} X_j)(a^s_{lu} r_{uj} X_j)].\end{aligned}$$

Because a_{lv} are independent with each other, we have $\mathrm{E}[a_{lv}a_{lu}] = 0$ for $v \neq u$, and $\mathrm{E}[a^2_{lv}] = 1/k$. Consequently, the second term in the above equation equals 0, and we have

$$\mathrm{Var}[Y_l] = \sum_{j=1}^p \sum_{v=1}^V \mathrm{E}[(a^s_{lv} r_{vj} X_j)^2] = \sum_{j=1}^p \sum_{v=1}^V \frac{r^2_{vj} X^2_j}{k}.$$

As such, $Y_i \sim \mathcal{N}(0, \sum_{j=1}^p \sum_{v=1}^V r^{s2}_{vj} X^2_j/k)$. Now we complete the proof using a standard Chernoff-bounding approach. In particular,

$$\begin{aligned}\Pr\{||\mathbf{Y}||^2 > (1+\epsilon)||\mathbf{X}||^2\} &= \Pr\{\exp(tk\frac{||\mathbf{Y}||^2}{||\mathbf{X}||^2}) > \exp(tk(1+\epsilon))\} \\ \text{Markov ineq.} &\leq \mathrm{E}\left[\exp(tk\frac{||\mathbf{Y}||^2}{||\mathbf{X}||^2})\right] / \exp(tk(1+\epsilon)) \\ Y_i \text{ indep.} &= \prod_{l=1}^k \mathrm{E}\left[\exp(tk\frac{Y_l^2}{\sum_{j=1}^p X_j^2})\right] / \exp(tk(1+\epsilon)),\end{aligned}$$

where

$$\frac{\sqrt{k}Y_l}{\sqrt{\sum_{j=1}^p X_j^2}} \sim \mathcal{N}(0, \frac{\sum_{j=1}^p \sum_{v=1}^V r_{vj}^2 X_j^2}{\sum_{j=1}^p X_j^2}).$$

Since $\mathbf{r}_v(v = 1, \ldots, V)$ are orthogonal projection directions with $\mathbf{r}_v'\mathbf{r}_v = 1$ and $\mathbf{r}_v'\mathbf{r}_q = 0, \forall v \neq q$, we have $\sum_{v=1}^V r_{vj}^2 \leq 1$ and

$$\frac{\sum_{j=1}^p \sum_{v=1}^V r_{vj}^2 X_j^2}{\sum_{j=1}^p X_j^2} \leq 1.$$

Therefore according to the property of the MGF of the χ^2 distribution, we have

$$\mathrm{E}\left[\exp(tk\frac{Y_l^2}{\sum_{j=1}^p X_j^2})\right] \leq \frac{1}{\sqrt{1-2t}}, \text{ for } t \leq \frac{1}{4}.$$

Therefore we have

$$\Pr\{||\mathbf{Y}||^2 > (1+\epsilon)||\mathbf{X}||^2\} \leq \left[\exp(t)\sqrt{(1-2t)}\right]^{-k} \exp(-kt\epsilon), \forall t < \frac{1}{4}.$$

Since

$$\begin{aligned} \left[\exp(t)\sqrt{(1-2t)}\right]^{-1} &= \exp(-t - \frac{1}{2}\log(1-2t)) \\ \text{Maclaurin S. } &= \exp(-t - \frac{1}{2}(-2t - \frac{(2t)^2}{2} - \ldots)) \\ &= \exp(\frac{(2t)^2}{4} + \frac{(2t)^3}{6} + \ldots) \\ &\leq \exp(t^2(1 + 2t + (2t)^2 + \ldots)) \\ &= \exp(t^2/(1-2t)), \end{aligned}$$

we have

$$\begin{aligned} \Pr\{||\mathbf{Y}||^2 > (1+\epsilon)||\mathbf{X}||^2\} &\leq \exp(kt^2/(1-2t) - kt\epsilon) \\ &\leq e^{-\epsilon^2 k/8} \end{aligned}$$

with $t = \epsilon \leq 1/4$. Similarly, we can prove $\Pr\{||\mathbf{Y}||^2 < (1-\epsilon)||\mathbf{X}||^2\} = \Pr\{-||\mathbf{Y}||^2 > (\epsilon - 1)||\mathbf{X}||^2\}$

and give the same bound. Consequently

$$\Pr\{(1-\epsilon)||\mathbf{X}||^2 \leq ||\mathbf{Y}||^2 \leq (1+\epsilon)||\mathbf{X}||^2\} \geq 1-\rho, \tag{8}$$

with $\rho = 2e^{-\epsilon^2 k/8}$. We can obtain $k = 8\epsilon^{-2}/(\log(\rho^{-1}) + \log 2)$, i.e., $k \in O(\epsilon^{-2}\log(\rho^{-1}))$.

For the $m_0 + t$ arbitrary points in $\mathcal{R}^{p\times 1}$, if we set $\rho = 1/(m_0+t)^2$, we have $k \in O(\epsilon^{-2}\log(m_0+t))$. In this way, applying union bound to (8) for all $\binom{m_0+t}{2}$ possible interpoint distances, we ensure that $\mathbf{P}^s$ embed the (m_0+t) points into k dimensions with probability that every pairwise distance is not disturbed than ϵ at least $1 - \binom{m_0+t}{2}\frac{1}{(m_0+t)^2} \geq \frac{1}{2}$. □