Computers & Industrial Engineering 70 (2014) 134–149

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

A weighted support vector machine method for control chart pattern recognition ☆

Petros Xanthopoulos *, Talayeh Razzaghi

Department of Industrial Engineering and Management Systems, University of Central Florida, 4000 Central Florida Blvd., Orlando, FL 32816, United States

ARTICLE INFO

Article history:
Received 23 May 2013
Received in revised form 17 January 2014
Accepted 23 January 2014
Available online 3 February 2014

Keywords:
Control chart
Pattern recognition
Weighted support vector machine
Classification
Imbalanced data
Quality control

ABSTRACT

Manual inspection and evaluation of quality control data is a tedious task that requires the undistracted attention of specialized personnel. On the other hand, automated monitoring of a production process is necessary, not only for real time product quality assessment, but also for potential machinery malfunction diagnosis. For this reason, control chart pattern recognition (CCPR) methods have received a lot of attention over the last two decades. Current state-of-the-art control monitoring methodology includes *K* charts which are based on support vector machines (SVM). Although *K* charts have some profound benefits, their performance deteriorate when the learning examples for the normal class greatly outnumbers the ones for the abnormal class. Such problems are termed imbalanced and represent the vast majority of the real life control pattern classification problems. Original SVM demonstrate poor performance when applied directly to these problems. In this paper, we propose the use of weighted support vector machines (WSVM) for automated process monitoring and early fault diagnosis. We show the benefits of WSVM over traditional SVM, compare them under various fault scenarios. We evaluate the proposed algorithm in binary and multi-class environments for the most popular abnormal quality control patterns as well as a real application from wafer manufacturing industry.

1. Introduction

Quality process engineering provides the necessary mathematical tools to determine whether a product satisfies some predetermined quality standards and is acceptable for market release. Statistical process control is essential for process improvement, safety assurance, and reliability analysis (Montgomery, 2007). Sometimes, quality control is also used for early diagnosis of machinery malfunctions. Sequential production of items that do not comply with the quality standards is an indicator of potential machine failure (Paté-Cornell, Lee, & Tagaras, 1987; Panagiotidou & Tagaras, 2012). In such cases, production process should stop, and the faulty part should be repaired or replaced. Thus, early detection of abnormal quality control patterns is essential for protecting expensive equipment and lowering maintenance costs. Manual quality inspection is a time consuming and tedious task often requiring attention of trained personnel. For this reason, many automated quality process monitoring methods, often termed control chart pattern recognition (CCPR) algorithms, have been proposed in the literature (Hachicha & Ghorbel, 2012).

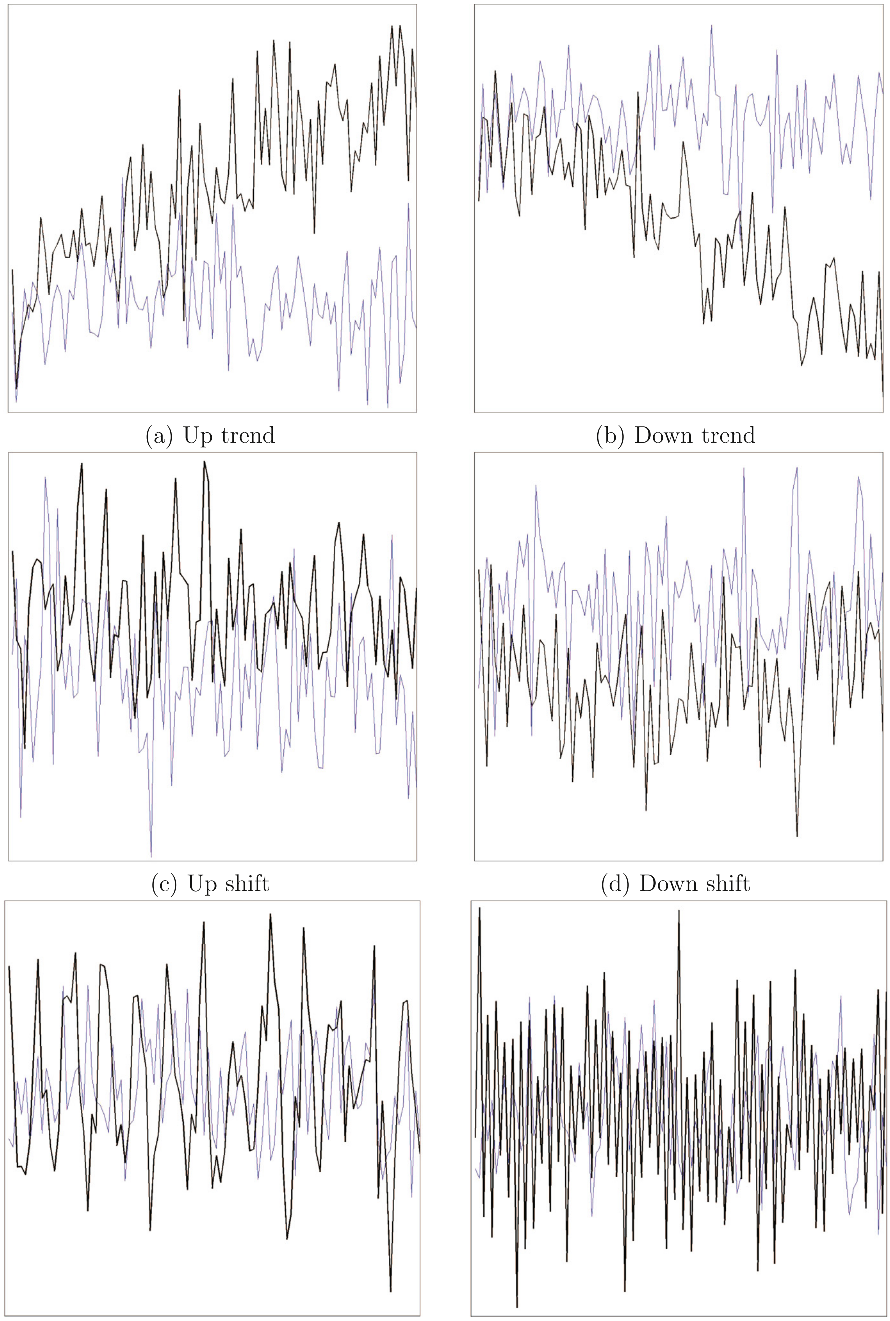
CCPR algorithms provide tools for automated detection of control patterns that demonstrate characteristics that differ significantly compared to the normal process patterns. Over the years, several abnormal patterns have been reported in real industrial problems, each of them reflecting a different underlying fault mechanism. In an early publication of Western Electric Company (1958), 15 normal and abnormal patterns are identified, one normal, seven *basic* abnormal and seven *composite* abnormal. Examples of basic patterns, namely (1) normal (N), (2) up trend (Ut) (3) down trend (Dt), (4) up shift (Us), (5) down shift (Ds), (6) cyclic (C), (7) systematic (S), and (8) stratification (F) patterns are illustrated in Fig. 1(a-g) and Fig. 2, whereas their mathematical model description can be found in Appendix A. The *composite* abnormal patterns are formed from linear combination of basic ones and they are more rare in practical applications.

Up trend and down trend patterns are associated with tool wear and malfunction in the crank case manufacturing operations (El-Midany, El-Baz, & Abd-Elwahed, 2010). The shift patterns may occur due to variations of material, machine or operator i.e. defect detection in a musical signal obtained from a broken disk or instrument (Davy, Desobry, Gretton, & Doncarli, 2006; El-Midany et al., 2010). The power supply voltage variability is often indicated

☆ This manuscript was processed by Area Editor H. Brian Hwarng.

* Corresponding author. Tel.: +1 4078235218; fax: +1 4078233413.
E-mail address: petrosx@ucf.edu (P. Xanthopoulos).

http://dx.doi.org/10.1016/j.cie.2014.01.014
0360-8352/© 2014 Elsevier Ltd. All rights reserved.

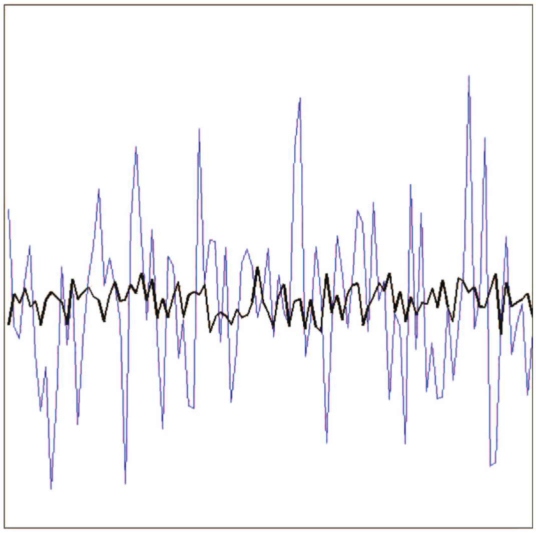


(a) Up trend (b) Down trend (c) Up shift (d) Down shift (e) Cyclic (f) Systematic

Fig. 1. Example of six abnormal patterns (bold) plotted versus an example of normal one.

by cyclic patterns (Kawamura, Chuarayapratip, & Haneyoshi, 1988). Cyclic patterns also arise in manufacturing processes, such as frozen orange juice packing (Hwarng, 1995). Jang, Yang, and Kang (2003) describe anomalies in automotive body assembly process as up/down trends, cyclic, and systematic patterns. Jin and Shi (2001) detect stamping tonnage abnormal signals by detecting up/down trend patterns. Cook and Chiu (1998) and Chinnam (2002) identify that the abnormal control chart patterns of paper making and viscosity data are up/down trend whereas, Zorriassatine, Al-Habaibeh, Parkin, Jackson, and Coy (2005) uses up trend patterns with a fault state in an end-milling process. Since each pattern uniquely characterizes a certain type of malfunction, with respect to a specific application, methods for efficient identification of abnormal patterns are necessary in order to improve fault diagnosis/repair decision making.

Early CCPR studies propose basic statistical heuristics for mean and variance shift detection (Swift, 1987). Knowledge based expert systems and artificial neural networks for CCPR were also employed in the seminal works of Hwarng and Hubele (1992), Hwarng (1995), Hwarng and Hubele (1993) and Hwarng and Hubele (1993). Other CCPR algorithms include principal component analysis (PCA) (Aparisi, 1996), time series modeling (Alwan & Roberts, 1988), regression (Mandel, 1969), and correlation analysis techniques (Al-Ghanim & Kamat, 1995; Yang & Yang, 2005). Moreover, there other are artificial intelligence-based CCPR approaches, such as the expert system (Alexander, 1987; Cheng & Hubele, 1992) and the artificial neural network (ANN) (Pugh, 1989; Cheng, 1997; Cheng & Cheng, 2008, 2009). Soft computing/data mining techniques are also used in CCPR based on the literature including clustering (Ghazanfari, Alaeddini, Niaki, & Aryanezhad,



(g) Stratification

Fig. 2. Examples of stratification abnormal pattern (bold) plotted versus an example of normal one.

2008), neurofuzzy approaches (Chang & Aw, 1996; Taylan & Darrab, 2012), fuzzy-clustering (Zarandi & Alaeddini, 2010), decision trees (Wang, Guo, Chiang, & Wong, 2008) and support vector machines (Chinnam, 2002, Sun & Tsung, 2003, Camci, Chinnam, & Ellis, 2008, Kuma, 2006, Cheng & Cheng, 2008, Issam & Mohamed, 2008, Sukchotrat, Kim, & Tsung, 2008, Sukchotra, Lu, Shao, & Li, 2009, Ra,, 2010, Lu et al., 2011). Computational studies show that the SVM based K charts perform better than PCA based T^2 charts when the data is not normally distributed. Sun and Tsung (2003) proposed the complementary use of those two control chart types based on the underlying data distribution assumption. Camci et al. (2008) proposed a robust approach for K charts along with a heuristic method for tuning the kernel parameters. The SVM based charts are based on quadratic programming and have been proved to have minimum generalization error. The classifier is obtained as an exact solution to the convex optimization problem that can be solved efficiently for large instances. These characteristics makes them a popular choice over other heuristic based classifiers (Burges, 1998; Byvatov, Fechner, Sadowski, & Schneider, 2003; Suykens, De Brabanter, Lukas, & Vandewalle, 2002).

Despite SVM's profound benefits, in the generic form it can be used only when the number of samples in each class is approximately of the same size (balanced problems). When this is not the case, the original formulation needs to be modified in order to address this special problem aspect. Imbalanced classification problems can be found in many areas, including security surveillance (Wu, Wu, Jiao, Wang, & Chang, 2003), disease diagnosis (Huang & Du, 2005), bioinformatics (Al-Shahib, Breitling, & Gilbert, 2005), geomatics (Kubat, Holte, & Matwin, 1998), telecommunications (Fawcett & Provost, 1997; Tang, Krasser, Judge, & Zhang, 2006), risk management (Ezawa, Singh, & Norton, 1996; Groth & Muntermann, 2011), manufacturing (Adam et al., 2011), quality estimation (Lee, Song, Song, & Yoon, 2005; Suresh, Venkatesh Babu, & Kim, 2009), and power systems (Hu, Zhu, & Ren, 2008).

Preprocessing methods and modification of generic classification schemes have been proposed in order to address these problems. Among the most popular preprocessing methods are the *resampling* techniques. Under these methods two major strategies can be identified, *undersampling* and *oversampling*. In these cases data points are added/removed from the dataset in order to create a balanced classification problem. The latter can be solved with any generic classifier for balanced problems (Chawala, Bower, Hall, & Kegelmeyer, 2002; Estabrooks, Jo, & Japkowicz, 2004, 2005; Akbani, Kwek, & Japkowicz, 2004; Tang, Zhang, Chawla, & Krasser, 2009; Liu, Wu, & Zhou, 2009). One issue with this approach is that the resulting classification problem can be biased, and in the particular case of undersampling it is possible to lose important information due to data removal. Another approach is the *cost sensitive learning* which in general assigns different importance to each data class and solves the weighted classification problem. The SVM adaptation to the cost sensitive learning framework is termed WSVM (also known as Fuzzy SVM in some works) which was originally proposed by Lin and Wang (2002) and further applied and studied in subsequent works (Fan & Ramamohanarao, 2005, Bao, Liu, Guo, & Wang, 2005, Hua, Hwang, Park, & Kim, 2005, Hwa,, 2011, Zhang et al., 2012). Their advantage is that the cost coefficient is directly factored into the SVM problem providing an exact optimal solution.

However, to date, to the best of authors knowledge, there is no CCPR algorithm that takes into consideration the imbalanced nature of the abnormal pattern detection problems. Furthermore, there have not been sufficient computational studies that evaluate the applicability of generic SVM in highly imbalanced environments. By nature, K charts assign equal importance to all data samples. This is not necessarily optimal and might give poor classification performance. In the present paper, we propose a WSVM approach for the abnormal quality control detection problems, compare against the traditional SVM, and demonstrate its benefits for highly imbalanced problems. For this objective, one or more abnormal patterns are examined and the proposed method and near optimal parameters of WSVM are presented. The remainder of this article is organized as follows. In Section 2, the SVM and WSVM methodologies are described along with the performance evaluation measures. In Section 3, computational results and comparisons are presented. In Section 4, we provide conclusions and future directions for future research.

2. Materials and methods

2.1. Support vector machines

The SVM is a popular supervised learning algorithm originally proposed by Vapnik (2000). It has been used in many real-world problems such as text categorization (Joachims, 1998; Pilászy, 2005), image classification (Chapelle, Haffner, & Vapnik, 1999; Foody & Mathur, 2004), bioinformatics (including protein classification and cancer classification) (Leslie, Eskin, & Noble, 2002; Zavaljevski, Stevens, & Reifman, 2002; Guyon, Weston, Barnhill, & Vapnik, 2002) and hand-written character recognition (Bahlmann, Haasdonk, & Burkhardt, 2002). Originally, the SVM is designed to solve binary classification problems, but multi-class extensions are also available.

Moreover, the SVM is developed to solve linearly separable problems only. However, it is possible to generalize them in order to classify non-linear problems by employing the *kernel trick* (Cristianini & Shawe-Taylor, 2000). The intuition behind *kernel trick* is that the original data points are mapped into a higher dimensional feature space in which they can be separated by a linear classifier. The projection of a linear classifier on the feature space is a non-linear one in the original (input) space.

Assume that a dataset is represented by a set $\mathcal{J} = \{(x_i, y_i)\}_{i=1}^{l}$ where $(x_i, y_i) \in \mathbb{R}^{n+1}$, l and n are the number of samples and features, respectively, and each x_i is a sample with n features and a class label $y_i \in \{-1, 1\}$. The SVM classifies the data points by identifying a separating hyperplane whose distance is maximum with respect to the data points of each class. The separation hyperplane defined by the parameters w and b can be obtained by solving the following convex optimization problem (Cortes & Vapnik, 1995):

$$\min \ \frac{1}{2}\ \|w\|^2 \tag{1a}$$

$$\text{s.t. } y_i(w^T\phi(x_i)+b) \geqslant 1 \quad i=1,\dots,l \tag{1b}$$

where ϕ is the kernel function $\phi:\mathbb{R}^n \to \mathbb{R}^m$ where $m \geqslant n$, i.e. each training sample x_i is mapped into a higher dimensional space by the function ϕ. For linear SVM, we have $\phi(x_i)=x_i$. Then, the class y_u of an arbitrary unknown point x_u is assigned based on the following rule:

$$y_u = sgn\{w^T\phi(x_u)+b\}, \tag{2}$$

where $sgn\{\cdot\}$ is the sign function. This formulation is known as *hard margin* SVM because it requires that the two classes to be separable through a classification hyperplane. If the classification problem is non-separable, then Problem (1) is infeasible. In this case, slack variables $\xi_i,\ i\in\{1,\dots,l\}$ are added to the objective function whose goal is to allow but penalize misclassified points. This approach is known as *soft margin* SVM and the corresponding quadratic programming problem can be formulated as (Cortes & Vapnik, 1995):

$$\min \ \frac{1}{2}\ \|w\|^2 + C\sum_{i=1}^{l}\xi_i \tag{3a}$$

$$\text{s.t. } y_i(w^T\phi(x_i)+b) \geqslant 1-\xi_i \quad i=1,\dots,l \tag{3b}$$

$$\xi_i \geqslant 0 \quad i=1,\dots,l \tag{3c}$$

The parameter C controls the magnitude of penalization. The soft margin formulation converges to the hard margin as $C\to+\infty$. Many algorithms, such as sequential minimal optimization (SMO), operate on the Lagrangian dual problem instead of Problem (3) for faster and more stable convergence. The Lagrangian dual of (3) will be:

$$\max \sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j K(x_i,x_j) \tag{4a}$$

$$\text{s.t. } \sum_{j=1}^{l}\alpha_i y_i = 0 \quad i=1,\dots,l \tag{4b}$$

$$0 \leqslant \alpha_i \leqslant C \quad i=1,\dots,l \tag{4c}$$

where $K(x_i,x_j)=\phi(x_i)^T\phi(x_j)$ is the kernel function that measures the similarity between two arbitrary points. However, through this formulation all data points are given the same importance in the training process. This might not be desirable especially in the case that one class contains outliers or in the case that one class contains considerably less point than the other. For this reason, a modified version of soft margin SVM has been proposed for making the training process more flexible. Suppose we are given a set of labeled samples with corresponding weights represented by the set $\mathcal{T}'=\{(x_i,y_i,s_i)\}_{i=1}^{l}$. Each training sample $(x_i,y_i)\in\mathbb{R}^{n+1}$ is associated to a given label $y_i\in\{-1,1\}$ and the corresponding weight $0\leqslant s_i\leqslant 1$ with $i=1,\dots,l$. The optimal hyperplane is again identified from the solution of the optimization problem (Lin & Wang, 2002).

$$\min \ \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}s_i\xi_i \tag{5a}$$

$$\text{s.t. } y_i(w^T\phi(x_i)+b) \geqslant 1-\xi_i \quad i=1,\dots,l \tag{5b}$$

$$\xi_i \geqslant 0 \quad i=1,\dots,l \tag{5c}$$

Through WSVM one can assign different weights to each data sample based on a predetermined importance measure. This provides a more flexible scheme compared to SVM where the overall penalization magnitude C is the only parameter. For the special case of imbalanced binary classification, Veropoulos, Campbell, and Cristianini (1999) proposed the usage of different costs associated with the positive (C^+) and negative (C^-) class

$$\min \ \frac{1}{2}\ \|w\|^2 + C^+\sum_{\{i|y_i=+1\}}^{n_+}\xi_i + C^-\sum_{\{j|y_j=-1\}}^{n_-}\xi_j \tag{6a}$$

$$\text{s.t. } y_i(w^T\phi(x_i)+b) \geqslant 1-\xi_i \quad i=1,\dots,l \tag{6b}$$

$$\xi_i \geqslant 0 \quad i=1,\dots,l \tag{6c}$$

The Problem (6)to find the optimal hyperplane is a Quadratic programming problem, which can transformed into the Lagrangian dual with the Kuhn–Tucker conditions. The Lagrangian dual is given by:

$$\max \sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j K(x_i,x_j) \tag{7a}$$

$$\text{s.t. } \sum_{j=1}^{l}\alpha_i y_i = 0 \quad i=1,\dots,l \tag{7b}$$

$$0 \leqslant \alpha_i \leqslant C^+ \quad \text{if } y_i=+1 \text{ and } i=1,\dots,l \tag{7c}$$

$$0 \leqslant \alpha_i \leqslant C^- \quad \text{if } y_i=-1 \text{ and } i=1,\dots,l \tag{7d}$$

The role of the weighting parameters C^+ and C^- is to assign different "importance" to the misclassification of the positive and negative class. In this way the minority class becomes more important in terms of objective function value. As it is shown in Fig. 3, WSVM classifies correctly minority class examples.

2.2. Performance measures

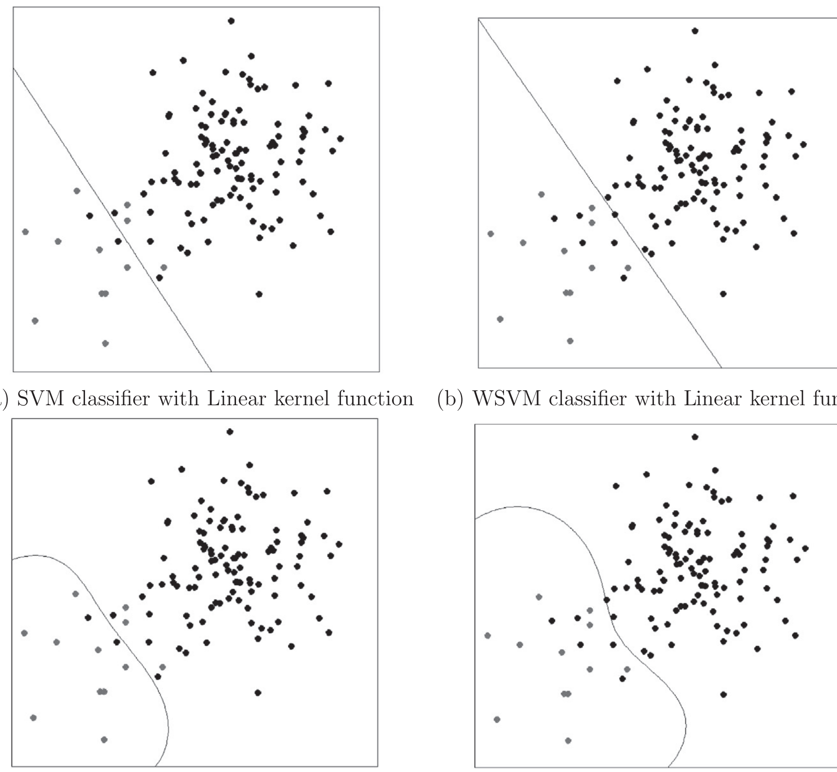
In this study we employed two types of measures for evaluating the performance of proposed algorithms: (a) *data mining based* measures especially tailored for imbalanced problems and (b) *Average Run Length (ARL)* based measures that is used to detect how fast, in terms of abnormal samples, an algorithm is able to detect the anomalous patterns.

2.2.1. Data mining based measures

Different performance measures can be used for imbalanced data classifications. Such measures can be obtained, directly or indirectly, from the classification *confusion matrix* (see Table 1). For a classification problem with k classes, the *confusion matrix* is a square matrix $C\in\mathbb{R}^{k\times k}$, with each of its entries c_{ij}, denoting the percentage of the samples that belong to the class i and classified to the class j. For the special case of binary classification (positive and negative), the confusion matrix is as follows: where TP, FP, FN, TN stand for true positives, false positives, false negatives and true negatives correspondingly. Obviously, the higher the values of the main diagonal the better the classifier is. A common performance measure for classification is the so-called *accuracy*, which is expressed as the correctly classified samples over the total number of training samples. However, for imbalanced classification problems this might not be a good performance indicator, since the majority class dominates the behavior of this metric. More specifically, naive decision rules can yield high classification accuracy. For example, the rule *"Assign all data point to the positive (majority) class"* will yield 95% classification accuracy in an imbalanced problem where 95% of the data belong to the positive class and 5% to the negative. Alternatively, sensitivity and specificity can be used. They are defined as

$$\text{Sensitivity} = \frac{TP}{TP+FN}, \quad \text{Specificity} = \frac{TN}{TN+FP}. \tag{8}$$

For the previous "toy" example, the discussed naive classifier would have 95% sensitivity and 0% specificity. Still, sensitivity is manipulated by the majority (positive) class. However, the



(a) SVM classifier with Linear kernel function (b) WSVM classifier with Linear kernel function

(c) SVM classifier with RBF kernel function (d) WSVM classifier with RBF kernel function

Fig. 3. Imbalanced data classification with linear (a and b) and RBF (c and d) kernel functions: the SVM classifier (left) will classify any sample to the majority class, the WSVM (right) with assigning different weights to each class. The decision boundaries are shifted as a result of high weighting coefficient assigned to the minority class.

Table 1
Confusion matrix for binary classification problem.

	Positive class	Negative class
Positive class	TP	FP
Negative class	FN	TN

specificity is not and therefore, it is a more appropriate measure for this purpose. The space spanned by sensitivity and specificity is termed Receiver Operator Characteristic (ROC) space. The ROC space provides a visual representation of classifiers. Another metric often used is the geometric mean of sensitivity and specificity (often abbreviated *G-mean*) which is defined as the square root of the product between sensitivity and specificity. In this study, sensitivity, specificity and *G-mean* are used as performance measures.

2.2.2. Average Run Length (ARL) based measures

In addition to data mining based evaluation we employed ARL based measures. The ARL many "faulty samples" does a process need to produce, on average, in order to make sure that an anomaly has been detected. In the CCPR framework we use the Average Target Pattern Run Length (ATPRL) (Hwarng & Hubele, 1991) and consists of the average number of samples needed for discovering an abnormal pattern. Since ATPRL can only be computed for discovered abnormal patterns one needs to take into account the rate of abnormal pattern discovery. For this, here we use the *Average Run Length Index* (ARLIDX) (Hwarng & Hubele, 1991) which equals to the fraction of ATPRL divided by the discovery rate of abnormal patterns. It is worth noting that when classification accuracy equal 100% the two measures, ATPRL and ARLIDX, are equivalent. The ARL based measures are important especially for applications where the production of each sample is cost and labor intensive. Ultimately one wants to detect an anomaly with the lower ATPRL possible.

3. Results and discussion

In this section, we present experimental results between SVM and the proposed WSVM. Experiments on both SVM and WSVM were conducted with LIBSVM-3.12 and LIBSVM-weights-3.12 (Chang & Lin, 2011). The LIBSVM was interfaced in MATLAB and the rest of the script was developed in it as well. All experiments are performed on an Intel core i5, 2.3 GHz with 4 Gb of RAM in a 64-bit platform. For each classification problem, we generate a total of 1000 data points and for cross validation purposes, 90% of the data was used for training and the rest 10% was used for testing. All data are normalized prior to classification, so that they have zero mean and unitary standard deviation (*zscore()* function in MATLAB was used). Data based on different normal and abnormal patterns are generated (for the mathematical models see Appendix A).

However, there are several kernel functions in the literature for this particular problem, we chose the *Radial Basis Function* (RBF) kernel. This is because the natural distribution of the classes is either normally distributed (for control data) and very close to normally distributed (for abnormal data). The RBF kernel is effective in producing spherical and ellipsoidal decision areas which makes it an appropriate kernel candidate for the problem under consideration. For this kernel function, the similarity between two data points x_i and x_j is given by:

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \quad \gamma \geqslant 0. \tag{9}$$

For each class, the weights are estimated as the inverse of the class size:

$$C^+ = \frac{C}{n^+}, \quad C^- = \frac{C}{n^-} \tag{10}$$

where n^+ and n^- are the size of normal and abnormal class, and C^+ and C^- are weights corresponding to the normal and abnormal classes respectively. Note that for balanced problems the weights become equal ($C^+ = C^-$) and the algorithm reduces to SVM. This weight strategy has been employed in a number of previous studies (Liu, Jia, & Ma, 2005; Du & Chen, 2005; Huang & Du, 2005; Hwang et al., 2011). We studied the behavior of our classifiers for different values of abnormal trend pattern values as well as for different window lengths w (different number of features). Our goal is to study and implement WSVM and SVM for CCPR and at the same time identify the set of parameters that generate the most challenging problems and also determine what is the minimum number of features (minimum w) for which the classifier is trusted. In some sense parameter w is related to the *Average Run Length (ARL)* since it shows how much data do we need in order for the abnormal patterns to be discoverable. In particular, the following experiments are conducted:

1. We compare WSVM against SVM for each abnormal pattern and for different window lengths w and different pattern parameters with respect to *G-mean*. Through this experiment, we identify the parameter values for which, (a) the problems easily solvable for both algorithms, (b) the problems cannot be solved by neither algorithm and (c) the rest (partially separable problems). For this test we consider highly imbalanced problems where 97.5% of the data belong to the normal class and only 2.5% belong to the abnormal.
2. For selected separable (Sep), partially separable (Ps) and inseparable (Is) problems, we perform a statistical test the performance between SVM and WSM. We observe that although the mechanism of generation of the various classes is different, the statistical test failed to reject the null hypothesis except for the case of separable problems where both algorithms perform equally well.
3. For selected instances of (Sep), (Ps) and (Is) problems, we perform detailed and *G-mean* analysis for different types of imbalanced problems. We observe that WSVM performance is highly robust for difference imbalanced instances whereas SVM highly depends on the imbalance ratio. The results are consistent with earlier imbalanced classification literature.
4. We conduct a comparison in terms of ARLIDX between WSVM and SVM for all the patterns and for different abnormal trend parameter values. We observe that for all the instance considered for a sliding window of 10 samples WSVM obtain lower ARLIDX values compared to SVM. For this experiment algorithms were evaluated for highly imbalanced (challenging) classification problems. Additional experiments for other window values can be found in the Supplementary material of this paper.
5. We measure the time needed for training and testing of the proposed algorithms in order to determine whether this scheme can be of practical usefulness. In addition to evaluation through G-means and ARLIDX, it is important to understand whether there is a computational bottleneck. It turns out that the proposed algorithms can handle large amounts of data in reasonable amount of time.
6. We perform a multi-class comparison between multiclass-WSVM and multiclass-SVM with a total of seven classes (one normal and six abnormal). Results are compared in terms of the *confusion matrix* of each classifier. Computational running time for training and testing is reported as well. In addition we performed a ARLIDX computational comparison.
7. We apply our proposed method to a real time series imbalanced classification problem from wafer manufacturing industry. The problem is imbalanced by nature and includes two classes (normal and abnormal). The abnormal patterns have characteristics that involve *shift* and *trend* changes.

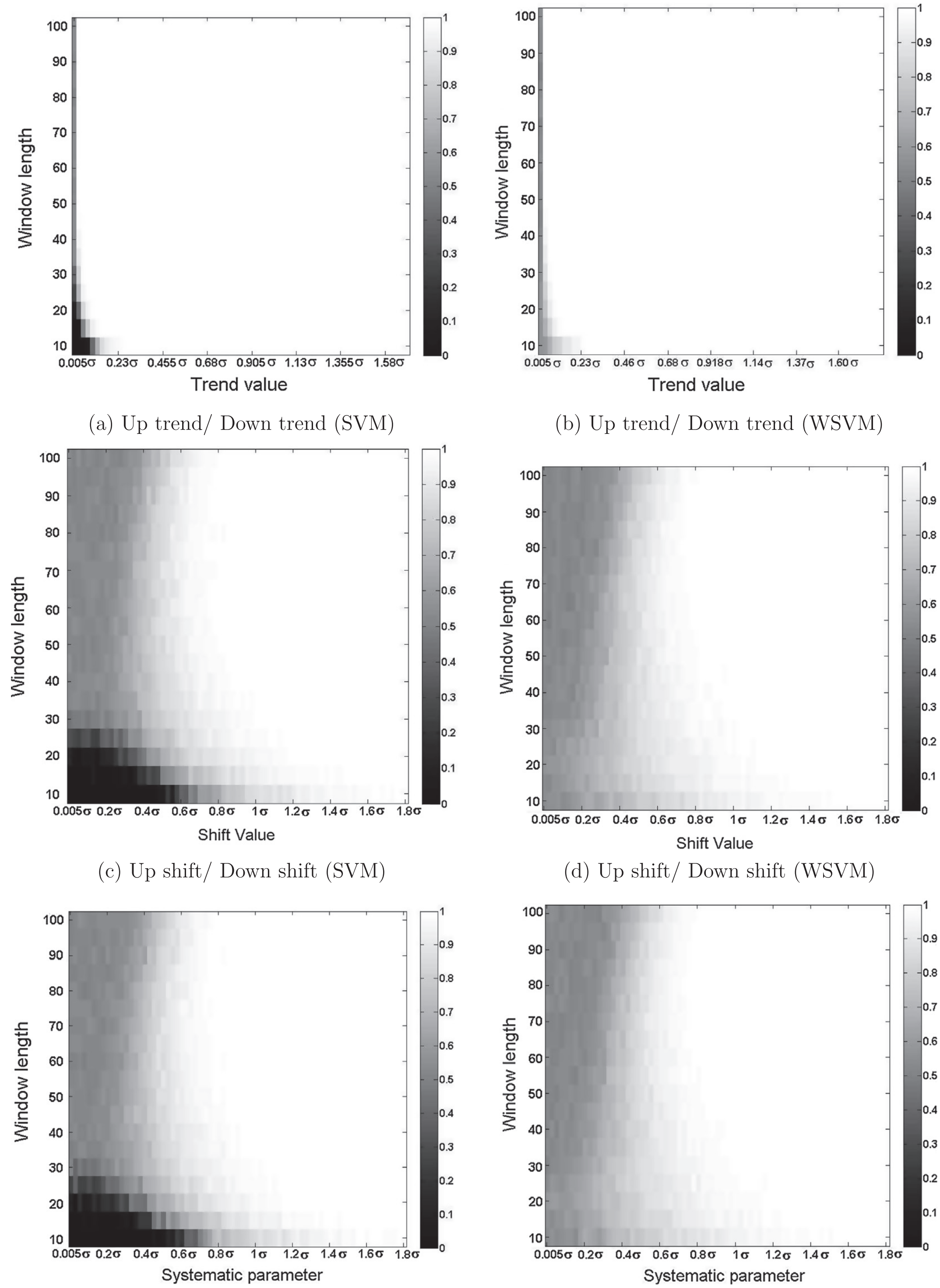
Table 2
Summary of parameter range for computational experiments.

Name	Symbol	Range
Window length	w	$[10, 100]$
Process mean (all patterns)	μ	0
Standard deviation of normal process	σ	1
Slope (up/down trend pattern)	λ	$[0.005\sigma, 0.605\sigma]$
Shift (up/down shift pattern)	ω	$[0.005\sigma, 1.805\sigma]$
Standard deviation (stratification pattern)	$\acute{\varepsilon}_t$	$[0.005\sigma, 0.8\sigma]$
Cyclic parameter (cyclic pattern)	α	$[0.005\sigma, 1.805\sigma]$
Systematic parameter (systematic pattern)	k	$[0.005\sigma, 1.805\sigma]$

The parameter selection C and γ for SVM and WSVM was performed through a uniform grid search over the parameter space. The two classification schemes (SVM and WSVM) were compared in terms of their *G-mean* for different values of the window w and abnormal parameter values Table 2.

We consider a wide range of window lengths (w) and a wide range of abnormal pattern parameters. In general higher w yields less challenging problems however this requires more data since w is the time window of the process. The results are shown in Figs. 4 and 5. The tables with the all the numerical values along with the sensitivity, specificity and classification accuracy for these experiments can be found in the Supplementary material section of this paper. On the other side higher deviation from normal data yields also to easier problems as expected. Ultimately one would like to detect slide parameter shifts with the smallest w possible. It turns out that for these instances WSVM provides a considerable improvement over SVM. We observe that overall in each pattern there are some problems that both SVM and WSVM can solve (white areas), some problems that are partially solvable (gray areas) and some problems that are inseparable (black areas). We can see that WSVM, for most of the cases can solve better the instances where SVM cannot solve. These are typically the ones with low w and low pattern parameter. We can also see that for small pattern parameter changes problems remain partially solvable for both classifiers. This means that the problem remains challenging for small parameter shifts. However such shifts are less likely to cause a severe malfunction. Overall we see that WSVM performs the same or better compared to SVM and further more its behavior is more robust since the performance does not change dramatically with the changes of the parameters. In addition, we observe a symmetric behavior of the algorithms for symmetric trend and shift patterns. In practice, the choice of w depends on the nature of the imbalanced problem as well as the magnitude of parameter shift that one wishes to detect.

Next we compare WSVM and SVM in terms of ARLIDX. Results for various parameter values are shown in Fig. 4. All the patterns were anchored to a common ARLIDXin value that is included in the first line of the table. Anchoring to a common value was achieved by appropriately tuning the parameters of the algorithm. That is we first adjusted the penalization coefficients C (for SVM) and C^+ and C^- for WSVM to a value for which both schemes have the ARLIDX is the same for normal patterns (ARLIDXin). The values of these parameters are shown in Table 3. Then we used these



(a) Up trend/ Down trend (SVM)

(b) Up trend/ Down trend (WSVM)

(c) Up shift/ Down shift (SVM)

(d) Up shift/ Down shift (WSVM)

(e) Systematic (SVM)

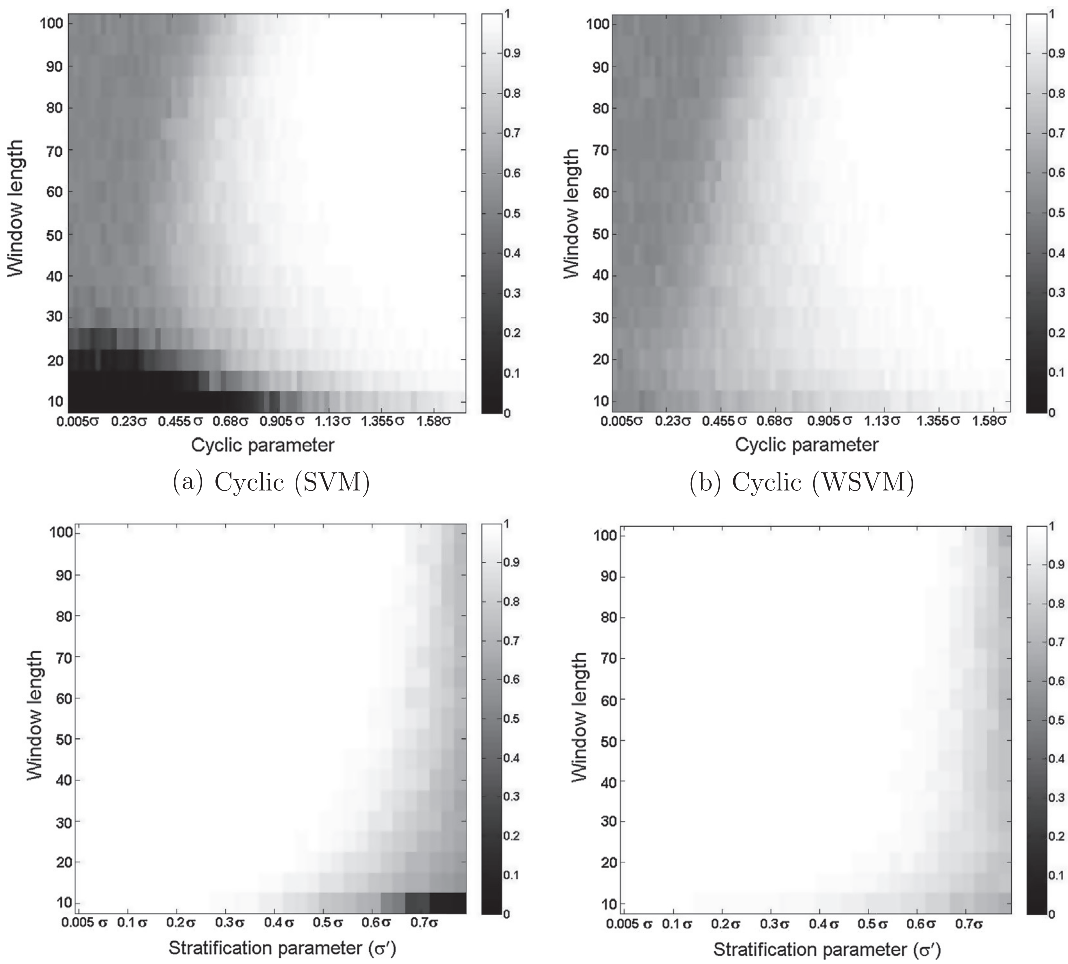
(f) Systematic (WSVM)

Fig. 4. Geometric mean of sensitivity for different parameters window lengths and patterns for highly imbalanced data.

parameter values for comparing the two schemes in terms of abnormal pattern ARLIDX.

We observe that for all patterns the majority of problem instances WSVM is lower (noted with boldface in Table 4). We observe that both detection schemes have considerable lower ARLIDX rate compared to the ARLIDXin. This is desirable according to Hwarng and Hubele (1992).

Another necessary aspect to consider is the computational running time of the algorithm. In a practical setting the training phase can occur off-line on the historical data and only the testing or prediction phase will be conducted on-line to generated data. However, if computational time allows, it would be beneficial to retrain the algorithm on-line as new data are generated real time. We recorded the time required for training and testing as a func-



(a) Cyclic (SVM) (b) Cyclic (WSVM)

(a) Stratification (SVM) (b) Stratification(WSVM)

Fig. 5. Boundary obtained for inseparable, partially separable, and separable classification problems for cyclic and stratification patterns.

Table 3
Parameters for anchoring ARLDX-in.

Uptrend			Upshift			Systematic			Cyclic			Stratification		
SVM	WSVM		SVM	WSVM		SVM	WSVM		SVM	WSVM		SVM	WSVM	
C	C^+	C^-	C	C^+	C^-	C	C^+	C^-	C	C^+	C^-	C	C^+	C^-
955.426	20.000	2.105	117.500	2.000	0.008	100.000	2.000	0.211	100.000	2.000	0.158	50.000	20.000	10.526

tion of the data size (Fig. 7) and and abnormal pattern parameter (Fig. 6 and Table 8). As expected the training and testing time for a fixed data size is negligible (order $\sim 10^{-2}$ s) compared to the order of training ($\sim$ 10 s) and the order of testing ($\sim$ 1 s) (see Table 5). It is noteworthy that these computational times are based on a nonembedded implementation developed for research in MATLAB environment. A potential industrial embedded implementation will propably have much lower computational times. Therefore the times reported in Figs. 7 and 6 and Table 8 should be seen as upper bound for a real time implementation.

Next we select some representative problems from (Sep), (Ps) and (Is) classes and compare them with respect to *accuracy*, *sensitivity*, *specificity* and *G-means*. Sensitivity and accuracy are greatly driven by the majority class and specificity is mostly affected by the correct classification of the minority class. However, *G-mean* is a measure that considers both sensitivity and specificity and can be a trusted performance metric of imbalanced classification. We can see that *G-mean* consistently improves for all the problems and all the patterns under consideration (Table 6). We perform a statistical test in order to measure the significance of WSVM improvement. Our null hypothesis is that the *G-mean* of SVM and WSVM are equal and the alternative hypothesis that they are not. The *t*-test rejects the null hypothesis in most of the instances except for the separable cases where both SVM and WSVM perform equally well (less challenging instances). We note that in many cases SVM achieves high sensitivity and zero specificity. This implies *null classification* meaning that all the test samples are classified in the same class and is indicative of poor performance. This has been noted in previous studies (Weiss, 2004) and it is in accordance with previous computational results (Anand, Pugalenthi, Fogel, & Suganthan, 2010; Hwang et al., 2011).

Next we compare the performance of SVM and WSVM for different imbalanced ratios and representative problems of different difficulty level ((Sep), (Ps) and (Is)). Parameters for each problem are the same as these in Table 6. Results are shown in Tables 7–9. We consider the size of imbalanced normal and abnormal data as $(50 + r)\%$ and $(50 - r)\%$ respectively. For all instances, and regardless the difficulty level of the problem we can see that

Table 4
ARLIDX for abnormal patterns. With boldface is the lower value between the two schemes.

Parameter	Uptrend		Upshift		Systematic		Cyclic		Stratification	
	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM
0	155.54	155.19	155.65	155.06	155.21	155.19	155.92	155.83	155.28	155.22
0.005	16.33	**14.60**	138.41	**111.11**	110.83	**90.91**	78.51	**66.85**	7.90	**7.36**
0.03	13.04	**9.83**	129.87	**96.67**	106.90	**70.87**	**56.12**	56.64	8.00	**7.49**
0.055	11.34	**6.64**	**81.49**	81.75	94.95	**67.39**	75.76	**67.50**	8.03	**7.46**
0.08	9.46	**6.61**	103.12	**66.67**	80.73	**53.62**	**51.37**	63.33	8.05	**7.47**
0.105	8.37	**7.28**	106.51	**59.09**	80.36	**44.00**	67.11	**62.50**	7.96	**7.47**
0.13	8.09	**7.84**	90.50	**51.73**	62.92	**53.36**	67.40	**54.04**	7.99	**7.53**
0.155	7.50	**6.96**	70.50	**53.83**	74.21	**26.98**	69.64	**66.67**	8.05	**7.43**
0.18	7.10	**6.96**	73.83	**33.55**	59.54	**25.72**	67.94	**42.83**	8.10	**7.37**
0.205	6.81	**6.24**	62.91	**24.07**	67.33	**22.34**	53.69	**44.60**	7.91	**7.50**
0.23	7.47	**7.12**	76.66	**20.64**	77.87	**10.29**	69.44	**38.27**	8.20	**7.44**
0.255	7.01	**6.89**	37.44	**19.59**	62.76	**10.34**	59.36	**26.54**	8.11	**7.57**
0.28	7.66	**6.97**	40.74	**6.99**	79.42	**9.38**	69.61	**21.10**	8.08	**7.54**
0.305	7.01	**6.75**	49.65	**7.62**	52.66	**4.89**	75.76	**12.86**	8.16	**7.46**
0.33	7.27	**6.94**	60.99	**6.82**	36.13	**5.16**	59.33	**11.31**	8.24	**7.68**
0.355	6.49	**6.27**	37.28	**8.05**	49.41	**6.12**	54.31	**7.20**	8.47	**7.56**
0.38	7.50	**7.37**	25.59	**6.52**	48.56	**5.39**	56.96	**5.67**	8.50	**7.77**
0.405	**6.67**	6.69	24.99	**6.57**	27.87	**4.64**	66.00	**5.69**	9.06	**7.85**
0.43	6.75	**6.69**	18.74	**6.04**	22.04	**4.96**	46.91	**6.11**	9.56	**8.60**
0.455	6.76	6.76	19.08	**6.13**	23.24	**5.39**	45.03	**7.40**	10.76	**9.52**
0.48	6.50	**6.48**	15.13	**5.59**	25.15	**5.11**	42.57	**6.45**	37.64	**24.38**
0.505	**6.50**	6.52	14.86	**5.86**	16.26	**5.20**	38.86	**7.29**	**16.39**	16.44
0.53	6.85	**6.84**	13.81	**6.34**	13.82	**5.94**	45.88	**7.08**	40.81	**34.74**
0.555	6.69	6.69	13.33	**6.40**	15.65	**5.84**	44.54	**6.39**	47.17	**32.24**
0.58	6.51	6.51	12.59	**5.44**	22.55	**4.75**	40.28	**5.57**	72.46	**32.87**
0.605	**6.61**	6.65	16.58	**6.20**	20.11	**6.02**	37.30	**6.43**		
0.63	6.44	**6.43**	11.40	**5.85**	13.22	**5.61**	39.77	**6.83**		
0.655	**6.45**	6.50	13.81	**5.75**	12.36	**5.16**	31.47	**6.24**		
0.68	**6.21**	6.23	10.61	**5.71**	10.90	**5.48**	21.48	**6.56**		
0.705	**6.29**	6.30	12.46	**6.32**	16.04	**5.14**	18.27	**5.51**		
0.73	6.38	**6.34**	8.33	**5.89**	10.78	**5.97**	25.24	**6.21**		
0.755	6.26	**6.23**	10.09	**5.91**	10.19	**5.95**	17.92	**5.59**		
0.78	**6.18**	6.21	8.07	**5.93**	11.58	**5.71**	15.82	**7.25**		
0.805	**6.13**	6.16	12.07	**5.94**	9.16	**5.93**	17.88	**7.46**		
0.83	**5.86**	5.87	8.05	**5.71**	9.07	**6.37**	15.13	**5.32**		
0.855	6.19	6.19	8.11	**5.23**	8.17	**5.96**	17.40	**7.20**		
0.88	6.25	**6.17**	8.62	**5.50**	8.80	**6.02**	13.86	**6.88**		
0.905	**6.02**	5.95	8.38	**6.14**	8.69	**6.64**	11.96	**7.85**		
0.93	6.21	**6.20**	7.09	**5.73**	8.47	**6.48**	12.24	**6.00**		
0.955	**6.08**	6.11	7.50	**5.94**	9.47	**6.37**	10.75	**6.77**		
0.98	6.08	**6.04**	7.86	**5.59**	8.60	**5.85**	11.50	**6.67**		
1.005	6.07	**6.05**	6.74	**5.79**	8.54	**6.41**	10.48	**6.02**		
1.03	5.97	**5.93**	6.74	**6.05**	8.08	**6.41**	13.60	**7.27**		
1.055	5.98	**5.94**	7.70	**6.63**	7.84	**5.97**	15.04	**7.72**		
1.08	**6.11**	6.14	6.78	**6.31**	8.49	**6.45**	15.72	**6.92**		
1.105	6.07	6.07	6.64	**5.56**	7.92	**6.38**	11.61	**7.65**		
1.13	**6.14**	6.17	6.81	**6.11**	7.57	**6.03**	10.10	**7.28**		
1.155	5.62	**5.61**	6.29	**5.64**	8.39	**6.22**	11.82	**6.86**		
1.18	**5.77**	5.84	7.15	**6.19**	7.84	**6.18**	10.76	**7.65**		
1.205	**5.84**	5.91	6.73	**5.65**	7.75	**6.89**	10.22	**7.19**		
1.23	5.90	**5.89**	7.29	**5.77**	7.61	**6.17**	9.97	**7.38**		
1.255	5.97	**5.95**	6.63	**5.72**	7.65	**6.68**	9.87	**7.89**		
1.28	5.71	**5.66**	6.71	**5.57**	7.86	**5.98**	10.58	**7.30**		
1.305	5.79	5.79	6.08	6.08	7.94	**6.39**	9.44	**7.57**		
1.33	5.96	5.96	7.15	**5.86**	7.32	**6.38**	10.08	**7.10**		
1.355	5.57	5.57	**5.96**	6.09	7.61	**6.43**	9.05	**8.13**		
1.38	**5.84**	5.85	7.30	**5.86**	7.13	**5.98**	9.58	**8.16**		
1.405	**5.85**	5.87	6.63	**5.80**	7.01	**6.11**	9.62	**7.72**		
1.43	5.97	**5.95**	6.44	**5.98**	7.19	**6.46**	9.23	**7.30**		
1.455	5.97	5.97	6.71	**6.21**	7.24	**6.29**	9.06	**7.92**		
1.48	5.90	**5.87**	6.34	**5.89**	6.91	**6.36**	10.02	**7.55**		
1.505	**5.79**	5.80	6.83	**5.92**	7.38	**6.51**	9.69	**7.90**		
1.53	**6.06**	6.07	**5.77**	5.86	7.53	**6.35**	9.20	**8.08**		
1.555	**5.66**	5.69	6.02	**5.85**	7.59	**6.21**	8.97	**8.27**		
1.58	**5.82**	5.89	**5.80**	6.07	6.62	**5.84**	10.01	**8.33**		
1.605	5.95	**5.93**	**5.94**	6.09	6.84	**6.59**	8.60	**7.88**		
1.63	**5.91**	5.92	**5.41**	6.29	6.57	**6.35**	8.94	**7.47**		
1.655	5.89	5.89	**5.09**	5.73	7.26	**6.18**	9.60	**7.91**		
1.68	5.65	**5.61**	5.83	**5.67**	6.86	**6.47**	9.19	**8.01**		
1.705	5.81	**5.78**	6.27	**5.60**	6.99	**6.40**	9.03	**7.76**		
1.73	**5.91**	5.92	5.53	**5.53**	7.36	**6.08**	8.81	**8.04**		
1.755	5.82	5.82	6.03	**5.72**	7.31	**6.56**	9.19	**8.35**		
1.78	5.86	5.86	**5.42**	5.87	7.20	**6.21**	8.98	**8.32**		
1.805	**5.85**	5.89	6.76	**5.43**	7.18	**6.16**	8.98	**8.28**		

Table 5
The maximum and minimum training and testing time of WSVM for different abnormal patterns.

Abnormal pattern	Training time		Testing time	
	Min	Max	Min	Max
Uptrend	0.0007	0.0748	0.0001	0.0174
Downtrend	0.0006	0.0747	0.0001	0.0172
Upshift	0.0017	0.1594	0.0002	0.0160
Downshift	0.0016	0.1325	0.0002	0.0209
Systematic	0.0014	0.1293	0.0001	0.0234
Cyclic	0.0028	0.1323	0.0003	0.0248
Stratification	0.0059	0.0218	0.0006	0.0026

WSVM demonstrates a more robust behavior for various values of r. On the other side SVM performs well for low r (more balanced problems and rapidly decreases as r decreases (with only exception the (Sep) problems)). When r is low then the two formulations become equivalent and thus the performance is very similar (see Tables 7–8).

We extend the control chart pattern recognition from binary classification to multi-class classification and compare the results of WSVM with regular SVM for highly imbalanced datasets. In this classification test, there are seven classes: Normal, Uptrend, Downtrend, Up shift, Down shift, Cyclic, Systematic and Stratification patterns. Although the generic version of SVM can accommodate only two classes it is possible to generalize the classifier to multi class by constructing all the classifier pairs (one-against-one policy) and then use a *majority voting* scheme for assigning a new point to its class. This method has been found to perform well however the computational time of the model is expected to increase as the number of the classes increase. To study highly imbalanced classification problem, we generate 1000 data which consist of 951 normal data (approximately 95% of total data), and 49 abnormal data (approximately 5% of the total data). For each abnormal control chart pattern, 7 examples were generated. Similar to binary classification, we consider the weight of each class as inverse of the class size (Veropoulos et al., 1999)

$$C_i = \frac{C}{n_i} \quad i = 1, 2, \ldots, m. \tag{11}$$

where n_i and C_i are the class size and weight related to the class i respectively, $i = 1, 2, \ldots, m$ and m is the number of classes. The parameters C and γ were tuned during the training process through the same parameter grid search as in the binary case. The parameters for abnormal patterns are selected from Table 2.

We evaluate the performance of WSVM versus SVM for a partially separable problem with the following abnormal pattern characteristics: λ=0.58 (Downtrend), λ=0.93 (Uptrend), k = 1.53 (Systematic), ω=0.63 (Downshift), ω=0.38 (Upshift), α=0.405 (Cyclic), and $\acute{\varepsilon}$=0.805 (Stratification). We show the performance for three representative window (w) parameter values (10, 50, 100). The parameters of each problem were chosen randomly for

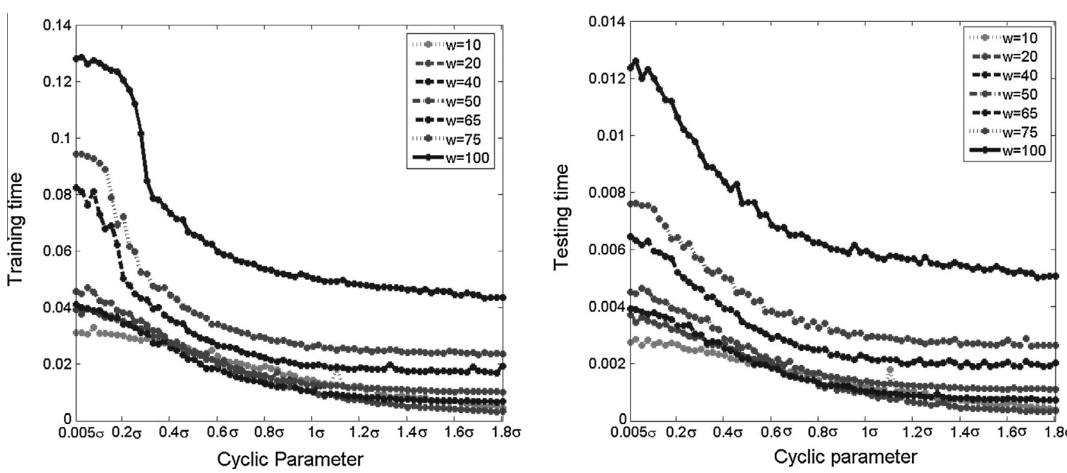


Fig. 6. WSVM training and testing time vs. abnormal parameter for cyclic patterns. The computation time decreases as the value of the parameter increases. This is expected since higher parameter values make the problem less challenging (more separable).

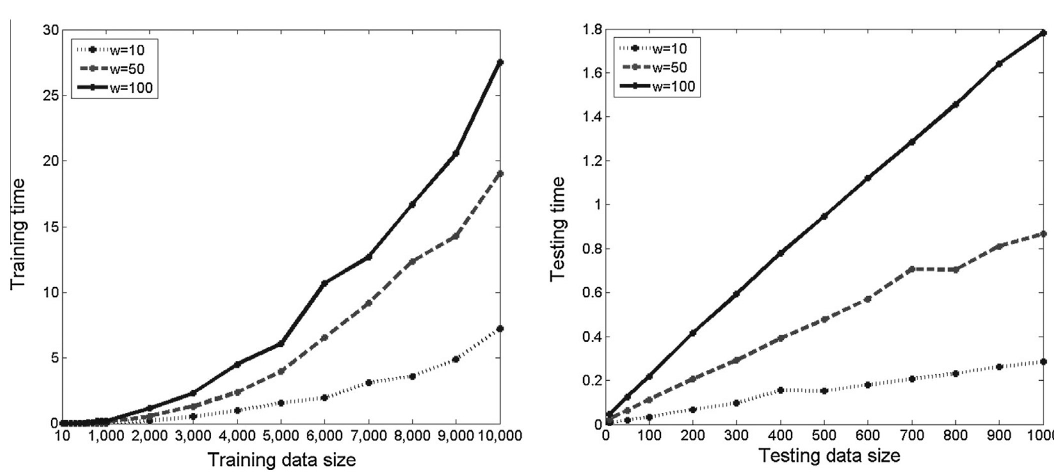


Fig. 7. WSVM training and testing time vs. training size for cyclic pattern.

Table 6
Sensitivity (Sen), Specificity (Spe), Accuracy (Acc), and G-means (G) of SVM and WSVM over all six abnormal patterns for different problems with three types including separable (Se), partially separable (Ps), and inseparable (Is). We define these three types based on SVM classification performance. With bold we denote the highest sensitivity, specificity and G-mean among the two algorithms.

Pattern	SVM				WSVM					Parameters	Type
	Sen	Spe	Acc	G	Sen	Spe	Acc	G	P-value		
	1.00	0.00	**0.93**	0.00	0.64	**0.54**	0.63	**0.57**	$< 10^{-4}$	(w = 10, λ=0.005)	Is
Ut	**0.99**	0.44	**0.95**	0.66	0.88	**0.83**	0.88	**0.86**	$< 10^{-4}$	(w = 15, λ=0.06)	Ps
	1.00	0.99	**1.00**	0.99	1.00	**0.99**	1.00	**1.00**	0.085	(w = 15, λ=0.16)	Se
	1.00	0.00	**0.92**	0.00	0.60	**0.45**	0.59	**0.52**	$< 10^{-4}$	(w = 10, λ=0.005)	Is
Dt	**0.99**	0.38	**0.94**	0.62	0.88	**0.81**	0.87	**0.84**	$< 10^{-4}$	(w = 15, λ=0.06)	Ps
	1.00	0.96	**1.00**	0.98	1.00	**0.98**	1.00	**0.99**	0.534	(w = 15, λ=0.16)	Se
	1.00	0.00	**0.93**	0.00	0.61	**0.51**	0.60	**0.54**	$< 10^{-4}$	(w = 10, ω=0.105)	Is
Us	**0.99**	0.40	**0.94**	0.63	0.90	**0.77**	0.89	**0.83**	$< 10^{-4}$	(w = 20, ω=0.43)	Ps
	1.00	0.98	**1.00**	0.99	1.00	**0.99**	1.00	**1.00**	0.085	(w = 20, ω=1.205)	Se
	1.00	0.00	**0.90**	0.00	0.64	**0.57**	0.63	**0.59**	$< 10^{-4}$	(w = 10, ω=0.105)	Is
Ds.	**0.97**	0.46	**0.93**	0.65	0.91	**0.72**	0.89	**0.81**	$< 10^{-4}$	(w = 20, ω=0.43)	Ps
	1.00	0.98	**1.00**	0.99	1.00	**1.00**	1.00	**1.00**	0.163	(w = 20, ω=1.205)	Se
	1.00	0.00	**0.93**	0.00	0.76	**0.74**	0.76	**0.74**	$< 10^{-4}$	(w = 10, k = 0.405)	Is
S	**1.00**	0.21	**0.94**	0.46	0.85	**0.70**	0.84	**0.77**	$< 10^{-4}$	(w = 15, k = 0.455)	Ps
	1.00	0.98	**1.00**	0.99	1.00	**0.98**	1.00	**0.99**	0.055	(w = 15, k = 1.405)	Se
	1.00	0.00	**0.94**	0.00	0.73	**0.50**	0.71	**0.59**	$< 10^{-4}$	(w = 15, α=0.23)	Is
C	**0.99**	0.19	**0.93**	0.43	0.87	**0.72**	0.86	**0.79**	$< 10^{-4}$	(w = 20, α=0.48)	Ps
	1.00	0.94	**0.99**	0.97	1.00	**0.98**	1.00	**0.99**	0.596	(w = 20, α=1.605)	Se
	1.00	0.00	**0.92**	0.00	0.63	**0.77**	0.64	**0.69**	$< 10^{-4}$	(w = 10, ε_t=0.78)	Is
Str	**1.00**	0.11	**0.93**	0.33	0.89	**0.96**	0.89	**0.92**	$< 10^{-4}$	(w = 15, ε_t=0.58)	Ps
	0.99	0.92	**0.98**	0.95	0.96	**1.00**	0.96	**0.98**	0.021	(w = 15, ε_t=0.43)	Se

Table 7
G-mean of SVM and WSVM of all patterns in Inseparable (Is) problems for different imbalanced ratio. The majority class contains $(50 + r)\%$ of the data and the minority $(50 - r)\%$.

r	Ut		Dt		Us		Ds		S		C		Str	
	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM
5	0.42	0.54	0.43	0.52	0.48	0.54	0.49	0.59	0.72	0.74	0.58	0.62	0.70	0.67
10	0.25	0.54	0.24	0.51	0.38	0.56	0.34	0.54	0.71	0.74	0.52	0.60	0.69	0.68
15	0.07	0.53	0.05	0.52	0.12	0.55	0.07	0.55	0.68	0.74	0.47	0.61	0.66	0.67
20	0.00	0.52	0.00	0.53	0.12	0.55	0.04	0.56	0.65	0.73	0.35	0.62	0.46	0.68
25	0.00	0.53	0.00	0.53	0.00	0.56	0.00	0.55	0.59	0.73	0.17	0.62	0.12	0.67
30	0.00	0.54	0.00	0.51	0.00	0.55	0.00	0.55	0.55	0.75	0.09	0.59	0.00	0.70
35	0.00	0.51	0.00	0.53	0.00	0.57	0.00	0.56	0.37	0.72	0.03	0.62	0.00	0.69
40	0.00	0.56	0.00	0.52	0.00	0.55	0.00	0.56	0.20	0.72	0.00	0.62	0.00	0.69
45	0.00	0.57	0.00	0.52	0.00	0.57	0.00	0.59	0.00	0.74	0.00	0.69	0.00	0.69

Table 8
G-mean of SVM and WSVM of all patterns in Partially separable (Ps) problems for different imbalanced ratio. The majority class contains $(50 + r)\%$ of the data and the minority $(50 - r)\%$.

r	Ut		Dt		Us		Ds		S		C		Str	
	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM
5	0.84	0.84	0.85	0.84	0.82	0.82	0.81	0.83	0.80	0.81	0.76	0.76	0.92	0.92
10	0.84	0.85	0.84	0.85	0.81	0.82	0.80	0.82	0.78	0.80	0.74	0.76	0.92	0.92
15	0.83	0.85	0.83	0.84	0.80	0.83	0.79	0.83	0.78	0.80	0.72	0.75	0.92	0.92
20	0.82	0.84	0.82	0.86	0.79	0.83	0.78	0.82	0.77	0.80	0.72	0.77	0.92	0.93
25	0.81	0.85	0.82	0.85	0.76	0.84	0.74	0.83	0.73	0.79	0.68	0.77	0.90	0.92
30	0.77	0.84	0.79	0.85	0.73	0.82	0.75	0.81	0.66	0.80	0.62	0.77	0.90	0.92
35	0.76	0.84	0.77	0.85	0.70	0.81	0.72	0.82	0.62	0.78	0.61	0.77	0.86	0.92
40	0.72	0.85	0.66	0.85	0.68	0.81	0.67	0.82	0.60	0.79	0.49	0.75	0.80	0.92
45	0.66	0.86	0.62	0.84	0.63	0.83	0.65	0.81	0.46	0.77	0.43	0.79	0.33	0.92

the parameter range shown in Table 2. the results are the average over ten such randomly selected problems. More computational experiments were conducted and can be found in the Supplementary material section of the paper. Since the problem contains more than two classes sensitivity specificity and G-mean are not defined. For this we provide the full confusion matrix that demonstrates the exact classification accuracy for each class separately (see Tables 10–12). As expected WSVM performs better in identifying the examples from the minority classes although it sacrifices some of the normal class accuracy. For short windows, SVM fails to detect

Table 9
G-mean of SVM and WSVM of all patterns in Separable (Se) problems for different imbalanced ratio. The majority class contains $(50 + r)\%$ of the data and the minority $(50 - r)\%$.

r	Ut		Dt		Us		Ds		S		C		Str	
	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM
5	1.00	1.00	1.00	1.00	0.99	1.00	0.99	0.99	1.00	1.00	0.99	0.99	0.98	0.98
10	1.00	1.00	1.00	1.00	0.99	1.00	1.00	0.99	1.00	1.00	0.99	0.99	0.98	0.98
15	1.00	1.00	1.00	1.00	0.99	1.00	0.99	0.99	1.00	1.00	0.99	0.99	0.98	0.98
20	1.00	1.00	1.00	1.00	0.99	1.00	0.99	0.99	1.00	0.99	0.99	0.99	0.98	0.97
25	1.00	1.00	1.00	1.00	0.99	1.00	0.99	0.99	0.99	1.00	0.99	0.99	0.98	0.98
30	1.00	1.00	1.00	1.00	0.99	1.00	0.99	0.99	0.99	0.99	0.99	0.99	0.98	0.98
35	1.00	1.00	0.99	1.00	0.99	1.00	0.99	0.99	1.00	1.00	0.99	0.99	0.99	0.98
40	1.00	1.00	0.99	1.00	0.99	1.00	0.99	1.00	0.99	1.00	0.99	0.99	0.99	0.98
45	0.99	1.00	0.98	0.99	0.99	1.00	0.99	1.00	0.99	0.99	0.97	0.99	0.99	0.98

Table 10
Classification results for multi-class SVM and WSVM for CCPR with window length = 10 and highly imbalanced data. Rows are related to predicted class labels and the columns are related to real labels. With bold we denote the highest accuracy value among the two algorithms.

	N	Dt	Ut	S	Ds	Us	C	Str
SVM								
N	**1.00**	0.00	0.00	0.05	1.00	1.00	1.00	1.00
Dt	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
Ut	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00
S	0.00	0.00	0.00	0.95	0.00	0.00	0.00	0.00
Ds	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Us	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
C	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Str	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
WSVM								
N	0.65	0.00	0.00	0.00	0.15	0.19	0.13	0.31
Dt	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00
Ut	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00
S	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00
Ds	0.06	0.00	0.00	0.00	**0.75**	0.00	0.00	0.08
Us	0.10	0.00	0.00	0.00	0.00	**0.77**	0.00	0.00
C	0.07	0.00	0.00	0.00	0.00	0.04	**0.70**	0.00
Str	0.11	0.00	0.00	0.00	0.10	0.00	0.17	**0.61**

Table 11
Classification results for multi-class SVM and WSVM for CCPR with window length = 50 and highly imbalanced data. With bold we denote the highest accuracy value among the two algorithms.

	N	Dt	Ut	S	Ds	Us	C	Str
SVM								
N	**1.00**	0.00	0.00	0.00	0.40	1.00	0.47	1.00
Dt	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
Ut	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00
S	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
Ds	0.00	0.00	0.00	0.00	0.60	0.00	0.00	0.00
Us	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
C	0.00	0.00	0.00	0.00	0.00	0.00	0.53	0.00
Str	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
WSVM								
N	0.98	0.00	0.00	0.00	0.20	0.37	0.27	0.37
Dt	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00
Ut	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00
S	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00
Ds	0.00	0.00	0.00	0.00	**0.80**	0.00	0.00	0.00
Us	0.00	0.00	0.00	0.00	0.00	**0.63**	0.00	0.00
C	0.00	0.00	0.00	0.00	0.00	0.00	**0.73**	0.00
Str	0.01	0.00	0.00	0.00	0.00	0.00	0.00	**0.63**

examples from four classes and for medium window for two. In general, a problem becomes less challenging as w increases. This is consistent with the binary classification examples.

The rows and columns of the *confusion matrix* show the predicted class and the real class respectively. Moreover, the diagonal elements show the accurate classification percentage. However, the accuracy for normal class has decreased slightly for multi-class WSVM, the fact is that the classification accuracy has increased for other patterns. In other words, results show that multi-class WSVM has higher performance than multi-class SVM for highly imbalanced data. The multi-class SVM assigns most of data to the majority class. In this highly imbalanced classification problem, we can observe that the behavior of multi-class SVM is very close to naive classification rules.

Table 12
Classification results for multi-class SVM and WSVM for CCPR with window length = 100 and highly imbalanced data. With bold we denote the highest accuracy value among the two algorithms.

	N	Dt	Ut	S	Ds	Us	C	Str
SVM								
N	**1.00**	0.00	0.00	0.00	0.25	0.28	0.47	1.00
Dt	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
Ut	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00
S	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
Ds	0.00	0.00	0.00	0.00	0.75	0.00	0.00	0.00
Us	0.00	0.00	0.00	0.00	0.00	0.72	0.00	0.00
C	0.00	0.00	0.00	0.00	0.00	0.00	0.53	0.00
Str	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
WSVM								
N	1.00	0.00	0.00	0.00	0.23	0.22	0.15	0.48
Dt	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00
Ut	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00
S	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00
Ds	0.00	0.00	0.00	0.00	**0.77**	0.00	0.00	0.00
Us	0.00	0.00	0.00	0.00	0.00	**0.78**	0.00	0.00
C	0.00	0.00	0.00	0.00	0.00	0.00	**0.85**	0.00
Str	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.52**

Next we compared SVM and WSVM in terms of ARL based measures for problems with various abnormal parameter values. Results for one instance are given in Table 13. Additional results in various experimental setups are included in the Supplementary material section. W e observe that WSVM performs the same as good or better in the discovery of most of abnormal patterns compared to SVM. For some instance SVM is not able to discover any of the abnormal patterns therefore ATPRL, ROT and ATPRIDS cannot be defined.

In addition as previous we computed the computational time for training and testing for the multiclass problem (Fig. 8). This experiment was conducted for three window lengths 10, 50, 100.

The trends observed are similar to binary classification however the seven classes do not add a high additional computational "overhead" compared to binary classification.

Last we compare the proposed algorithm for a real life application from wafer manufacturing industry. In wafer manufacturing. Electronics manufacturing usually involve a large number of steps (> 250) which can induce defects to the final product. Quality control is performed by recording the different frequencies that are emitted by the plasma during the process. The dataset is composed

Table 13
Multi-class SVM and WSVM with abnormal parameters values with w = 10,50,100. NaN stands for "Not a Number" and it assigned to instances where none abnormal patterns were discovered.

Parameters	Values	w	Pattern	SVM			WSVM		
				ATPRL	ROT	ATPRLIDX	ATPRL	ROT	ATPRLIDX
λ	0.58	10	Dt	7.60	1.00	7.60	7.10	1.00	7.10
λ	0.93		Ut	6.70	1.00	6.70	6.90	1.00	6.90
k	1.53		S	9.20	1.00	9.20	7.60	1.00	7.60
ω	0.63		Ds	NaN	NaN	NaN	4.60	1.00	4.60
ω	0.38		Us	NaN	NaN	NaN	5.80	1.00	5.80
α	0.405		C	NaN	NaN	NaN	6.78	0.90	7.53
ε	0.805		Str	NaN	NaN	NaN	4.70	1.00	4.70
		50	Dt	42.20	1.00	42.20	41.60	1.00	41.60
			Ut	46.20	1.00	46.20	45.30	1.00	45.30
			S	33.60	1.00	33.60	35.60	1.00	35.60
			Ds	28.80	1.00	28.80	38.30	1.00	38.30
			Us	48.88	0.80	61.09	42.60	1.00	42.60
			C	50.00	0.30	166.67	50.00	0.70	71.43
			Str	NaN	NaN	NaN	21.57	0.70	30.82
		100	Dt	92.40	1.00	92.40	92.80	1.00	92.80
			Ut	95.80	1.00	95.80	96.10	1.00	96.10
			S	78.80	1.00	78.80	81.00	1.00	81.00
			Ds	79.80	1.00	79.80	85.80	1.00	85.80
			Us	100.00	0.70	142.86	74.67	0.90	82.96
			C	100.00	0.60	166.67	95.00	0.80	118.75
			Str	NaN	NaN	NaN	95.89	0.90	106.54

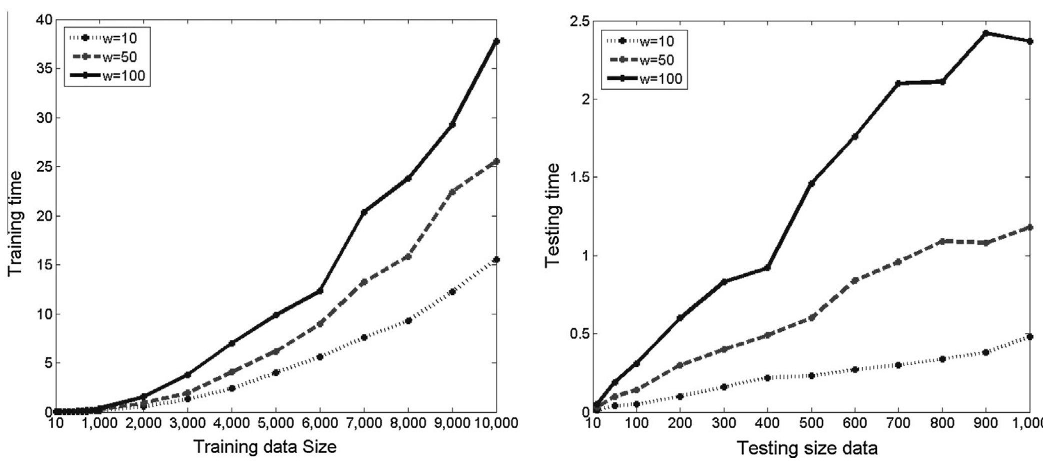


Fig. 8. WSVM training and testing time vs. training size for multi-class classification.

Table 14
Training and testing performance for the wafer dataset. With bold we denote the highest sensitivity, specificity, Gmean and accuracy value among the two algorithms.

	Sensitivity	Specificity	Gmean	Accuracy
Training				
SVM	**0.9996**	0.9160	0.9156	**0.9913**
SVM	0.9967	**0.9350**	**0.9319**	0.9905
Testing				
SVM	**0.9971**	0.9654	0.9811	**0.9937**
WSVM	0.9895	**0.9895**	**0.9895**	0.9895

Table 15
Testing performance for wafer dataset (with bold color is denoted the highest *G-mean* score).

C	SVM Gmean	WSVM Gmean
0.1	0.8756	**0.9684**
1	0.9790	**0.9895**
10	**0.9811**	**0.9811**
100	**0.9811**	**0.9811**
1000	**0.9811**	**0.9811**

out of 1000 training samples (of length152 each) and 6174 testing samples of the same length (Olszewski, 2001; Keogh et al., 2011). The training samples are imbalanced (903 are majority and 97 minority). We performed cross validation on training data and then we used the model developed from training for testing on the larger testing dataset (Table 14).

Last we perform sensitivity analysis by evaluating prediction performance for different values of penalty parameter C. Results are shown in Table 15.

4. Conclusions

Detecting abnormal patterns is an important task that has practical value related to diagnostic and maintenance operations. In this paper, we compared SVM against WSVM for the imbalanced CCPR problem. We tested the two algorithms for several normal and abnormal classification problems as well as multi-class classification in highly imbalanced environment. Comparison demonstrated that WSVM is better in terms of specificity and G-mean, two measures that are used for imbalanced classification problems. However, sensitivity and classification accuracy for WSVM drops, which is a compromise for correctly detecting the rare abnormal patterns. Therefore, the choice of the algorithm and the associated parameters is dominated by the proportion of available historical data, the cost of acquisition of new data and the minimal abnormal pattern alterations that ones wishes to detect. Note that, due to the uneven generation rates of normal and abnormal data, the imbalanced nature of the problem remains even in the case of off line construction of the training dataset. However even in the unlike case where the rate of abnormal patterns is very close to the normal ones WSVM just reduces to SVM since the penalty weights for the positive and negative class become equal.

Further work has to be done in order to determine a supervised or unsupervised discrimination scheme that would allow high specificity without compromising the sensitivity of the algorithm. Smart feature selection might need to be employed in order to improve classification accuracy whereas alternative imbalanced classification techniques are worthwhile to be explored. Re-sampling methods by themselves introduce biases and might not be optimal, however, their paired usage with cost sensitive methods has to be explored in future research. Some research works have started looking at this combined preprocessing strategy for other imbalanced problems (Anand et al., 2010; Akbani et al., 2004). Their potential usage has to be explored for CCPR as well. The potential application of clustering as preprocessing is another interesting research direction (Jo & Japkowicz, 2004). In this paper we focused our efforts in test involving fixed data time series of of various w length. In the future we will focus on stream data mining and attempt an on-line classifier with an incremental real time retraining. Current study results are encouraging enough in terms of accuracy, average run length and computational time. Another important aspect for the verification and validation of the proposed methods is the testing through real case studies and datasets. However, the lack of real data is common in the majority of CCPR literature. As it is pointed out in the review paper of Hachicha and Ghorbel (2012) approximately 95.59% of CCPR literature uses simulated time series data for CCPR algorithm validation.

Finally, we believe that future CCPR research should focus more on multi-class generalization. Since in reality one is interested to discriminate not only the normal versus abnormal problem but have as much information as possible about the abnormality, multi-class CCPR need to receive more attention in future works. The vast majority of previous studies focuses on the binary problem with only few exceptions (Ghanem, Venkatesh, & West, 2010; Shao, 2012). In this paper, we presented very promising multi-class results under a highly imbalanced environment, however, additional investigation and computational testing needs to be conducted in the future.

Appendix A. Mathematical models of control chart patterns

The western electric company (1958) first documented several typical control chart patterns. These patterns were subsequently used in a large number of CCPR publications (Hwarng & Hubele, 1992; Cheng, 1997; Al-Ghanim, 1997; Al-Assaf, 2004; Assaleh & Al-assaf, 2005; Gauri & Chakraborty, 2008; Cheng, Cheng, & Huang, 2009; Guh, 2010; Shao, 2012). These simulated control charts ($a(t)$) consist of three major components, namely a constant term μ, a random and normally distributed term ε_t, and a function $d(t)$ that models a particular abnormal pattern. This term is zero for in-control data The mathematical model for all components considered in this study can be written as:

$$a(t) = \mu + \varepsilon_t + d(t) \tag{12}$$

Without loss of generality, we use $\mu = 0$ and $\varepsilon_t \sim N(0,1)$ which is consistent with previous works (Cheng et al., 2009; Yang, 2010; Shao, 2012). In particular the form of $d(t)$ for each particular pattern is as follows:

(a) *Up/down trends*

$$d(t) = \lambda t \tag{13}$$

where λ is the trend slope in terms of σ_ε. The parameter $\lambda > 0$ is chosen for up trends and $\lambda < 0$ for downtrends.

(b) *Up/down shifts*

$$d(t) = \omega \tag{14}$$

where parameter ω denotes the shift magnitude. Similarly to the trend patterns $\omega > 0$ for up shift and $\omega < 0$ for down shift.

(c) *Cyclic trends*

$$d(t) = \alpha \sin\left(\frac{2\pi t}{\Omega}\right) \tag{15}$$

where α is the amplitude of the cyclic patterns, and Ω is the cyclic pattern period. For this paper, we fix $\Omega = 8$ and treat α as parameter similar to previous works (Cheng et al., 2009; Shao, 2012).

(d) *Systematic trends*

$$d(t) = k(-1)^t \tag{16}$$

where k is magnitude of the systematic pattern.

(d) *Stratification trend*

$$d(t) = \acute{\varepsilon}_t \tag{17}$$

is another abnormal pattern related to shift in the process standard deviation (ε_t). Parameter $\acute{\varepsilon}_t$ is a fraction of the natural process standard deviation.

Appendix B. Supplementary material

Supplementary data associated with this article can be found, in the online version, at http://dx.doi.org/10.1016/j.cie.2014.01.014.

References

Adam, A., Chew, L., Shapiai, M., Jau, L., Ibrahim, Z., & Khalid, M. (2011). A hybrid artificial neural network-naive bayes for solving imbalanced dataset problems in semiconductor manufacturing test process. In *2011 11th International conference on hybrid intelligent systems (HIS)* (pp. 133–138). IEEE.

Akbani, R., Kwek, S., & Japkowicz, N. (2004). Applying support vector machines to imbalanced datasets. *Machine Learning: ECML 2004*, 39–50.

Al-Assaf, Y. (2004). Multi-resolution wavelets analysis approach for the recognition of concurrent control chart patterns. *Quality Engineering, 17*(1), 11–21.

Al-Ghanim, A. (1997). An unsupervised learning neural algorithm for identifying process behavior on control charts and a comparison with supervised learning approaches. *Computers & Industrial Engineering, 32*(3), 627–639.

Al-Ghanim, A., & Kamat, S. (1995). Unnatural pattern recognition on control charts using correlation analysis techniques. *Computers & Industrial Engineering, 29*(1), 43–47.

Al-Shahib, A., Breitling, R., & Gilbert, D. (2005). Feature selection and the class imbalance problem in predicting protein function from sequence. *Applied Bioinformatics, 4*(3), 195–203.

Alexander, S. (1987). The application of expert systems to manufacturing process control. *Computers & Industrial Engineering, 12*(4), 307–314.

Alwan, L., & Roberts, H. (1988). Time-series modeling for statistical process control. *Journal of Business & Economic Statistics, 6*(1), 87–95.

Anand, A., Pugalenthi, G., Fogel, G., & Suganthan, P. (2010). An approach for classification of highly imbalanced data using weighting and undersampling. *Amino Acids, 39*(5), 1385–1391.

Aparisi, F. (1996). Hotelling's T^2 control chart with adaptive sample sizes. *International Journal of Production Research, 34*(10), 2853–2862.

Assaleh, K., & Al-assaf, Y. (2005). Features extraction and analysis for classifying causable patterns in control charts. *Computers & Industrial Engineering, 49*(1), 168–181.

Bahlmann, C., Haasdonk, B., & Burkhardt, H. (2002). Online handwriting recognition with support vector machines – A kernel approach. In *Proceedings. Eighth international workshop on frontiers in handwriting recognition* (pp. 49–54). IEEE.

Bao, Y., Liu, Z., Guo, L., & Wang, W. (2005). Forecasting stock composite index by fuzzy support vector machines regression. *Proceedings of 2005 international conference on machine learning and cybernetics* (vol. 6, pp. 3535–3540). IEEE.

Burges, C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery, 2*(2), 121–167.

Byvatov, E., Fechner, U., Sadowski, J., & Schneider, G. (2003). Comparison of support vector machine and artificial neural network systems for drug/nondrug classification. *Journal of Chemical Information and Computer Sciences, 43*(6), 1882–1889.

Camci, F., Chinnam, R., & Ellis, R. (2008). Robust kernel distance multivariate control chart using support vector principles. *International Journal of Production Research, 46*(18), 5075–5095.

Chang, C., & Lin, C. (2011). Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST), 2*(3), 1–27.

Chang, S., & Aw, C. (1996). A neural fuzzy control chart for detecting and classifying process mean shifts. *International Journal of Production Research, 34*(8), 2265–2278.

Chapelle, O., Haffner, P., & Vapnik, V. (1999). Support vector machines for histogram-based image classification. *IEEE Transactions on Neural Networks, 10*(5), 1055–1064.

Chawala, N. V., Bower, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research, 16*, 321–357.

Chawla, N., Hall, L., & Joshi, A. (2005). Wrapper-based computation and evaluation of sampling methods for imbalanced datasets. In *Proceedings of the 1st international workshop on utility-based data mining* (pp. 24–33). ACM.

Cheng, C. (1997). A neural network approach for the analysis of control chart patterns. *International Journal of Production Research, 35*(3), 667–697.

Cheng, C., Cheng, H., & Huang, K. (2009). A support vector machine-based pattern recognizer using selected features for control chart patterns analysis. In *IEEE international conference industrial engineering and engineering management, 2009. IEEM 2009* (pp. 419–423). IEEE.

Cheng, C.-S., & Cheng, H.-P. (2008). Identifying the source of variance shifts in the multivariate process using neural networks and support vector machines. *Expert Systems with Applications, 35*(1), 198–206.

Cheng, C.-S., & Hubele, N. F. (1992). Design of a knowledge-based expert system for statistical process control. *Computers & industrial engineering, 22*(4), 501–517.

Cheng, H., & Cheng, C. (2009). Control chart pattern recognition using wavelet analysis and neural networks. *Journal of Quality, 16*(5), 311–321.

Chinnam, R. B. (2002). Support vector machines for recognizing shifts in correlated and other manufacturing processes. *International Journal of Production Research, 40*(17), 4449–4466.

Cook, D., & Chiu, C. (1998). Using radial basis function neural networks to recognize shifts in correlated manufacturing process parameters. *IIE Transactions, 30*(3), 227–234.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning, 20*(3), 273–297.

Cristianini, N., & Shawe-Taylor, J. (2000). *An introduction to support vector machines: And other kernel-based learning methods*. Cambridge University Press.

Davy, M., Desobry, F., Gretton, A., & Doncarli, C. (2006). An online support vector machine for abnormal events detection. *Signal Processing, 86*(8), 2009–2025.

Du, S., & Chen, S. (2005). Weighted support vector machine for classification. *IEEE international conference systems, man and cybernetics* (Vol. 4, pp. 3866–3871). IEEE.

El-Midany, T., El-Baz, M., & Abd-Elwahed, M. (2010). A proposed framework for control chart pattern recognition in multivariate process using artificial neural networks. *Expert Systems with Applications, 37*(2), 1035–1042.

Estabrooks, A., Jo, T., & Japkowicz, N. (2004). A multiple resampling method for learning from imbalanced data sets. *Computational Intelligence, 20*(1), 18–36.

Ezawa, K., Singh, M., & Norton, S. (1996). Learning goal oriented bayesian networks for telecommunications risk management. In *Machine learning-international workshop conference* (pp. 139–147). Morgan Kaufmann Publishers, Inc..

Fan, H., & Ramamohanarao, K. (2005). A weighting scheme based on emerging patterns for weighted support vector machines. *2005 IEEE international conference on granular computing* (Vol. 2, pp. 435–440). IEEE.

Fawcett, T., & Provost, F. (1997). Adaptive fraud detection. *Data Mining and Knowledge Discovery, 1*(3), 291–316.

Foody, G., & Mathur, A. (2004). A relative evaluation of multiclass image classification by support vector machines. *IEEE Transactions on Geoscience and Remote Sensing, 42*(6), 1335–1343.

Gauri, S., & Chakraborty, S. (2008). Improved recognition of control chart patterns using artificial neural networks. *International Journal of Advanced Manufacturing Technology, 36*(11), 1191–1201.

Ghanem, A., Venkatesh, S., & West, G. (2010). Multi-class pattern classification in imbalanced data. In *Proceedings of the 2010 20th international conference on pattern recognition* (pp. 2881–2884). IEEE Computer Society.

Ghazanfari, M., Alaeddini, A., Niaki, S., & Aryanezhad, M. (2008). A clustering approach to identify the time of a step change in shewhart control charts. *Quality and Reliability Engineering International, 24*(7), 765–778.

Groth, S. S., & Muntermann, J. (2011). An intraday market risk management approach based on textual analysis. *Decision Support Systems, 50*(4), 680–691.

Guh, R. (2010). Simultaneous process mean and variance monitoring using artificial neural networks. *Computers & Industrial Engineering, 58*(4), 739–753.

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning, 46*(1), 389–422.

Hachicha, W., & Ghorbel, A. (2012). A survey of control-chart pattern-recognition literature (1991–2010) based on a new conceptual classification scheme. *Computers & Industrial Engineering, 63*(1), 204–222.

Hu, G., Zhu, F., & Ren, Z. (2008). Power quality disturbance identification using wavelet packet energy entropy and weighted support vector machines. *Expert Systems with Applications, 35*(1), 143–149.

Huang, Y., & Du, S. (2005). Weighted support vector machine for classification with uneven training class sizes. *Proceedings of 2005 international conference on machine learning and cybernetics, 2005* (Vol. 7, pp. 4365–4369). IEEE.

Hwang, J., Park, S., & Kim, E. (2011). A new weighted approach to imbalanced data classification problem via support vector machine with quadratic cost function. *Expert Systems with Applications, 38*(7), 8580–8585.

Hwarng, H. (1995). Multilayer perceptions for detecting cyclic data on control charts. *International Journal of Production Research, 33*(11), 3101–3117.

Hwarng, H. B., & Hubele, N. F. (1991). X-bar chart pattern recognition using neural nets. In: *ASQC Quality Congress Transactions* (Vol. 45, pp. 884–889). Milwaukee.

Hwarng, H. B., & Hubele, N. F. (1992). Boltzmann machines that learn to recognize patterns on control charts. *Statistics and Computing, 2*(4), 191–202.

Hwarng, H. B., & Hubele, N. F. (1993). Back-propagation pattern recognizers for x control charts: methodology and performance. *Computers & Industrial Engineering, 24*(2), 219–235.

Hwarng, H. B., & Hubele, N. F. (1993). X control chart pattern identification through efficient off-line neural network training. *IIE Transactions, 25*(3), 27–40.

Issam, B. K., & Mohamed, L. (2008). Support vector regression based residual mcusum control chart for autocorrelated process. *Applied Mathematics and Computation, 201*(1), 565–574.

Jang, K., Yang, K., & Kang, C. (2003). Application of artificial neural network to identify non-random variation patterns on the run chart in automotive assembly process. *International Journal of Production Research, 41*(6), 1239–1254.

Jin, J., & Shi, J. (2001). Automatic feature extraction of waveform signals for in-process diagnostic performance improvement. *Journal of Intelligent Manufacturing, 12*(3), 257–268.

Jo, T., & Japkowicz, N. (2004). Class imbalances versus small disjuncts. *ACM SIGKDD Explorations Newsletter, 6*(1), 40–49.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. *Machine Learning: ECML-98*, 137–142.

Kawamura, A., Chuarayapratip, R., & Haneyoshi, T. (1988). Deadbeat control of pwm inverter with modified pulse patterns for uninterruptible power supply. *IEEE Transactions Industrial Electronics, 35*(2), 295–300.

Keogh, E., Zhu, Q., Hu, B., Hao. Y., Xi, X., Wei, L., & Ratanamahatana, C. A. (2011). The UCR time series classification/clustering. <http://www.cs.ucr.edu/~eamonn/time_series_data/>.

Kubat, M., Holte, R., & Matwin, S. (1998). Machine learning for the detection of oil spills in satellite radar images. *Machine Learning, 30*(2), 195–215.

Kumar, S., Choudhary, A., Kumar, M., Shankar, R., & Tiwari, M. (2006). Kernel distance-based robust support vector methods and its application in developing a robust *k*-chart. *International Journal of Production Research, 44*(1), 77–96.

Lee, D. E., Song, J.-H., Song, S.-O., & Yoon, E. S. (2005). Weighted support vector machine for quality estimation in the polymerization process. *Industrial & Engineering Chemistry Research, 44*(7), 2101–2105.

Leslie, C., Eskin, E., & Noble, W. (2002). The spectrum kernel: A string kernel for svm protein classification. In: *Proceedings of the Pacific symposium on biocomputing, Hawaii, USA* (Vol. 7, pp. 566–575).

Lin, C., & Wang, S. (2002). Fuzzy support vector machines. *IEEE Transactions on Neural Networks, 13*(2), 464–471.

Liu, S., Jia, C., & Ma, H. (2005). A new weighted support vector machine with GA-based parameter selection. *Proceedings of 2005 international conference machine learning and cybernetics, 2005* (Vol. 7, pp. 4351–4355). IEEE.

Liu, X., Wu, J., & Zhou, Z. (2009). Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 39*(2), 539–550.

Lu, C.-J., Shao, Y. E., & Li, P.-H. (2011). Mixture control chart patterns recognition using independent component analysis and support vector machine. *Neurocomputing, 74*(11), 1908–1914.

Mandel, B. (1969). The regression control chart. *Journal of Quality Technology, 1*(1), 1–9.

Montgomery, D. (2007). *Introduction to statistical quality control*. John Wiley & Sons.

Olszewski, R. T. (2001). *Generalized feature extraction for structural pattern recognition in time-series data*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA.

Panagiotidou, S., & Tagaras, G. (2012). Optimal integrated process control and maintenance under general deterioration. *Reliability Engineering & System Safety*.

Paté-Cornell, M., Lee, H., & Tagaras, G. (1987). Warnings of malfunction: the decision to inspect and maintain production processes on schedule or on demand. *Management Science, 33*(10), 1277–1290.

Pilászy, I. (2005). Text categorization and support vector machines. In: *The proceedings of the 6th international symposium of Hungarian researchers on computational intelligence*.

Pugh, G. (1989). Synthetic neural networks for process control. *Computers & Industrial Engineering, 17*(1), 24–26.

Ranaee, V., Ebrahimzadeh, A., Ghaderi, R., et al. (2010). Application of the PSO-SVM model for recognition of control chart patterns. *ISA Transactions, 49*(4), 577.

Shao, X. (2012). Recognition of control chart patterns using decision tree of multi-class SVM. *Advances in Intelligent Systems*, 33–41.
Sukchotrat, T., Kim, S., & Tsung, F. (2009). One-class classification-based control charts for multivariate process monitoring. *IIE Transactions, 42*(2), 107–120.
Sun, R., & Tsung, F. (2003). A kernel-distance-based multivariate control chart using support vector methods. *International Journal of Production Research, 41*(13), 2975–2989.
Suresh, S., Venkatesh Babu, R., & Kim, H. (2009). No-reference image quality assessment using modified extreme learning machine classifier. *Applied Soft Computing, 9*(2), 541–552.
Suykens, J., De Brabanter, J., Lukas, L., & Vandewalle, J. (2002). Weighted least squares support vector machines: Robustness and sparse approximation. *Neurocomputing, 48*(1), 85–105.
Swift, J. A. (1987). *Development of a knowledge-based expert system for control-chart pattern recognition and analysis*. Tech. rep., Oklahoma State Univ., Stillwater, OK (USA).
Tang, Y., Krasser, S., Judge, P., & Zhang, Y. (2006). Fast and effective spam sender detection with granular SVM on highly imbalanced mail server behavior data. In *Collaborative Computing: Networking, Applications and Worksharing* (pp. 1–6). IEEE.
Tang, Y., Zhang, Y., Chawla, N., & Krasser, S. (2009). SVMs modeling for highly imbalanced classification. *IEEE Transactions Systems, Man, and Cybernetics, Part B: Cybernetics, 39*(1), 281–288.
Taylan, O., & Darrab, I. (2012). Fuzzy control charts for process quality improvement and product assessment in tip shear carpet industry. *Journal of Manufacturing Technology Management, 23*(3), 402–420.
Vapnik, V. (2000). *The nature of statistical learning theory*. Springer Verlag New York Inc..
Veropoulos, K., Campbell, C., & Cristianini, N. (1999). Controlling the sensitivity of support vector machines. In: *Proceedings of the international joint conference on artificial intelligence* (pp. 55–60).
Wang, C., Guo, R., Chiang, M., & Wong, J. (2008). Decision tree based control chart pattern recognition. *International Journal of Production Research, 46*(17), 4889–4901.
Wang, X. (2008). Hybrid abnormal patterns recognition of control chart using support vector machining. *International conference on computational intelligence and security, 2008. CIS'08* (Vol. 2, pp. 238–241). IEEE.
Weiss, G. (2004). Mining with rarity: A unifying framework. *SIGKDD Explorations, 6*(1), 7–19.
Wu, G., Wu, Y., Jiao, L., Wang, Y., & Chang, E. (2003). Multi-camera spatio-temporal fusion and biased sequence-data learning for security surveillance. In *Proceedings of the eleventh ACM international conference on multimedia* (pp. 528–538). ACM.
Yang, J., & Yang, M. (2005). A control chart pattern recognition system using a statistical correlation coefficient method. *Computers & Industrial Engineering, 48*(2), 205–221.
Yang, S. (2010). Process control using VSI cause selecting control charts. *Journal of Intelligent Manufacturing, 21*(6), 853–867.
Zarandi, M., & Alaeddini, A. (2010). A general fuzzy-statistical clustering approach for estimating the time of change in variable sampling control charts. *Information Sciences, 180*(16), 3033–3044.
Zavaljevski, N., Stevens, F., & Reifman, J. (2002). Support vector machines with selective kernel scaling for protein classification and identification of key amino acid positions. *Bioinformatics, 18*(5), 689–696.
Zhang, Q., Liu, D., Fan, Z., Lee, Y., & Li, Z. (2012). Feature and sample weighted support vector machine. In *Knowledge engineering and management*. *Advances in intelligent and soft computing* (Vol. 123, pp. 365–371). Berlin/Heidelberg: Springer.
Zorriassatine, F., Al-Habaibeh, A., Parkin, R., Jackson, M., & Coy, J. (2005). Novelty detection for practical pattern recognition in condition monitoring of multivariate processes: A case study. *The International Journal of Advanced Manufacturing Technology, 25*(9), 954–963.