OPERATIONS RESEARCH
Vol. 58, No. 5, September–October 2010, pp. 1469–1480
ISSN 0030-364X | EISSN 1526-5463 | 10 | 5805 | 1469

DOI 10.1287/opre.1100.0825
© 2010 INFORMS

A Computable Plug-In Estimator of Minimum Volume Sets for Novelty Detection

Chiwoo Park
Department of Industrial and Systems Engineering, Texas A&M University, College Station, Texas 77843,
chiwoo.park@tamu.edu

Jianhua Z. Huang
Department of Statistics, Texas A&M University, College Station, Texas 77843, jianhua@stat.tamu.edu

Yu Ding
Department of Industrial and Systems Engineering, Texas A&M University, College Station, Texas 77843,
yuding@iemail.tamu.edu

A minimum volume set of a probability density is a region of minimum size among the regions covering a given probability mass of the density. Effective methods for finding the minimum volume sets are very useful for detecting failures or anomalies in commercial and security applications—a problem known as *novelty detection*. One theoretical approach of estimating the minimum volume set is to use a density level set where a kernel density estimator is plugged into the optimization problem that yields the appropriate level. Such a plug-in estimator is not of practical use because solving the corresponding minimization problem is usually intractable. A modified plug-in estimator was proposed by Hyndman in 1996 to overcome the computation difficulty of the theoretical approach but is not well studied in the literature. In this paper, we provide theoretical support to this estimator by showing its asymptotic consistency. We also show that this estimator is very competitive to other existing novelty detection methods through an extensive empirical study.

Subject classifications: density level sets; minimum volume sets; novelty detection; generalized statistical control chart; plug-in estimator; asymptotic consistency.
Area of review: Stochastic Models.
History: Received March 2009; revisions received September 2009, November 2009; accepted December 2009. Published online in *Articles in Advance* September 13, 2010.

1. Introduction

Novelty detection is to identify unknown, anomalous events and separate them from normal events. The capacity of novelty detection is critical in many commercial or security applications because detection of potential failures or abnormal activities provides the opportunity to prevent a catastrophic outcome from happening. Novelty detection methodologies find numerous applications, such as in reliability improvement via fault detection of mission-critical systems (Hayton et al. 2000, Sanseverino and Zio 2007), quality control in manufacturing and production systems (Guh et al. 1999, Jin and Shi 2001), medical diagnosis (Tarassenko et al. 1995), and structural health monitoring (Worden et al. 2000).

Novelty detection methods are based on a single class of data, which is the data set of the normal events. The reasons for novelty detection to be based on the normal events data alone are twofold: (a) there are usually plenty of normal events data but not sufficient abnormal events, especially in mission-critical systems where people would expect failures to occur rarely; (b) even if there are a decent amount of abnormal events, they might represent only one type of fault or failure. The faults or failures occurring in the future could be distinctively different not only from the normal events but also from the past failures. Thus, novelty detection is often achieved by profiling the features that can well describe past normal events. Markou and Singh (2003a, b) provide a comprehensive, two-part review of novelty detection methods and their applications.

A classic method for novelty detection is the Hotelling's T^2 chart (Montgomery 1997), widely used in statistical quality control. In recent years, there have emerged more powerful novelty detection methods based on minimum volume (MV) set estimation. These methods find the minimal closed set covering a certain probability mass α with respect to the unknown density of the normal events. If a new event belongs to the minimal set, one regards the event as normalcy; otherwise, anomaly. For the resulting novelty detection rule, the probability that normal events lie outside the minimal set, i.e., the type-I error, is controlled at $1-\alpha$. Because the volume of the set is minimized, the probability that potential abnormal events fall inside the set, i.e., the type-II error, is also minimized.

There are primarily two schools of thought about estimating an MV-set. The first school attempts to directly estimate the MV-set by choosing the minimal set containing

α portion of the sample points among a special class of measurable sets such as a Glivenko-Cantelli (GC), Vapnik-Chervonenkis (VC), or Donsker class. Many (reproducing) kernel-based methods find the MV-set from a simple GC class in reproducing kernel Hilbert space, or RKHS: The one-class support vector machine method considered half spaces (Schölkopf et al. 2001); the support vector domain description method, the closed balls (Tax and Duin 1999); and the kernel minimum volume covering ellipsoid method, the ellipsoids (Dolia et al. 2004, 2007). Because of the simplicity of the GC class considered, computationally efficient algorithms based on quadratic programming are developed for these methods. However, because the MV-set in RKHS is not directly associated with the MV-set in the original space, these methods tend to generate loose set coverings and thus have high type-II error rate. For example, see Figure 2 in §3 of this paper or Figures 5 and 6 of Hoffmann (2007). More complicated instances of GC-classes could be formed as a composition of simple GC-classes, for example, the k-constructible for a finite union of the sets in GC-classes (Polonik 1995) or the dyadic decision tree for a composition of boxes (Scott and Nowak 2006). Through composition of simple sets, the MV-set estimator can be more flexible and yields a smaller type-II error rate. Nonetheless, because the number of the possible combinations of composing the simple GC-classes increases exponentially as the dimension of data increases, implementation of the composition-based methods is a challenge. In fact, the k-constructible remains a theoretical development without a computational implementation, and the application of the dyadic decision tree has been restricted to low-dimensional settings.

The second school of thought for MV-set estimation utilizes the relationship between the MV-set and the level set of a probability density and reduces the MV-set estimation problem to a density level set estimation problem (Garcia et al. 2003). The appropriate level of the density level set is a solution to an optimization problem. Because a kernel density estimator needs to be plugged into the solution of the optimization problem to obtain an estimated density level set, the resulting estimator is called a *plug-in estimator*. Although such a plug-in estimator has some nice theoretical properties (Garcia et al. 2003, Baillo 2003, Cadre 2006), it is not computable and thus is mainly of theoretical value rather than being practically useful. On the other hand, Hyndman (1996) discussed an appealing idea to obtain a computable plug-in estimator. However, this computable estimator is neither analyzed theoretically nor evaluated by empirical studies. The primary objective of this paper is to provide a theoretical support and extensive empirical studies on the plug-in estimator proposed by Hyndman (1996).

The rest of this paper is organized as follows. In §2, we formulate the novelty detection problem as a level set estimation problem and present a computable plug-in estimator of a density level set. We show that this plug-in estimator is asymptotically consistent. In §3, we test the computable plug-in estimator in a number of examples, including two artificial data sets and four real data sets, and compare its performances (in terms of type-I and type-II error rates) with the T^2 control regions and four existing MV-set estimators, which are the one-class SVM, the support vector domain description, the minimum volume covering ellipsoid, and the dyadic decision tree. Some concluding remarks are given in §4.

2. Novelty Detection by Set Estimation

We want to define a novelty detection rule D so that if a new data point $x \in \mathscr{X} \subset \mathbb{R}^d$ meets $D(x) > t$, we infer that x is a normal event, otherwise x is an abnormal event. Borrowing terminology from statistical hypothesis testing, we call $A = \{x\colon D(x) > t\}$ the acceptance region. We assume that a normal event is a random draw from a probability distribution on $\mathscr{X}$ with density function $f(x)$. For a given $\alpha \in (0,1)$, it is required that $P(x \in A \mid x$ is a normal event$) \geqslant \alpha$. This requirement controls the probability that a normal event being incorrectly classified, i.e., the type-I error, to be no greater than $1-\alpha$. The requirement can also be written in terms of the probability density function of the normal event as $\int_A f(x)\,dx \geqslant \alpha$.

Let $\mathscr{A} = \{A\colon \int_A f(x)\,dx \geqslant \alpha\}$ denote a collection of the acceptance regions meeting the requirement that controls the type-I error. To select the most suitable acceptance region from the collection $\mathscr{A}$, a sensible criterion is to minimize the probability that an abnormal event belongs to $A \in \mathscr{A}$, i.e., the type-II error of the novelty detection. If the abnormal event is from a probability distribution on $\mathscr{X}$ with a density bounded above by a constant C on $A \in \mathscr{A}$, then the type-II error is bounded by $C\lambda(A)$, where $\lambda(A)$ denote the Lebesgue measure of A. Because the density function of the abnormal event is unknown, we reduce the target of minimizing the type-II error to minimizing the upper bound $C\lambda(A)$, or equivalently, the volume $\lambda(A)$ of the set A. Thus, the acceptance region $A^* \in \mathscr{A}$ for novelty detection is defined as a solution of the following minimization problem:

$$\min\left\{\lambda(A)\colon \int_A f(x)\,dx \geqslant \alpha\right\}. \tag{1}$$

The detection rule is obtained by setting $D(x) = 1_{A^*}(x)$ and $t = 0$. A solution A^* of (1) is called a minimum volume (MV) set (Polonik 1995, Scott and Nowak 2006). Therefore, the problem of deriving a novelty detection rule is reduced to the one of finding an MV-set. In practice, the density function f that appeared in (1) is unknown and needs to be estimated using observations from normal events.

2.1. Plug-In Estimation

In this section, we briefly review the development of a plug-in estimator of a density level set for MV-set estimation.

We also show that the plug-in estimator is not computable, and thus motivate our study of a computable estimation in this paper.

Define the level set of density f at level y by $A_y = \{x\colon f(x) \geqslant y\}$. Garcia et al. (2003) showed that, under some regularity conditions, the density level set at a suitable level y is a minimum volume set. In fact, Garcia et al. (2003) proved the result for univariate densities but the argument goes through for multivariate densities. According to this result, we can restrict our attention to density level sets when finding an MV-set. Therefore, the optimization problem (1) is reduced to

$$\min\left\{\lambda(A_y)\colon \int_{A_y} f(x)\,dx \geqslant \alpha\right\}. \tag{2}$$

By monotonicity of the Lebesgue measure, $\lambda(A_y)$ is a decreasing function of y and minimizing $\lambda(A_y)$ is equivalent to maximizing y. Thus, optimization problem (2) is equivalent to

$$\max\left\{y \in \mathbb{R}^+\colon \int_{A_y} f(x)\,dx \geqslant \alpha\right\}. \tag{3}$$

If f is continuous and $\lambda(\{x\colon f(x) = y\}) = 0$ for all $y \in (0, \sup_x f(x))$, problem (3) has a unique solution (Garcia et al. 2003). Denote the solution as y^*. We call the corresponding level set A_{y^*} a minimum volume cut and denote it as $MVC(\alpha; f)$.

Because f is unknown, one can replace it by the following kernel density estimator:

$$\hat{f}_n(x) = \frac{1}{nh_n^d}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h_n}\right), \tag{4}$$

where K is a kernel function, $h_n > 0$ is the bandwidth, and $x_1, \ldots, x_n$ are observations from the distribution with density f, i.e., the normal events. Optimization problem (1) then becomes

$$\max\left\{y \in \mathbb{R}^+\colon \int_{\hat{A}_{n,y}} \hat{f}_n(x)\,dx \geqslant \alpha\right\},\quad \text{where } \hat{A}_{n,y} = \{x\colon \hat{f}_n(x) \geqslant y\}. \tag{5}$$

The estimated level set $\hat{A}_{n,y}$ corresponding to the solution of this problem is called the plug-in estimator of $MVC(\alpha; f)$ and is denoted as $MVC(\alpha; \hat{f}_n)$. Under some regularity conditions, Cadre (2006) proved that $MVC(\alpha; \hat{f}_n)$ is a consistent estimator of $MVC(\alpha; f)$.

Computing $MVC(\alpha; \hat{f}_n)$ is difficult. To solve optimization problem (5), one needs to know which y satisfies the inequality constraint, $\int_{\hat{A}_{n,y}} \hat{f}_n(x)\,dx \geqslant \alpha$. Unfortunately, the integral of $\hat{f}_n(x)$ over the complicated set $\{x\colon \hat{f}_n(x) \geqslant y\}$ is usually intractable. As a consequence, the plug-in estimation is not applicable in practice.

2.2. A Computable Plug-In Estimator

The evaluation of the integral in optimization problem (5) can be avoided, using an idea of Hyndman (1996). The idea is to replace the integral with respect to the kernel density estimation by the integral with respect to the empirical distribution. We show in this section that the plug-in estimator induced by this idea is consistent.

Denote the empirical distribution by $P_n(A) = (1/n)\sum_{i=1}^{n} 1_A(x_i)$ for given data points $x_1, .., x_n$. We solve

$$\max\{y \in \mathbb{R}^+\colon P_n(\hat{A}_{n,y}) \geqslant \alpha\},\quad \text{where } \hat{A}_{n,y} = \{x\colon \hat{f}_n(x) \geqslant y\}. \tag{6}$$

This problem has a closed-form solution that is easily computable. Note that

$$P_n(\hat{A}_{n,y}) = \frac{1}{n}\sum_{i=1}^{n} 1_{\{\hat{f}_n(x_i) \geqslant y\}} = \frac{N_{n,y}}{n},$$

where $N_{n,y}$ is the number of observations that satisfy $\hat{f}_n(x_i) \geqslant y$. Using this equality, it is easy to see that $P_n(\hat{A}_{n,y})$ is a left-continuous, nonincreasing step function with steps $1/n$ and $\hat{f}_n(x_i)$ as jump points. Therefore, the solution z_n of optimization problem (6) is the $(\lfloor n(1-\alpha)\rfloor)$th-order statistics of $\{\hat{f}_n(x_1), \hat{f}_n(x_2), \ldots, \hat{f}_n(x_n)\}$, where $\lfloor a \rfloor$ denote the largest integer that is smaller than or equal to a. For future use, denote the level set of $\hat{f}_n$ at level z_n, $\{x\colon \hat{f}_n(x) \geqslant z_n\}$, by $MVC(\alpha; \hat{f}_n, P_n)$. The novelty detection rule is given as follows: if a new data point x belongs to $MVC(\alpha; \hat{f}_n, P_n)$, x is regarded as a normal event, otherwise as an abnormal one.

To prove the consistency of the computable plug-in estimator given in the previous paragraph, we introduce three regularity conditions. These conditions are all used in Cadre (2006) to obtain the consistency of the original plug-in estimator. In the following, let $\Theta \subset (0, \sup f)$ be an open interval that contains the level y^* corresponding to $MVC(\alpha; f)$, and let $\|\cdot\|$ stand for the Euclidean norm over any finite-dimensional space. Let $A\Delta B = (A \cap B^c) \cup (A^c \cap B)$ denote the symmetric difference of sets A and B.

Assumption 1. *The kernel function K is continuously differentiable and has compact support. Moreover, there exists a monotone nondecreasing function μ: $\mathbb{R}_+ \to \mathbb{R}$ such that $K(x) = \mu(\|x\|)$ for all $x \in \mathbb{R}^d$.*

Assumption 2. *The density function f is twice continuously differentiable and $f(x) \to 0$ as $\|x\| \to \infty$.*

Assumption 3. *For any $t \in \Theta$, $\inf_{f^{-1}(\{t\})} \|\nabla f\| > 0$, where $\nabla f(x)$ is the gradient of f at x.*

Assumptions 2 and 3 imply that for any $t \in \Theta$, $\lambda(f^{-1}[t-\epsilon, t+\epsilon]) \to 0$ as $\epsilon \to 0$ (Cadre 2006).

Theorem 1. *Suppose that Assumptions 1, 2, and 3 hold. If the bandwidth h_n used in the kernel density estimation* *satisfies that* $nh_n^{d+4}(\log n)^2 \to 0$ *and* $nh_n^{d+2}/(\log n) \to \infty$, *then*

$$\int_{MVC(\alpha;\,\hat{f}_n,\,P_n)} f(x)\,dx \to \alpha \quad \textit{in probability,}$$

$$\lambda\{MVC(\alpha;\hat{f}_n,P_n)\Delta MVC(\alpha;f)\} \to 0 \quad \textit{in probability.}$$

The first part of the theorem says that the coverage probability of $MVC(\alpha;\hat{f}_n,P_n)$ for normal events converges to the target value α. The second part says that $MVC(\alpha;\hat{f}_n,P_n)$ is a consistent estimate of the theoretical MV-set $MVC(\alpha;f)$. The proof of the theorem is given in the online appendix.

An electronic companion to this paper is available as part of the online version that can be found at http://or.journal.informs.org/.

Remark 1. To get some guidance on what bandwidth to use in practice, we write $h_n = n^{-s}$ with $s > 0$. To make such h_n satisfy the conditions in Theorem 1, s should be in the range that $(d+3)/((d+2)(d+4)) < s < (2d+3)/(2(d+2)^2)$. Because we do not have any result on convergence rate, we cannot say which value within the range gives the best rate of convergence. However, for $d \geqslant 2$, the gap between $(d+3)/((d+2)(d+4))$ and $(2d+3)/(2(d+2)^2)$ is as small as 0.01, and the gap decreases as d increases. In our implementation of the method, we use the average value of the two bound values:

$$s = 0.5\frac{(d+3)}{(d+2)(d+4)} + 0.5\frac{(2d+3)}{2(d+2)^2}. \tag{7}$$

Remark 2 (MV-Set Estimation for Multivariate Normal Distribution). This remark connects the classical statistical theory of quality control to the theory of minimum volume set. We show that Hotelling's T^2 control region is a special case of the MV-set estimation. Suppose that $f(x)$ is the density function of the multivariate normal distribution $N(\mu_0,\Sigma_0)$. Its level set with level y is given by $A_y = \{x\colon (x-\mu_0)^T\Sigma_0^{-1}(x-\mu_0) \leqslant \delta(y)\}$, where $\delta(y)$ is a function of y. For a random vector X from $N(\mu_0,\Sigma_0)$, $(X-\mu_0)^T\Sigma_0^{-1}(X-\mu_0)$ follows a χ^2 distribution with d degrees of freedom (see Mardia et al. 1980, pp. 66–76). Using this fact, we obtain that $\delta(y) = \chi^2_{\alpha,d}$ ensures the $P(A_y) = \alpha$, where $\chi^2_{\alpha,d}$ is the αth-quantile of the χ^2_d distribution. Therefore, $MVC(\alpha;f) = \{x\colon (x-\mu_0)^T\Sigma_0^{-1}(x-\mu_0) \leqslant \chi^2_{\alpha,d}\}$. When (μ_0,Σ_0) are unknown, one can replace them by the MLE, $(\hat{\mu}_n,\hat{\Sigma}_n)$, in $MVC(\alpha;f)$ and obtain a plug-in estimator:

$$MVC(\alpha;\hat{f}_n) = \{x\colon (x-\hat{\mu}_n)^T\hat{\Sigma}_n^{-1}(x-\hat{\mu}_n) \leqslant \chi^2_{\alpha,d}\}. \tag{8}$$

The consistency of this plug-in estimator follows directly from the standard asymptotic theory of the MLE. The novelty detection rule associated with (8) is the same as Hotelling's T^2 control region, where $(X-\hat{\mu}_n)^T\hat{\Sigma}_n^{-1}(X-\hat{\mu}_n)$ is the test statistic and $\chi^2_{\alpha,d}$ is equivalent to the upper control limit (UCL) for sufficiently large n (see Montgomery 1997, pp. 369–371).

3. Experiments

We applied the computable plug-in estimator to a number of examples of artificial or real data, and we compared it with four existing methods: the one-class SVM (OC-SVM), the support vector domain description (SVDD), the kernel minimum volume covering ellipsoid (KMVCE), and the dyadic decision tree (DDT). Although the plug-in estimator works for high-dimensional data, we chose to perform data reduction first by using principal component analysis (PCA) to speed up the learning process. The reduced dimension is chosen so that the retained principal components explain 90% of the original variability. Because determination of reduced dimension is important to the subsequent novelty detection, further study of this issue is of interest but beyond the scope of this paper.

We used two kernel functions when implementing the computable plug-in estimator: the standard Gaussian kernel and a truncated Gaussian kernel that satisfies Assumption 1. Let $B(a,b)$ denote the open cube of width $2b$ centered at a. The truncated kernel has support of $B(0,3)$ and has the form

$$K(x) = \frac{\exp\{-(1/2)x^Tx\}}{\int_{B(0,3)}\exp\{-(1/2)x^Tx\}\,dx} \quad \text{for } x \in B(0,3).$$

Note that the standard Gaussian kernel does not have a compact support, so Assumption 1 used in Theorem 1 is not satisfied. Because the standard Gaussian kernel has tails that decay quickly to zero, we expect it behaves similarly to compactly supported kernels. Cadre (2006) commented that the compact support condition was used mainly to simplify proofs.

For the reproducing kernel-based methods such as OC-SVM, SVDD, and KMVCE, we used a radial basis function kernel $K(x,y) = \exp\{-\sigma\|x-y\|^2\}$. To choose the value of σ, we used a five-fold cross-validation on the training data set of normal events. We used the OC-SVM implementation from `R` package `e1071` (Dimitriadou et al. 2009) and the SVDD implementation from `DDTools` kindly provided by the author (Tax 2009). For the KMVCE, we implemented the algorithm given by Dolia et al. (2007). For the DDT, we used the authors' implementation of the method (Scott 2006) and followed the guideline in Scott and Nowak (2006) to select tuning parameters. In particular, we used the Rademacher penalty for the penalty function (Scott and Nowak 2006) and chose the maximum number of the cuts so that the dictionary size at the maximum depth is close to the training sample size, where the dictionary is a kind of data structure representing the tree nodes.

In all experiments, the target coverage probability of normal events is set to 0.95, which is equivalent to the type-I error rate of 0.05. However, for the KMVCE, we did not control its type-I error rate because there is no way to do so by the nature of the method.

For each data set, we constructed the training and test data sets as follows. The training data are randomly chosen to have two-thirds of data points from normal events; the test data set consists of the remaining data points from normal events and all data points from abnormal events. For all methods, the training data are used to construct the decision rule, and the test data are used to measure performance.

Comparison of methods was conducted in two ways. First, we analyzed the boundary of the MV-set produced by each method to see if the boundary compactly covers data points from normal events. Second, we compared the methods using three performance measures for detection capability: the type-I error, type-II error, and overall misclassification error. To reduce the variability of these performance measures, the splits to training and test data are repeated 50 times, and we take the average value of each performance measure over the 50 repetitions.

3.1. Artificial Data: Gaussian

We start with a simple set of artificial data of normal events generated from a Gaussian density. The objective of using this data set are to verify how well the MV-set estimated by various methods matches the T^2 control region under normality. We generated 1,000 data points of normal events from a Gaussian distribution defined on $\mathbb{R}^2$ and generated 600 data points of abnormal events from a mixture of three Gaussian distributions that overlap with the Gaussian distribution representing normal events. We used `R` package `mclust` (http://cran.r-project.org/web/packages/mclust/) to generate the random samples.

Figure 1 compares the detection boundaries of the MV-set estimation methods when the 1,000 data points of normal events are randomly split into a training and test data set in the ratio of 2:1. In this figure, the training data are plotted as solid dots, the test data of normal events as circles, and the test data of abnormal events as crosses. The solid contour lines describe the boundaries covering α portion of the normal events from various methods. The dotted contour lines represent the boundary of the T^2 control region. We observe that the boundaries of the two plug-in estimators are close to the T^2 boundary, much as expected, because of the normality assumption. The boundaries of the OC-SVM and KMVCE are also close to the T^2 boundary but more sensitive to some outliers. The boundary of the DDT is relatively too compact when compared to the T^2 boundary. It is also interesting to see that there is no big difference between the two plug-in methods using different kernel functions. This indicates that the compact support requirement on the kernel function could be relaxed in practice.

Table 1 compares the six methods in terms of their type-I, type-II, and overall misclassification errors, based on 50 random splits of normal events into training and test data sets. The two settings of the plug-in estimators and the SVDD show similar classification error rates to the T^2s. The other three methods have a higher type-I error or a higher type-II error. These results are consistent with the qualitative analysis on the boundaries from Figure 1.

Table 1. Error rates of various methods for Gaussian distribution.

Method	Type-I error (false alarm)	Type-II error (miss detection)	Misclassification
T^2	0.0492	0.0129	0.0258
OC-SVM	0.1088	0.0076	0.0435
SVDD	0.0544	0.0112	0.0265
KMVCE	0.0248	0.0404	0.0349
DDT	0.1104	0.0126	0.0578
Plug-in (Gaussian)	0.0612	0.0121	0.0295
Plug-in (truncated)	0.0617	0.0115	0.0293

3.2. Artificial Data: Gaussian Mixture

We generate 1,000 data points of normal events from a mixture of three Gaussian distributions on $\mathbb{R}^2$. We also generate 600 data points of abnormal events from another mixture of three Gaussian distributions.

Figure 2 shows the decision boundaries of the six methods, based on one random formation of the training and test data set. Both versions of the plug-in estimators represent the support of the Gaussian mixture distribution tightly, and we also reaffirmed that their boundaries are similar. We observe that the OC-SVM produces an irregular boundary to cover a few outliers. The SVDD and MVCE methods are better than the OC-SVM, but they are still sensitive to outliers, generating less-compact bounds and having higher type-II errors. Note that all three methods are based on RKHS. Boundaries in the RKHS might not be compact in the original space even though they are compact in the RKHS. The DDT method generates an inflexible boundary with some over-ballooned parts and some too-compact parts. As a result, it has relatively higher type-I errors. Table 2 shows that both versions of the plug-in estimators outperform other methods in terms of the misclassification rate.

3.3. Breast Cancer Detection

This data set is the Wisconsin Diagnostic Breast Cancer data set, available on the UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/). The data set contains

Table 2. Error rates of various methods for Gaussian mixture.

Method	Type-I error (false alarm)	Type-II error (miss detection)	Misclassification
OC-SVM	0.1061	0.0241	0.0532
SVDD	0.0533	0.0315	0.0392
KMVCE	0.0241	0.0919	0.0678
Dyadic tree	0.1349	0.0105	0.0680
Plug-in (Gaussian)	0.0579	0.0205	0.0338
Plug-in (truncated)	0.0582	0.0200	0.0335

Figure 1. Data and detection boundaries of various methods for normal events from a single Gaussian distribution (solid dots: training data; circles: test data from normal events; crosses: test data from abnormal events; solid lines: detection boundary from each method; dotted lines: T^2 boundary): (a) OC-SVM, (b) SVDD, (c) KMVCE, (d) DDT, (e) plug-in estimator with Gaussian kernel, (f) plug-in estimator with truncated Gaussian kernel.

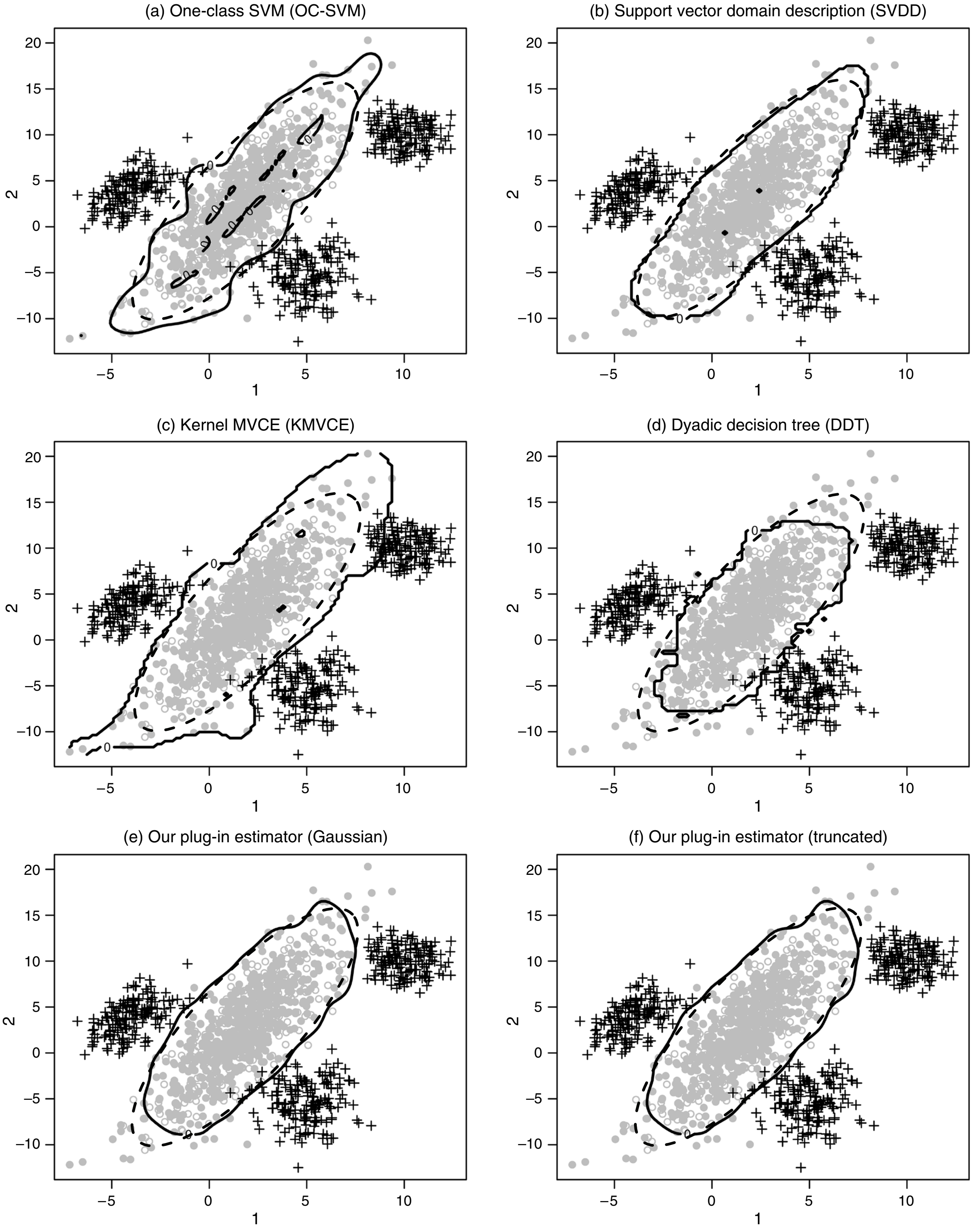
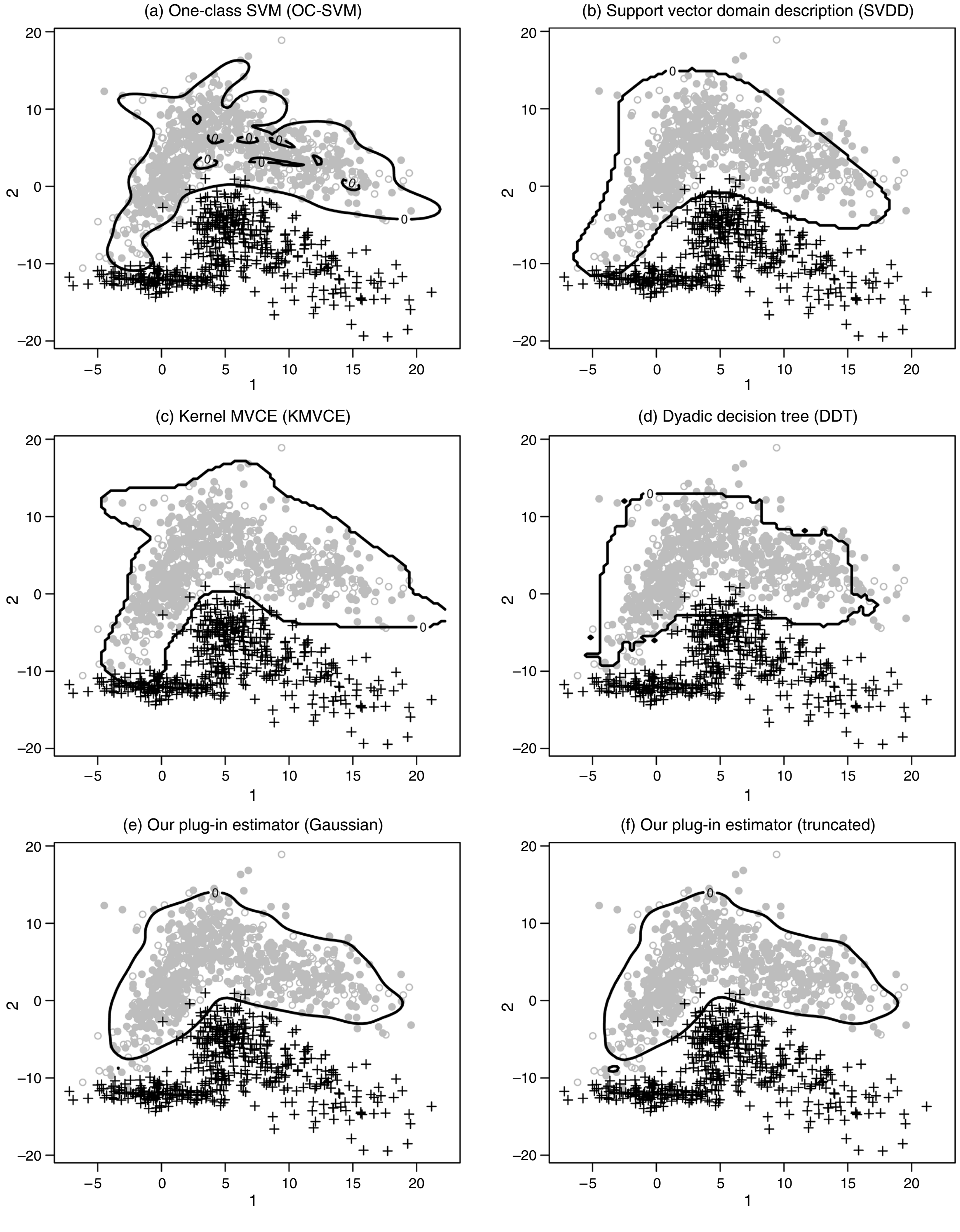


Figure 2. Data and detection boundaries of various methods for normal events from a Gaussian mixture (solid dots: training data; circles: test data from normal events; crosses: test data from abnormal events; solid lines: detection boundary from each method): (a) OC-SVM, (b) SVDD, (c) KMVCE, (d) DDT, (e) plug-in estimator with Gaussian kernel, (f) plug-in estimator with truncated Gaussian kernel.


(a) One-class SVM (OC-SVM)

(b) Support vector domain description (SVDD)

(c) Kernel MVCE (KMVCE)

(d) Dyadic decision tree (DDT)

(e) Our plug-in estimator (Gaussian)

(f) Our plug-in estimator (truncated)

Table 3. Error rates of various methods for breast cancer data.

Method	Type-I error (false alarm)	Type-II error (miss detection)	Misclassification
OC-SVM	0.1088	0.0490	0.0718
SVDD	0.0534	0.1163	0.0925
KMVCE	0.0490	0.0936	0.0766
DTT	0.0944	0.0033	0.0480
Plug-in (Gaussian)	0.0604	0.0045	0.0258
Plug-in (truncated)	0.0610	0.0045	0.0260

Table 4. Error rates of various methods for image segmentation data.

Method	Type-I error (false alarm)	Type-II error (miss detection)	Misclassification
OC-SVM	0.0947	0.0247	0.0283
SVDD	0.0707	0.0241	0.0265
KMVCE	0.0646	0.0621	0.0622
Dyadic tree	0.0670	0.0161	0.0201
Plug-in (Gaussian)	0.1017	0.0090	0.0138
Plug-in (truncated)	0.1028	0.0089	0.0138

699 instances of cancer cases, among which there are 458 benign instances and 241 malignant instances. Each case is represented by nine cancer-related attributes. The nine attributes are reduced to two principal components by performing the PCA.

Figure 3 shows the decision boundaries of the six methods, based on one random formation of the training and test data sets along with the data points. All three kernel methods overemphasize the long extruding region just to cover a few benign instances so that it misses many malignant cases. Their decision boundaries echo what we observed in the artificial data sets, namely, that the kernel methods appear sensitive to outlying training data points. The dyadic decision tree method and the two versions of the plug-in estimators show similar performance, all providing reasonably tight boundaries describing the benign data. This qualitative analysis on Figure 3 is reaffirmed by the quantitative results in Table 3. The plug-in estimators have the type-I error rates close to the target value 5%, and the type-II error rates are at most one-tenth of the type-II errors of the kernel methods. The DDT method performs similarly to the plug-in methods in terms of type-II error rates, but its type-I error rate is elevated due to its over-tight boundary.

3.4. Image Segmentation

This data set, also available from the UCI Machine Learning Repository, has 2,310 instances of the features describing one of the seven different images (a brick face, a sky, a foliage, a cement, a window, a path, and a grass image, respectively). The number of instances for each type of image is equal to 330. Each image is characterized by a set of 19 attributes such as colors, hue, saturation, and line density, among others. It was used as a data set for testing multiclass classification methods, but here we modify this data set to test novelty detection methods. We treat the 330 instances from the brick face image as the normal events and treat the rest of the images as abnormal events. The PCA is again used to reduce the 19-dimensional attribute to two principal components.

As shown in Figure 4, all methods generate good compact supports of the sample points from the brick face image. However, the three kernel methods again make extruded regions to cover a few outliers. Table 4 shows that the plug-in estimators have the smallest misclassification error rate among all methods.

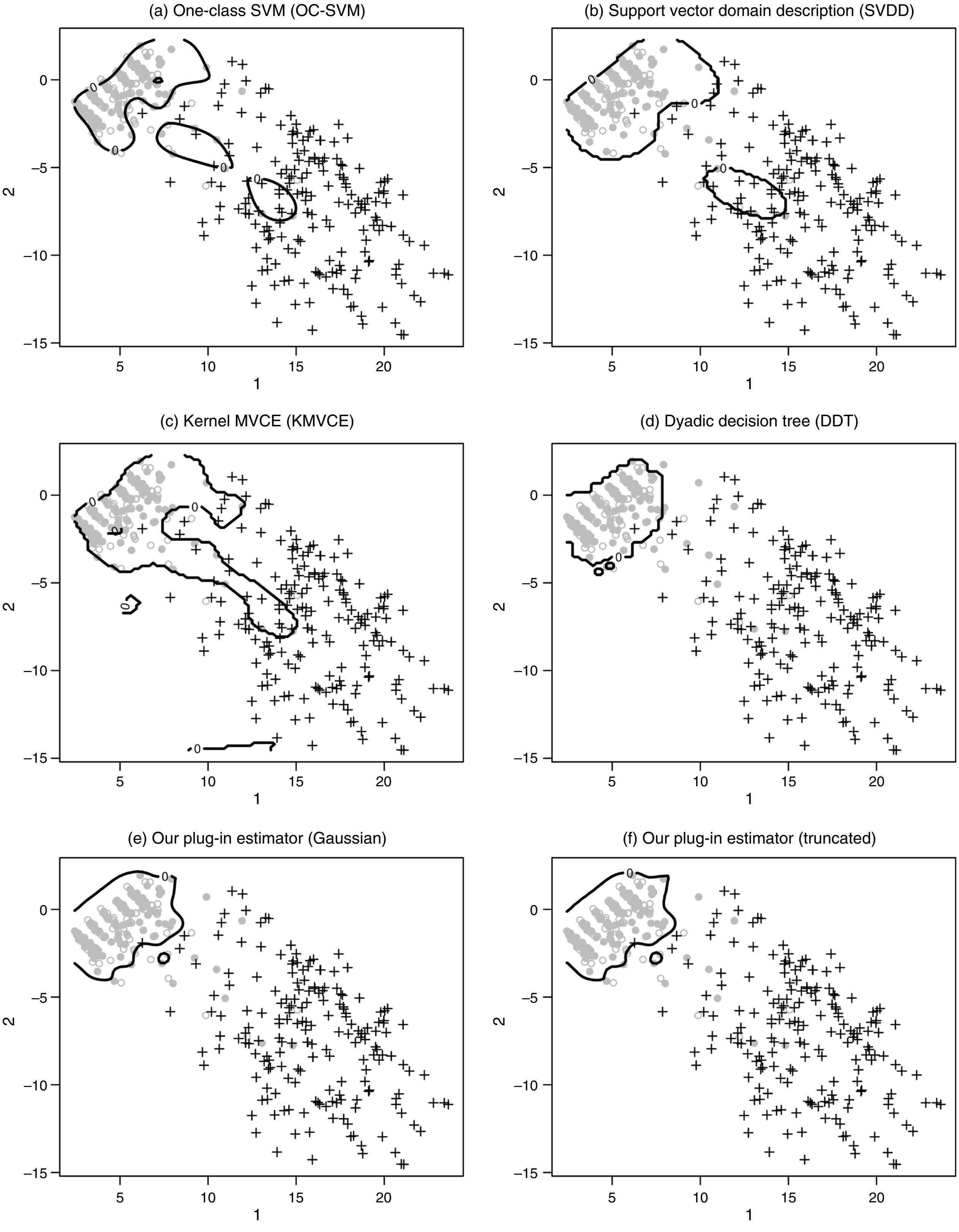
3.5. Ionosphere Data

This data set is radar data, available on the UCI Machine Learning Repository, collected by a system in Goose Bay, Labrador for the information on the ionosphere. It has 351 signals received from the system, and each signal consists of 17 pulse numbers. A pulse number is complex, so that it has a real part and an imaginary part, i.e., each signal consists of 34 attributes. The objective is to check if the received signals are good enough to contain meaningful information on the ionosphere. Among the 351 signals, 225 were labeled as “Good” and the remaining as “Not Good.” We treat the “Good” signals as normal events and the “Not Good” signals as abnormal events. As in the previous examples, the PCA is used to perform data reduction before novelty detection is conducted. In this case, the reduced dimension is five. For data dimension higher than two, it is difficult to present the graphical illustration. Therefore, we report only numerical performance measures in Table 5. In this case, the plug-in estimators have relatively higher type-I error rates but lower type-II error rates than other methods. The plug-in estimators perform the best in terms of misclassification error rate.

3.6. Speech (Vowel) Recognition

This data set, also available on the UCI Machine Learning Repository, consists of 11 vowel sounds pronounced by 15 individual speakers. Each speaker says each vowel 6 times, so we have a total of 990 vowel sounds. Each sound is represented by 10 sample points of the sound signal, so there are 10 attributes. To test the novelty detection methods, we treat the first 3 vowel sounds as normal events and the rest of the vowel sounds as abnormal events. The PCA reduces the 10 attributes to 4. As in the ionosphere data example, we cannot show the graphical illustration here but simply present the numerical performance measures in Table 6. Again, the plug-in methods have relatively higher type-I error rates but smaller type-II error rates. The plug-in methods are superior to other methods in terms of misclassification error rate.

Figure 3. Data and detection boundaries of various methods for the breast cancer data set (solid dots: training data; circles: test data from normal events; crosses: test data from abnormal events; solid lines: detection boundary from each method): (a) OC-SVM, (b) SVDD, (c) KMVCE, (d) DDT, (e) plug-in estimator with Gaussian kernel, (f) plug-in estimator with truncated Gaussian kernel.



(a) One-class SVM (OC-SVM)

(b) Support vector domain description (SVDD)

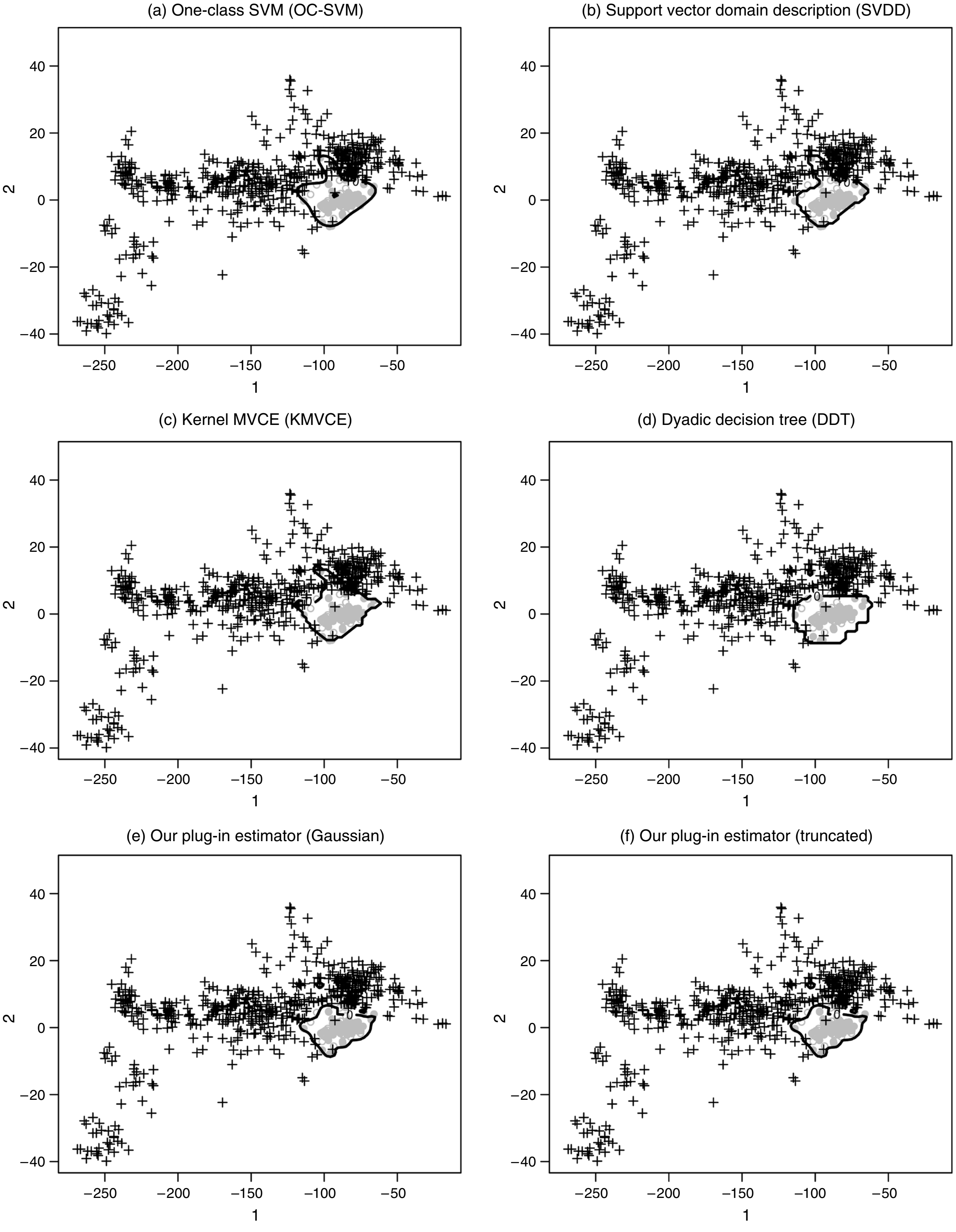
(c) Kernel MVCE (KMVCE)

(d) Dyadic decision tree (DDT)

(e) Our plug-in estimator (Gaussian)

(f) Our plug-in estimator (truncated)

Figure 4. Data and detection boundaries of various methods for the image segmentation data set (solid dots: training data; circles: test data from normal events; crosses: test data from abnormal events; solid lines: detection boundary from each method): (a) OC-SVM, (b) SVDD, (c) KMVCE, (d) DDT, (e) plug-in estimator with Gaussian kernel, (f) plug-in estimator with truncated Gaussian kernel.



(a) One-class SVM (OC-SVM)

(b) Support vector domain description (SVDD)

(c) Kernel MVCE (KMVCE)

(d) Dyadic decision tree (DDT)

(e) Our plug-in estimator (Gaussian)

(f) Our plug-in estimator (truncated)

Table 5. Error rates of various methods for ionosphere data.

Method	Type-I error (false alarm)	Type-II error (miss detection)	Misclassification
OC-SVM	0.1109	0.3292	0.2478
SVDD	0.1292	0.2676	0.2164
KMVCE	0.0048	0.9251	0.5817
DDT	0.1873	0.2260	0.2075
Plug-in (Gaussian)	0.1952	0.1524	0.1684
Plug-in (truncated)	0.1984	0.1457	0.1654

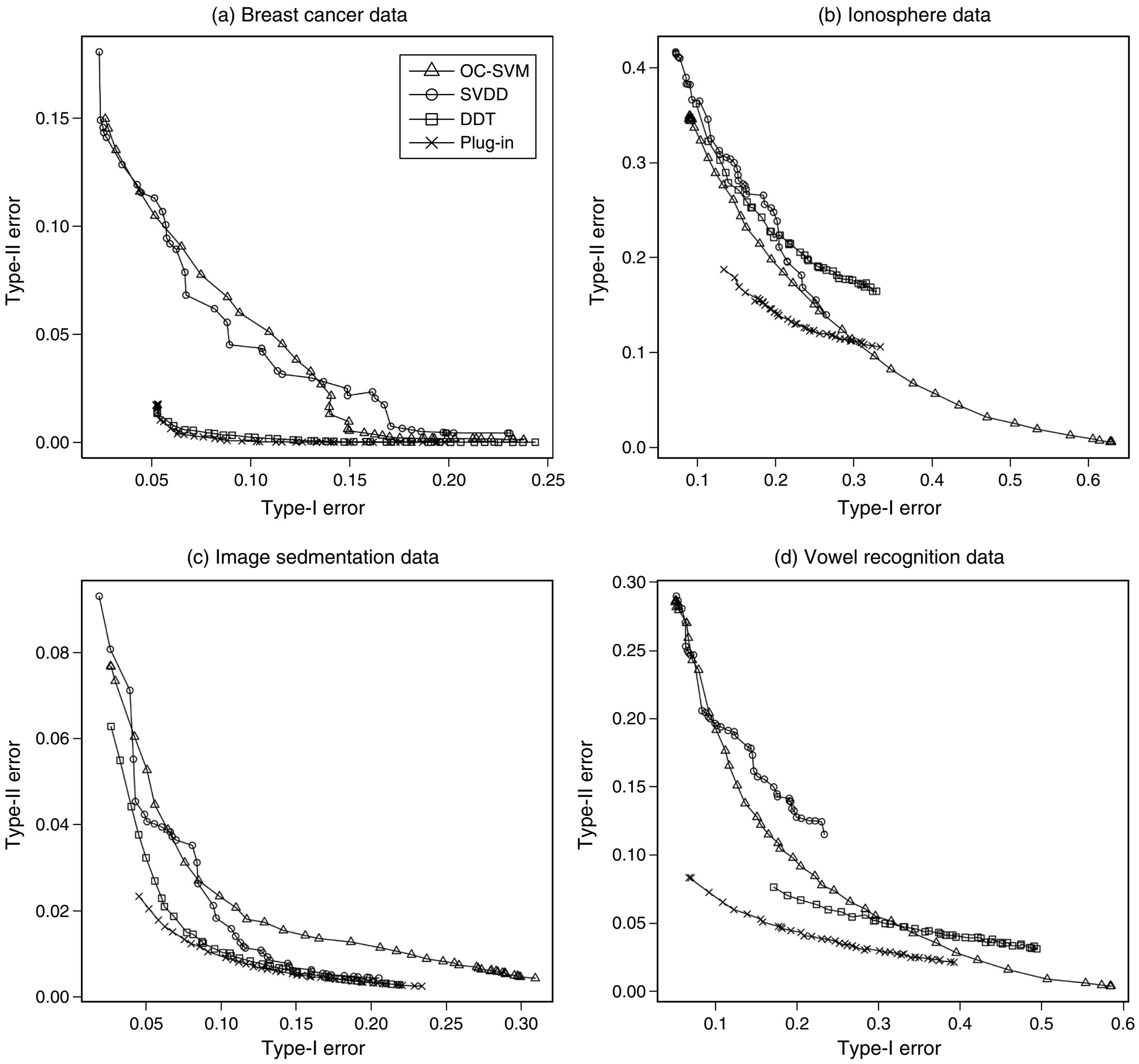
Table 6. Error rates of various methods for vowel recognition data.

Method	Type-I error (false alarm)	Type-II error (miss detection)	Misclassification
OC-SVM	0.0769	0.2331	0.2157
SVDD	0.0652	0.1999	0.1851
KMVCE	0.0489	0.4766	0.4290
DDT	0.2933	0.0518	0.0901
Plug-in (Gaussian)	0.1724	0.0502	0.0638
Plug-in (truncated)	0.1756	0.0476	0.0618

3.7. Operating Characteristic Analysis

So far, we compared the error rates of various methods for different data sets for a fixed target type-I error rate. In this subsection, we present an alternative performance comparison by using a modified operating characteristic curve (OC curve), where we plot the realized type-I error rate versus the realized type-II error rate. More precisely, we randomly form training and test data sets as in the

Figure 5. Operating characteristic curves for four real data sets: each curve plots the type-II error rates of each method given specific type-I error rates. (A curve going through the left and lower portion of the graph represents a better method.)


(a) Breast cancer data

(b) Ionosphere data

(c) Image sedmentation data

(d) Vowel recognition data

previous subsections, where the test data set contains one-third of the normal events and all abnormal events. For each target α value that ranges from 0.005 to 0.2 by step size 0.005, we apply all methods on the training data set and compute the type-I and type-II error rates on the test data set. We repeat 500 times the process of randomly forming the training and test data sets, applying the methods and computing the error rates. The error rates averaged over 500 repetitions are then used in the plot of a modified OC curve.

The modified OC curves for four methods (i.e., the OC-SVM, SVDD, DDT, and the plug-in estimator) are plotted in Figure 5 separately for the four real data sets. We do not consider KMVCE because there is no way to control its α level. The results of the plug-in estimator with the standard Gaussian kernel are omitted because they are almost the same as those based on the truncated Gaussian kernel. In Figure 5, the modified OC curves for the plug-in estimator are almost always located at the lower-left corner, representing smaller type-I and type-II error rates than other methods.

4. Conclusion

The computable plug-in estimator for novelty detection has the following features. First, the type-I error, i.e., the false alarm rate, can be directly controlled. This is different from other machine learning methods (such as the one-class SVM) or heuristics-based methods. Second, the type-II error of novelty detection is minimized. Third, the computable plug-in estimator does not require the normality assumption. When the normality assumption does hold, the plug-in estimator produces an acceptance region similar to the T^2 control region. Finally, the computable plug-in estimator has a practical algorithm that is straightforward to implement. The computable plug-in estimator is not a new methodology (Hyndman 1996), but lack of theoretical support and empirical evaluation makes it less well known. We hope this paper will improve the acceptance of this simple but powerful methodology as an alternative to many other novelty detection methods.

5. Electronic Companion

An electronic companion to this paper is available as part of the online version that can be found at http://or.journal.informs.org/. This electronic companion contains a proof of consistency of the computable plug-in estimator.

Acknowledgments

Park and Ding's research was partially supported by grants from the National Science Foundation (CMMI-0348150 and CMMI-0529026). Huang's research was partially supported by grants from the National Science Foundation (DMS-0606580 and DMS-0907170), the National Cancer Institute (CA57030), and award number KUS-CI-016-04, made by King Abdullah University of Science and Technology (KAUST). The authors are also grateful for the insightful comments and constructive suggestions made by the associate editor and two reviewers that helped improve the paper.

References

Baillo, A. 2003. Total error in a plug-in estimator of level sets. *Statist. Probab. Lett.* **65**(4) 411–417.

Cadre, B. 2006. Kernel estimation of density level sets. *J. Multivariate Anal.* **97**(4) 999–1023.

Dimitriadou, E., K. Hornik, F. Leisch, D. Meyer, A. Weingessel. 2009. Support vector machines—The interface to libsvm in package e1071. http://cran.r-project.org/web/packages/e1071/index.html. Version 1.5.19.

Dolia, A. N., C. J. Harris, J. S. Shawe-Taylor, D. M. Titterington. 2007. Kernel ellipsoidal trimming. *Computational Statist. Data Anal.* **52**(1) 309–324.

Dolia, A. N., S. F. Page, N. M. White, C. J. Harris. 2004. D-optimality for minimum volume ellipsoid with outliers. *Proc. 7th All Ukranian Conf. Signal/Image Processing Pattern Recognition, Kiev, Ukraine*, 73–76.

Garcia, J. N., Z. Kutalik, K.-H. Cho, O. Wolkenhauer. 2003. Level sets and minimum volume sets of probability density functions. *Internat. J. Approximate Reasoning* **34**(1) 25–47.

Guh, R.-S., F. Zorriassatine, J. D. T. Tannock, C. O'Brien. 1999. On-line control chart pattern detection and discrimination—A neural network approach. *Artificial Intelligence Engrg.* **13**(4) 413–425.

Hayton, P., B. Schölkopf, L. Tarassenko, P. Anuzis. 2000. Support vector novelty detection applied to jet engine vibration spectra. *Proc. Adv. Neural Inform. Processing Systems*, Vol. 13. MIT Press, Cambridge, MA, 946–952.

Hoffmann, H. 2007. Kernel PCA for novelty detection. *Pattern Recognition* **40**(3) 863–874.

Hyndman, R. J. 1996. Computing and graphing highest density regions. *Amer. Statistician* **50**(2) 120–126.

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

Mardia, K. V., J. T. Kent, J. M. Bibby. 1980. *Multivariate Analysis*. Academic Press, San Diego, 66–76.

Markou, M., S. Singh. 2003a. Novelty detection: A review—Part 1: Statistical approaches. *Signal Processing* **83**(12) 2481–2497.

Markou, M., S. Singh. 2003b. Novelty detection: A review—Part 2: Neural network based approaches. *Signal Processing* **83**(12) 2499–2521.

Montgomery, D. C. 1997. *Introduction to Statistical Quality Control*. John Wiley & Sons, New York.

Polonik, W. 1995. Measuring mass concentrations and estimating density contour clusters—An excess mass approach. *Ann. Statist.* **23**(3) 855–881.

Sanseverino, C. M. R., E. Zio. 2007. A support vector machine integrated system for the classification of operation anomalies in nuclear components and systems. *Reliability Engrg. System Safety* **92**(5) 593–600.

Schölkopf, B., J. C. Platt, J. C. Shawe-Taylor, A. J. Smola, R. C. Williamson. 2001. Estimating the support of a high-dimensional distribution. *Neural Comput.* **13**(7) 1443–1471.

Scott, C. D. 2006. Matlab/mex code for solving several set estimation problems. http://www.eecs.umich.edu/~cscott/code.html.

Scott, C. D., R. D. Nowak. 2006. Learning minimum volume sets. *J. Machine Learn. Res.* **7** 665–704.

Tarassenko, L., P. Hayton, N. Cerneaz, M. Brady. 1995. Novelty detection for the identification of masses in mammograms. *Artificial Neural Networks, 1995, Fourth Internat. Conf., Cambridge, UK,* 442–447.

Tax, D. M. J. 2009. Ddtools, the data description toolbox for Matlab. http://ict.ewi.tudelft.nl/~davidt/dd_tools.html. Version 1.7.3.

Tax, D. M. J., R. P. W. Duin. 1999. Support vector domain description. *Pattern Recognition Lett.* **20**(11–13) 1191–1199.

Worden, K., S. G. Pierce, G. Manson, W. R. Philp, W. J. Staszewski, B. Culshaw. 2000. Detection of defects in composite plates using Lamb waves and novelty detection. *Internat. J. Systems Sci.* **31**(11) 1397–1409.