R1SVM: a Randomised Nonlinear Approach to Large-Scale Anomaly Detection

Sarah M. Erfani[*†], **Mahsa Baktashmotlagh**[†], **Sutharshan Rajasegarar**[†]
Shanika Karunasekera[*†], **Chris Leckie**[*†]
[†]NICTA Victoria Research Laboratory
[*]Department of Computing and Information Systems,
The University of Melbourne, Australia

Abstract

The problem of unsupervised anomaly detection arises in a wide variety of practical applications. While one-class support vector machines have demonstrated their effectiveness as an anomaly detection technique, their ability to model large datasets is limited due to their memory and time complexity for training. To address this issue for supervised learning of kernel machines, there has been growing interest in random projection methods as an alternative to the computationally expensive problems of kernel matrix construction and support vector optimisation. In this paper we leverage the theory of nonlinear random projections and propose the *Randomised One-class SVM (R1SVM)*, which is an efficient and scalable anomaly detection technique that can be trained on large-scale datasets. Our empirical analysis on several real-life and synthetic datasets shows that our randomised 1SVM algorithm achieves comparable or better accuracy to deep autoencoder and traditional kernelised approaches for anomaly detection, while being approximately 100 times faster in training and testing.

Introduction

Unsupervised anomaly detection (also known as outlier detection) plays a significant role in a variety of applications, such as fraud detection, network intrusion detection and fault diagnosis. One-class Support Vector Machines (1SVMs) (Schölkopf et al. 2001; Tax and Duin 2004) have proven to be a very effective unsupervised learning method to construct highly accurate classifiers for anomaly detection. However, 1SVMs are often impractical for use on very large datasets due to the computational and memory complexity of their underlying optimisation problem during training (Vapnik 1998; Vishwanathan, Smola, and Murty 2003; Bengio and LeCun 2007). Recently, there has been growing interest in randomised approaches to improve the efficiency of kernel methods for supervised learning of SVMs (Rahimi and Recht 2007; 2009). In this paper, we build on the theory of nonlinear random projections in order to accelerate the training of 1SVMs, and propose a new form of anomaly detector called *Randomised One-class SVM* (R1SVM). We show that R1SVM can achieve comparable or better accuracy than an existing 1SVM method, while reducing training and testing time by up to two orders of magnitude.

A key challenge in anomaly detection is how to characterise the distribution of "normal" (i.e., non-anomalous) data. This is particularly challenging when the process that generated the normal data has an unknown, potentially complex underlying distribution. 1SVM approaches have had considerable success in addressing this modelling task by using a kernel function to implicitly map the data from the input space to a higher dimensional feature space, in which a relatively simple model such as a hyperplane (Schölkopf et al. 2001), hypersphere (Tax and Duin 2004) or hyperellipsoid (Wang, Yeung, and Tsang 2006; Rajasegarar et al. 2010) can be used to characterise normal observations.

A practical limitation of 1SVM approaches is their computational and memory complexity for training. In a dataset with n records, each with d dimensions, training using a nonlinear kernel requires $O(dn^2)$ computational complexity, as well as $O(dn^2)$ memory complexity for the kernel matrix (Tax and Duin 2004). This limits the utility of 1SVM in applications involving large datasets. While training can be performed on a smaller sample of the training data, this can reduce the accuracy of the 1SVM due to the sparse sampling of the underlying distribution, particularly in applications that involve high dimensional input spaces.

Recently, there has been significant progress in using randomised features in conjunction with linear algorithms to reveal nonlinear patterns in data. In particular, a nonlinear, randomised variant of component analysis methods such as Principal Component Analysis (RPCA) and Canonical Correlation Analysis (RCCA) has been proposed (Lopez-Paz et al. 2014). These randomised variants have been applied to the tasks of regression and classification of large datasets, and exhibited significant savings in computation time while incurring little or no loss in accuracy.

In this paper, we propose a novel application of randomised methods by deriving a highly scalable algorithm for anomaly detection based on training a linear one-class SVM using randomised, nonlinear features. By using randomised features rather than finding a set of optimised support vectors, we can substantially reduce the cost of training our one-class SVM. We provide extensive empirical testing to show that our randomised 1SVM method achieves substantial improvement in both computational complexity and

accuracy over exact kernel methods.

To the best of our knowledge this is the first attempt to exploit nonlinear random features in kernel-based methods for anomaly detection. In addition to significantly reducing computational complexity, we show that our randomised 1SVM algorithm achieves comparable or better accuracy compared to autoencoder and kernelised approaches to anomaly detection. We postulate that this improvement in accuracy is due to the implicit regularisation induced by randomness, as well as an improvement in the separation between normal and anomalous data points when compared in the nonlinear feature space. By improving the efficiency of training 1SVMs in this way, we believe it will be possible to apply anomaly detection to data-intensive applications in resource constrained environments, such as wireless sensor networks.

Related Work and Background

While numerous 1SVM formulations using nonlinear kernel have been proposed in the literature (Schölkopf et al. 2001; Tax and Duin 2004; Bottou and Lin 2007), a common feature of many formulations is the solution of a quadratic programming (QP) problem. In particular, these kernel-based methods rely on the calculation of a kernel matrix over all pairs of data points, which limits the scalability of training 1SVMs on large datasets. This can also limit the effectiveness of 1SVMs on high dimensional input spaces, given the need to have a sufficiently large training dataset that spans the variation in the high dimensional space.

Existing approaches to address the scalability problems of SVMs can be classified into two general categories. One category comprises hybrid and complimentary methods for SVMs, which are used to preprocess the data prior to processing by the SVM. For example, clustering (Sun et al. 2004), dimension reduction techniques such as PCA or KPCA (Cao et al. 2003; Subasi and Ismail Gursoy 2010), and deep belief networks (Bengio and LeCun 2007) are some of the most well-known approaches. Although these approaches play an important role in building the model, they do not directly address the scalability of the SVM itself. The second category includes methods that aim to alleviate the QP problem of kernel machines. A more heuristic approach is to reduce the size of the QP problem by breaking it into smaller pieces, for example by using chunking (Vapnik 1998; Sonnenburg et al. 2006), decomposition (Vishwanathan, Smola, and Murty 2003; Joachims 1999), or Sequential Minimal Optimisation (SMO) (Platt 1999). An alternative to enhance the computational efficiency of SVMs is to instead use an approximation of QP (Fung and Mangasarian 2001). A more radical approach is to totally avoid the QP problem, and obtain the solution through a fast iterative scheme (Fung and Mangasarian 2003; Yang, Duraiswami, and Davis 2005).

To reduce the memory and computational complexity, a popular approach is to obtain a low-rank approximation of the kernel matrix. Selective sampling (or active learning) methods iterate through the training data and sample a small subset of the records that are near the boundary in the feature space with higher probabilities, e.g., (Tong and Koller 2002), or see (Settles 2010) for a survey. To avoid the computational cost of processing the whole dataset, Lee and Mangasarian (2001) propose the use of random sampling to obtain a result that is close to the original SVM.

A more recent trend explores the use of randomisation, such as linear random projection (Blum 2006) as a substitute for the computationally expensive cost of kernel matrix construction. An early example is the work of Achlioptas, McSherry, and Schölkopf (2002), which replaces the kernel function by a randomised kernel to speedup KPCA. The work of Rahimi and Recht (2007; 2009) made a breakthrough in this approach. They replicated an RBF kernel by randomly projecting the data to a lower dimensional space and then used linear algorithms. Random projection avoids the computational complexity of traditional optimisation methods needed for nonlinear kernels. More recently, Kar and Karnick (2012), and Hamid et al. (2014) have extended the method of Rahimi and Recht to other types of kernels, e.g., dot-product and polynomial kernels.

In this paper, we build on the work of Rahimi and Recht by developing a model for using randomised projection in the context of unsupervised learning of a 1SVM. In the next section we describe our proposed randomised 1SVM scheme.

Proposed Approach — R1SVM

In this section we present our Randomised 1SVM (R1SVM) model for anomaly detection. We begin by recalling a few key aspects of one-class SVMs, and then introduce the use of nonlinear random projections for detecting anomalies in large-scale data. Random projections have been utilised mainly in distance-based classification or data reconstruction schemes to speedup the search, as it approximately preserves L_2 distances among a set of points. Thus instead of performing the search in a high-dimensional space, the search is conducted in a space of reduced dimension but on a larger neighbourhood. Note that in our context, ultimately, our goal is anomaly detection. Therefore, we are not necessarily interested in deriving a representation that allows for the best classification or reconstruction of the data, but we rather seek to find a model of the underlying distribution of the data which can then be used to detect anomalies.

One-class SVM (1SVM)

Let $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n]$ be the $d \times n$ matrix containing n training points $\boldsymbol{x}_i \in \mathbb{R}^d$ of one specific class, and let $\boldsymbol{\phi}$ be a feature map $\boldsymbol{X} \to \mathcal{H}$ such that the dot product in $\mathcal{H}$ can be computed using some kernel $k(\boldsymbol{x}, \boldsymbol{x'}) =< \boldsymbol{\phi}(\boldsymbol{x}), \boldsymbol{\phi}(\boldsymbol{x'}) >_{\mathcal{H}}$.

A One-class Support Vector Machine (1SVM) (Schölkopf et al. 2001; Manevitz and Yousef 2002) finds anomalies by first projecting the data to the feature space $\mathcal{H}$, and then finding a hyperplane that best separates the data from the origin. In other words, the decision function in the 1SVM returns +1 in a region where most of the data points occur (i.e., where the probability density is high), and returns -1 elsewhere.

Defining a family of sets $C_{\boldsymbol{s},\rho} = \{\boldsymbol{x} | f_{\boldsymbol{s},\rho}(\boldsymbol{x}) > 0\}$, the 1SVM estimates a function $f_{\boldsymbol{s},\rho}(\boldsymbol{x}) = \operatorname{sgn}(\boldsymbol{s}.\boldsymbol{\phi}(\boldsymbol{x}) - \rho)$ that maximises the distance of all the data points (in the feature

space $\mathcal{F}$) from the hyperplane to the origin, parameterised by a weight vector $\boldsymbol{s}$ and an offset ρ.

Thus, the resulting binary function $f_{\boldsymbol{s},\rho}(\boldsymbol{x})$ can be estimated by minimising the regularised risk:

$$R^{reg}[f_{\boldsymbol{s},\rho}(\boldsymbol{x})] = R^{emp}[f_{\boldsymbol{s},\rho}(\boldsymbol{x})] + \frac{1}{2}\|f_{\boldsymbol{s},\rho}(\boldsymbol{x})\|^2_{\mathcal{H}} \quad (1)$$

where $R^{emp}(.)$ is the empirical risk and $\frac{1}{2}\|f_{\boldsymbol{s},\rho}(\boldsymbol{x})\|^2_{\mathcal{H}}$ is the regulariser. The empirical risk is the average loss and can be written as

$$R^{emp}[f_{\boldsymbol{s},\rho}(\boldsymbol{x})] \equiv \frac{1}{n}\sum_{i=1}^{n} c(f_{\boldsymbol{s},\rho}(\boldsymbol{x}_i), y_i) \quad (2)$$

where $c(f_{\boldsymbol{s},\rho}(\boldsymbol{x}_i), y_i)$ is the loss function that penalises the deviation between the prediction $f_{\boldsymbol{s},\rho}(.)$ and the label y, i.e., this captures the cost of the errors caused when $f_{\boldsymbol{s},\rho}(.)$ is negative on training vectors.

Replacing $\|f_{\boldsymbol{s},\rho}(.)\|^2_{\mathcal{H}}$ with a *maximum margin* regulariser $\|\boldsymbol{s}\|^2$ to penalise complex regions, we can setup the following quadratic program for 1SVM:

$$\begin{aligned} \min_{\boldsymbol{s},\xi_i,\rho} \quad & \tfrac{1}{2}\|\boldsymbol{s}\|^2 + \tfrac{1}{\nu n}\textstyle\sum_{i=1}^{n}\xi_i - \rho \\ s.t. \quad & (\boldsymbol{s}.\boldsymbol{\phi}(\boldsymbol{x}_i)) \geq \rho - \xi_i, \\ & \xi_i \geq 0 \end{aligned} \quad (3)$$

where $\nu \in (0, 1]$ is a regularisation parameter that controls the fraction of anomalies and the fraction of support vectors, and ξ_i are the slack variables that allow some of the data vectors to lie on the wrong side of the hyperplane. By introducing the Lagrange multipliers, we arrive at the following quadratic program, which is the dual of the primal program in (3):

$$\begin{aligned} \min_{\alpha} \quad & \tfrac{1}{2}\textstyle\sum_{ij}\alpha_i\alpha_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ s.t. \quad & 0 \leq \alpha_i \leq \tfrac{1}{\nu n}, \\ & \textstyle\sum_1^n \alpha_i = 1 \end{aligned} \quad (4)$$

where α_i are the Lagrange multipliers. Further, $\boldsymbol{s} = \sum_i \alpha_i \boldsymbol{\phi}(\boldsymbol{x}_i)$. Using the Karush-Kuhn-Tucker optimality conditions (KKT conditions) the data vectors can be characterised in terms of whether they fall below, above, or on the hyperplane boundary in the feature space depending on the corresponding α_i values. Data vectors with positive α_i values are the support vectors. Further, for $0 < \alpha_i < 1/\nu n$, the data vectors fall on the hyperplane and hence ρ can be recovered using these vectors, vis-a-vis $\rho =< \boldsymbol{s}, \boldsymbol{\phi}(\boldsymbol{x}_i) >= \sum_j \alpha_j k(\boldsymbol{x}_j, \boldsymbol{x}_i)$. Therefore, the decision function can now be written as

$$\begin{aligned} f_{\boldsymbol{s},\rho}(\boldsymbol{x}) &= \mathrm{sgn}(\boldsymbol{s}.\boldsymbol{\phi}(\boldsymbol{x}) - \rho) \\ &= \mathrm{sgn}(\alpha_i k(\boldsymbol{x}_i, \boldsymbol{x}) - \rho) \end{aligned} \quad (5)$$

The solution to the quadratic program in (4) is characterised by the parameter $\nu \in (0, 1]$, which sets an upper bound on the fraction of anomalies (training examples regarded as out-of-class) and a lower bound on the number of training examples used as support vectors.

The computational complexity of 1SVM using an SMO solver is approximately $O(dn^2)$ for the RBF kernel (Vempati et al. 2010), and $O(dn)$ for a linear kernel with n being the number of samples and d the number of dimensions in feature space. However, it has been noted that when 1SVM is used with a linear kernel, it introduces a bias to the origin. This problem can be removed by using an RBF kernel, which has a higher computational complexity associated with the higher dimensional kernels, thus making it cumbersome for processing with large scale data.

In order to overcome this limitation, in the next section, we propose to exploit nonlinear random projections inside a linear 1SVM, which serves as a good approximation of a nonlinear 1SVM.

Randomised 1SVM

We propose R1SVM, a nonlinear randomised variant of 1SVM, which applies the original linear 1SVM method on a randomised nonlinear projection of the data. We first discuss how to generate the nonlinear random features from the original data, and then we show how to employ these features to detect anomalies using a linear 1SVM. This approach eliminates the need to deal with large kernel matrices for large datasets, consequently reducing the computational complexity while achieving comparable or better anomaly detection performance than a traditional 1SVM (as shown in the Evaluation section).

Generating Nonlinear Random Features Consider the problem of fitting a function f (note that the subscripts s and ρ in $f_{s,\rho}$ are omitted for brevity) to the data set $\{\boldsymbol{x}_i, y_i\}$, where y_i values are always set to 1 for the one-class problem. This fitting problem consists of finding f that minimises the empirical risk in equation (2). For the 1SVM problem, the loss function $c(y, y')$ is of the form $c(y, y') = max(0, 1 - yy')$. Using the kernel function, the function $f(\boldsymbol{x}) = sgn(\boldsymbol{s}.\boldsymbol{\phi}(\boldsymbol{x}) - \rho)$ becomes $f(\boldsymbol{x}) = \sum_i^n \alpha_i k(\boldsymbol{x}, \boldsymbol{x}_i)$. Therefore, in the general form, the function $f(\boldsymbol{x})$ can be written as $f(\boldsymbol{x}) = \sum_{i=1}^{\infty} \alpha(\boldsymbol{s}_i)\boldsymbol{\phi}(\boldsymbol{x}; \boldsymbol{s}_i)$, where $\boldsymbol{\phi}$ are parameterised by vector $\boldsymbol{s}$ and weighted by α_i. By jointly optimising over $\boldsymbol{s}$ and α_i, in a greedy manner, the solution can be found (Rahimi and Recht 2007). However, this is computationally intensive. Rahimi and Recht (2009) have proved that this nonlinear optimisation problem over $(\boldsymbol{\alpha}, \boldsymbol{s}_1, \cdots, \boldsymbol{s}_n)$ in f, can be solved by randomly sampling the $\boldsymbol{s}_i \in \mathbb{R}^d$ from a data-independent distribution $p(\boldsymbol{s})$ and creating k-dimensional random features $\boldsymbol{z}(\boldsymbol{X}) = [\boldsymbol{z}_1 \cdots \boldsymbol{z}_k]$, where $\boldsymbol{z}_i = [\cos(\boldsymbol{s}_i^T\boldsymbol{x}_1 + b_i), \cdots, \cos(\boldsymbol{s}_i^T\boldsymbol{x}_n + b_i)]$ are Fourier based random features. For more details refer to (Rahimi and Recht 2007). Thus, we arrive at the following simplified optimisation problem:

$$\begin{aligned} \min_{\boldsymbol{\alpha}\in\mathbb{R}^k} \quad & \tfrac{1}{n}\textstyle\sum_i c(\boldsymbol{\alpha}^T\boldsymbol{z}_i, y_i) \\ s.t. \quad & \|\boldsymbol{\alpha}\|_\infty \leq B \end{aligned} \quad (6)$$

where B is a regularisation constant. Furthermore, it is shown by Rahimi and Recht (2009) that using randomly selected features in nonlinear spaces causes only *bounded* error compared to using optimised features:

Theorem 1. *Let p be a distribution on Ω and $|\phi(\boldsymbol{x};\boldsymbol{s})| \leq 1$. Let $\mathcal{F} = \left\{ f(\boldsymbol{x}) = \int_{\delta} \alpha(\boldsymbol{s})\phi(\boldsymbol{x};\boldsymbol{s})d\boldsymbol{s} \mid |\alpha(\boldsymbol{s})| \leq Bp(\boldsymbol{s}) \right\}$. Draw $\boldsymbol{s}_1, \cdots, \boldsymbol{s}_k$ iid from p. Further let $\lambda > 0$, and c be some L-Lipschitz loss function, then the function $f_k(\boldsymbol{x}) = \sum_{i=1}^{k} \alpha_i \phi(\boldsymbol{x};\boldsymbol{s}_i)$ minimises the empirical risk $c(f_k(\boldsymbol{x}), y)$ has a distance from the c-optimal estimator in F bounded by*

$$E_p[c(f_k(\boldsymbol{x}), y)] - \min_{f \in \mathcal{F}} E_p[c(f(\boldsymbol{x}), y)] \leq O\left(\frac{LB}{\sqrt{n}} + \frac{1}{\sqrt{k}} LB \sqrt{\log \frac{1}{\delta}} \right) \quad (7)$$

with a probability of at least $1 - 2\delta$.

The convergence rate of our randomised R1SVM to its original kernel 1SVM version can be expressed by the following theorem (Lopez-Paz et al. 2014):

Theorem 2. *Given the data $\boldsymbol{X} \in \mathbb{R}^{n \times d}$, a shift invariant kernel k, a kernel matrix $\boldsymbol{K}_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and its approximation $\hat{\boldsymbol{K}}$ using k random features, it can be proven that*

$$\mathbb{E}\|\hat{\boldsymbol{K}} - \boldsymbol{K}\| \leq \sqrt{\frac{3n^2 \log n}{k}} + \frac{2n \log n}{k}. \quad (8)$$

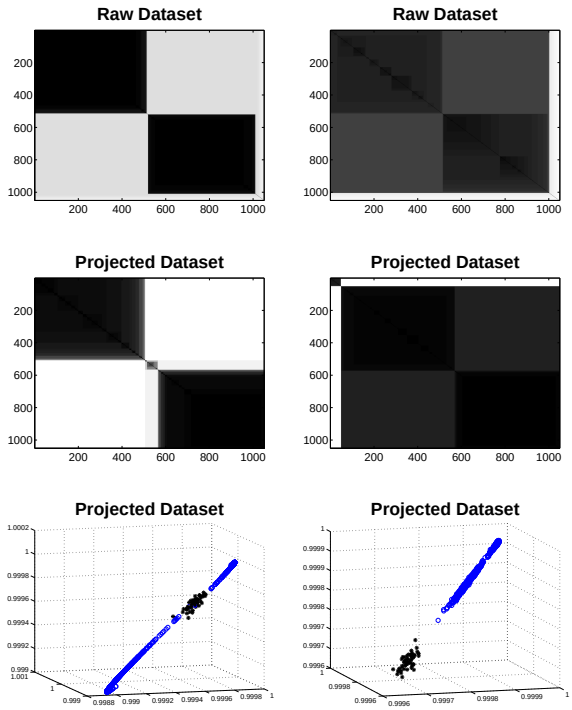
The proof to this theorem can be found in (Lopez-Paz et al. 2014).

Evaluation and Discussion

In this section, we evaluate the effectiveness of our R1SVM method for anomaly detection by conducting the following two experiments. First, we empirically explore the impact of random projections on the separability of normal data records from anomalous records. Then we compare the performance of R1SVM in terms of accuracy, training and testing time, with a 1SVM scheme called Support Vector Data Decomposition (SVDD), and a deep autoencoder (AE).

Experimental setup: For visualisation purposes, in the first experiment, we used a tool called *improved Visual Assessment of cluster Tendency* (iVAT) (Wang et al. 2010), which helps visualise the possible number of clusters in, or the cluster tendency of, a set of objects. iVAT reorders the dissimilarity matrix of the given set of objects so that it can display any clusters as dark blocks along the diagonal of the image.

In the second experiment we used the `svdd` implementation from Dd-tools (Tax 2013) as the one-class SVM method. Note that the hypersphere-based SVDD model using an RBF kernel is equivalent to a hyperplane-based 1SVM model. In the case of the autoencoder, we implemented a basic autoencoder including five-layers with tied weights and a sigmoid activation function for both the encoder and decoder. The training is conducted in mini-batches of $q = 100$ records. Initially the autoencoder was trained based on greedy layer-wise pre-training (i.e., training one layer at a time) to extract features, and then using these features to train the next layer. Training a network implies finding parameters (network weight and bias) that minimise the reconstruction error between the inputs $\boldsymbol{x}$ and the reconstruction of $\boldsymbol{x}$ at the output $\bar{\boldsymbol{x}}$, $l(\boldsymbol{x}, \bar{\boldsymbol{x}}) = \| \boldsymbol{x} - \bar{\boldsymbol{x}} \|^2$.



a) Banana dataset b) HAR dataset

Figure 1: Demonstration of the effect of nonlinear projection on normal and anomalous records.

Once the network was trained, the learned parameter values were used as initialisation values of a multilayer perceptron (MLP) with the same number of inputs and outputs. Then the network was fine-tuned by gradient descent to adjust the parameters. The whole process of pre-training and fine-tuning was performed in an unsupervised manner for anomaly detection. Anomalies then can be identified by the autoencoder based on the history of the squared error between the inputs and outputs for the training records. Let e be the reconstruction error of $\boldsymbol{x}_i \in \mathbf{X}$, where $i = 1, \ldots, n$. If the reconstruction error for a test sample is larger than the threshold $\tau = \mu(\mathsf{e}) + 3\,\sigma(\mathsf{e})$, the record is identified as anomalous, otherwise it is identified as normal.

Datasets: The experiments are conducted on six real-life datasets from the UCI Machine Learning Repository: (i) Forest (ii) Adult (iii) Gas Sensor Array Drift (Gas), (iv) Opportunity Activity Recognition (OAR), (v) Daily and Sport Activity (DSA), and (vi) Human Activity Recognition using Smartphones (HAR), with dimensionalities of 54, 123,128, 242, 315[1] and 561 features, respectively. We also use two synthetic datasets. One is a "Smiley" dataset, generated from a mixture of two compact Gaussians and an arc shaped distribution. The dataset contains 20 dimensions and in any two dimensions the components of the face are ran-

[1]DSA is a large dataset comprising the time series measurements from 45 wearable sensors for 19 activities. We select a portion of the time series for each of the first 7 activities, yielding a total of 315 concatenated time series features.

Table 1: Comparison of AUC, train and test time of R1SVM with SVDD and autoencoder (AE).

		$SVDD$			AE			$R1SVM$		
Dataset	Features	AUC	Train time	Test time	AUC	Train Time	Test time	AUC	Train time	Test time
Smiley	20	0.85	1.58	2.3×10^{-2}	**0.98**	0.61	1.1×10^{-3}	**0.98**	$\mathbf{7.8 \times 10^{-3}}$	$\mathbf{1.0 \times 10^{-5}}$
Forest	54	0.97	2.12	2.2×10^{-2}	**0.99**	0.47	1.5×10^{-3}	**0.99**	$\mathbf{5.3 \times 10^{-3}}$	$\mathbf{1.2 \times 10^{-5}}$
Banana	100	0.92	2.75	2.4×10^{-2}	**0.99**	0.79	2.6×10^{-3}	**0.99**	$\mathbf{5.3 \times 10^{-3}}$	$\mathbf{1.2 \times 10^{-5}}$
Adult	123	0.87	2.83	2.7×10^{-2}	**0.99**	0.65	2.6×10^{-3}	**0.99**	$\mathbf{5.3 \times 10^{-3}}$	$\mathbf{1.3 \times 10^{-5}}$
Gas	128	0.91	1.06	2.6×10^{-2}	**0.98**	0.99	2.2×10^{-3}	**0.98**	$\mathbf{2.0 \times 10^{-3}}$	$\mathbf{1.0 \times 10^{-5}}$
OAR	242	0.91	1.08	2.5×10^{-2}	**0.97**	0.67	5.8×10^{-3}	**0.97**	$\mathbf{2.3 \times 10^{-3}}$	$\mathbf{1.1 \times 10^{-5}}$
DSA	315	0.84	1.13	3.4×10^{-2}	**0.98**	0.63	5.5×10^{-3}	**0.98**	$\mathbf{5.0 \times 10^{-3}}$	$\mathbf{1.5 \times 10^{-5}}$
HAR	561	0.88	2.11	4.2×10^{-2}	**0.99**	1.02	1.4×10^{-2}	**0.99**	$\mathbf{8.4 \times 10^{-3}}$	$\mathbf{1.4 \times 10^{-5}}$

*Note: The reported time is in seconds.

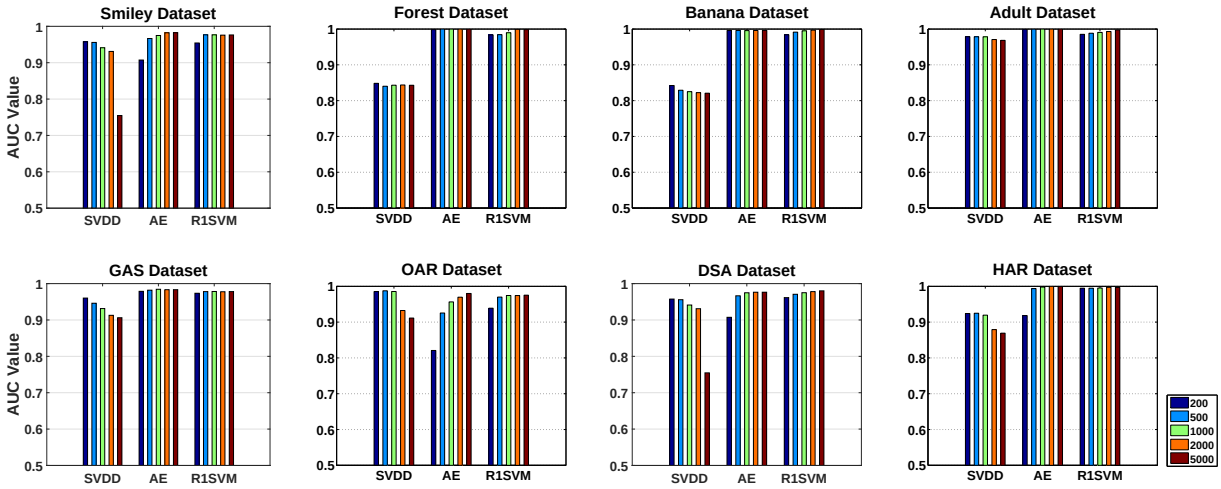


Figure 2: Comparison of accuracy of anomaly detection methods as the number of training records is varied.

domly moved. The other is the "Banana" dataset that is a mixture of two banana shaped distributions, which are randomly moved in 100 dimensions. All the records in each dataset are normalised between [0,1].

Although R1SVM is designed to overcome the challenges that arise for anomaly detection in large datasets, we conducted our experiments on datasets with varying numbers of dimensions and records, from 200 to 40,000 records, to assess the effect of data size on its performance. In each experiment, 80% of records are randomly selected for training and 20% for testing, and then, respectively, mixed with 5% and 20% anomalous records, randomly drawn from $U(0, 1)$. Note that training is performed in an unsupervised way, and labels are only used for testing.

Accuracy Metric: We use the Receiver Operating Characteristic (ROC) curve and the corresponding Area Under the Curve (AUC) to measure the performance of all the methods. The reported training/testing times are in seconds based on experiments run on a machine with an Intel Core i7 CPU at 3.40 GHz and 8 GB RAM. The stated AUC values and training/testing times are the average of 1000 iterations for each experiment.

Experiment 1: Interpretation of Basis for Anomaly Detection with R1SVM

Fig. 1 demonstrates the benefits of using random nonlinear features for anomaly detection on two datasets, Banana and HAR (due to the shortage of space images of the other datasets are not shown). The interpretation of the subfigures, from the top is as follows: iVAT image of the raw datasets, the iVAT image of the projected datasets, and the scatter plot of the projected datasets (projected to $\mathbb{R}^3$). As can be seen in the image of the raw datasets, normal records (first 1000 records) appear in dense blocks, while the anomalous records (last 50 records) are shown in gray shadow (since they are distributed across the dataset). When the data are projected to a lower space, a clearer separation appears between the normal records and anomalies, as reflected in the improved clarity and contrast in the block structure. This is also reflected in the corresponding scatter plots, which show the effect of projection on normal records (shown in blue) and anomalies (shown in black). We postulate that the explanation of this effect is the concentration of the data around its mean as a result of the random projection (Dasgupta 2000; Zimek, Schubert, and Kriegel 2012)

Table 2: Comparing computational and memory complexity of SVDD, AE and R1SVM

Technique	SVDD	AE	R1SVM
Training	$O(dn^2)$	$O(dmn)$	$O(kn)$
Testing	$O(dn)$	$O(dmn)$	$O(k+d)$
Memory	$O(dn^2)$	$O(dq)$	$O(kn)$

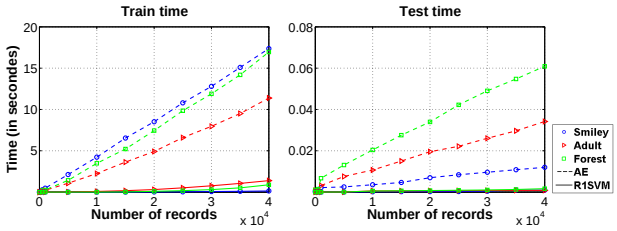


Figure 3: Comparison of the training and testing time of AE and R1SVM on large datasets.

Experiment 2: Empirical Performance of R1SVM

Table 1 compares the AUC results, training and testing time of R1SVM with SVDD and AE for several medium size (2000 records) datasets — a limit of 2000 records was chosen because the performance of SVDD significantly degrades on larger datasets. As shown in the table, our proposed approach delivers a comparable AUC to the state-of-the-art AE. However, the AUC of SVDD is significantly lower — e.g., up to 14% for DSA and 13% for Smiley datasets. A more significant advantage of R1SVM is its reduction in training/testing time. R1SVM reduces theses measures by factor of approximately 100 and 1000 times compared to AE and SVDD, respectively.

When selecting an anomaly detection technique, the size of the training dataset is an immediate concern. Some techniques, such as kernel machines, can perform best with small datasets, i.e., their performance decays as the number of records grows. In contrast, methods like AE can be inaccurate if trained with small numbers of records (Bengio and LeCun 2007). Additionally, the training time for some techniques is prohibitive for large numbers of records, e.g., the time complexity of kernel-based methods can grow at least quadratically in the number of data records. Fig. 2 and Fig. 3 show how the AUC and training/testing time vary with the number of training records.

Fig. 2 shows the impact of larger-scale training and high-dimensional data on the accuracy of the studied approaches. We observe that the accuracy of SVDD decreases as the number of training records increases. In some cases, e.g., Smiley, OAR and DSA datasets, SVDD experiences a substantial decrease in AUC when the number of training records reaches 5000. In contrast, the accuracy of AE can initially be low but increases as the data size grows. Overall, only R1SVM delivers more consistent results across various ranges of data sizes.

An attractive property of autoencoders is their efficiency in training/testing time, which scales linearly with the number of records. Therefore, we compared the training and testing time of AE and R1SVM for large datasets. The results in Fig. 3 suggest that although both measures grow linearly for the two techniques, the training/testing time of AE grow at a much faster rate. The AUC values for this experiment are not included, since they were more or less consistant. Table 2 summarises the computational and memory complexity of these techniques, where m and q are the size of the bottleneck layer and batch in the AE.

In summary, the above experiments demonstrate that R1SVM is as accurate as a deep autoencoder, but approximately 100 times faster in terms of training/testing time. It is also important to note that the training algorithm of R1SVM requires only one parameter, k, to be set. Since we are interested in anomaly detection, modelling the underlying distribution is more important than preserving distances among classes, therefore the data can be projected to very low dimensions. In our empirical analysis we have also tested the sensitivity of R1SVM to the choice of the parameter k. Overall, the value of k has little effect on accuracy, while larger values of the k increase the training time. Hence, smaller values of k are most effective.

Conclusion and Future Work

We presented R1SVM, an unsupervised anomaly detection technique that approximates a nonlinear 1SVM through applying the original linear 1SVM method on a randomised nonlinear projection of the data. Using a simple but effective random projection overcomes the scalability issues of 1SVM methods while enhancing the accuracy of anomaly detection. Our empirical analysis on several benchmark datasets shows that R1SVM not only delivers significant improvements over a conventional nonlinear 1SVM, but it matches the performance of a state-of-the-art deep autoencoder — while reducing its training and testing time by up to two orders of magnitude. These savings in time and space enable R1SVM to execute anomaly detection on large datasets more efficiently, in real-time applications or memory constrained devices, such as smart phones and wireless sensor networks. In addition to large-scale training, the other major advantage of R1SVM is that it is also shown to be effective at maintaining its accuracy on small datasets when training data is limited.

In future work, we will also explore the changes in the eigen-spectrum of the kernel matrix and generalisation error bound when using the Nyström method. Unlike random Fourier features, the basis functions in the Nyström method are data dependent and randomly sampled from the training data.

Acknowledgment

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

References

Achlioptas, D.; McSherry, F.; and Schölkopf, B. 2002. Sampling techniques for kernel methods. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, volume 1, 335.

Bengio, Y., and LeCun, Y. 2007. Scaling learning algorithms towards AI. *Large-Scale Kernel Machines* 34.

Blum, A. 2006. Random projection, margins, kernels, and feature-selection. In *Subspace, Latent Structure and Feature Selection*. 52–68.

Bottou, L., and Lin, C.-J. 2007. Support vector machine solvers. *Large scale kernel machines* 301–320.

Cao, L. J.; Chua, K. S.; Chong, W. K.; Lee, H. P.; and Gu, Q. M. 2003. A comparison of PCA, KPCA and ICA for dimensionality reduction in support vector machine. *Neurocomputing* 55(1):321–336.

Dasgupta, S. 2000. Experiments with random projection. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI)*, 143–151.

Fung, G., and Mangasarian, O. L. 2001. Proximal support vector machine classifiers. In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 77–86.

Fung, G., and Mangasarian, O. L. 2003. Finite Newton method for Lagrangian Support Vector Machine classification. *Neurocomputing* 55(1):39–55.

Hamid, R.; Xiao, Y.; Gittens, A.; and DeCoste, D. 2014. Compact random feature maps. In *Proceedings of 31st International Conference on Machine Learning (ICML)*.

Joachims, T. 1999. Making large scale SVM learning practical. *Advances in Kernel Methods - Support Vector Learning*.

Kar, P., and Karnick, H. 2012. Random feature maps for dot product kernels. In *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 583–591.

Lee, Y.-J., and Mangasarian, O. L. 2001. RSVM: Reduced Support Vector Machines. In *Proceedings of the SIAM International Conference on Data Mining (SDM)*, volume 1, 325–361.

Lopez-Paz, D.; Sra, S.; Smola, A.; Ghahramani, Z.; and Schölkopf, B. 2014. Randomized nonlinear component analysis. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*.

Manevitz, L. M., and Yousef, M. 2002. One-class SVMs for document classification. *Journal of Machine Learning Research (JMLR)* 2:139–154.

Platt, J. C. 1999. Fast training of support vector machines using sequential minimal optimization. In *Advances in Kernel Methods*, 185–208.

Rahimi, A., and Recht, B. 2007. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems (NIPS)*, 1177–1184.

Rahimi, A., and Recht, B. 2009. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In *Advances in Neural Information Processing Systems (NIPS)*, 1313–1320.

Rajasegarar, S.; Leckie, C.; Bezdek, J. C.; and Palaniswami, M. 2010. Centered hyperspherical and hyperellipsoidal one-class support vector machines for anomaly detection in sensor networks. *IEEE Transactions on Information Forensics and Security (IEEE IFS)* 5(3):518–533.

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

Settles, B. 2010. Active learning literature survey. *University of Wisconsin, Madison* 52:55–66.

Sonnenburg, S.; Rätsch, G.; Schäfer, C.; and Schölkopf, B. 2006. Large scale multiple kernel learning. *Journal of Machine Learning Research (JMLR)* 7:1531–1565.

Subasi, A., and Ismail Gursoy, M. 2010. EEG signal classification using PCA, ICA, LDA and support vector machines. *Expert Systems with Applications* 37(12):8659–8666.

Sun, S.-Y.; Tseng, C.-L.; Chen, Y.; Chuang, S.; and Fu, H. 2004. Cluster-based support vector machines in text-independent speaker identification. In *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, volume 1.

Tax, D. M., and Duin, R. P. 2004. Support Vector Data Description. *Machine Learning* 54:45–66.

Tax, D. 2013. DDtools, the Data Description toolbox for Matlab. Version 2.0.2.

Tong, S., and Koller, D. 2002. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research (JMLR)* 2:45–66.

Vapnik, V. N. 1998. Statistical learning theory.

Vempati, S.; Vedaldi, A.; Zisserman, A.; and Jawahar, C. 2010. Generalized RBF feature maps for Efficient Detection. In *Proceedings of the 21st British Machine Vision Conference*, 1–11.

Vishwanathan, S.; Smola, A. J.; and Murty, M. N. 2003. Simplesvm. In *Proceedings of the International Conference on Machine Learning (ICML)*, 760–767.

Wang, L.; Nguyen, U. T.; Bezdek, J. C.; Leckie, C.; and Ramamohanarao, K. 2010. iVAT and aVAT: enhanced visual analysis for cluster tendency assessment. In *Advances in Knowledge Discovery and Data Mining*. 16–27.

Wang, D.; Yeung, D. S.; and Tsang, E. C. 2006. Structured one-class classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics,* 36(6):1283–1295.

Yang, C.; Duraiswami, R.; and Davis, L. S. 2005. Efficient kernel machines using the improved fast Gauss transform. In *Advances in Neural Information Processing Systems*, 1561–1568.

Zimek, A.; Schubert, E.; and Kriegel, H.-P. 2012. A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining* 5(5):363–387.