Artif Intell Rev (2007) 27:273–294
DOI 10.1007/s10462-008-9082-5

An evaluation of dimension reduction techniques for one-class classification

Santiago D. Villalba · Pádraig Cunningham

Published online: 21 September 2008

Abstract Dimension reduction (DR) is important in the processing of data in domains such as multimedia or bioinformatics because such data can be of very high dimension. Dimension reduction in a supervised learning context is a well posed problem in that there is a clear objective of discovering a reduced representation of the data where the classes are well separated. By contrast DR in an unsupervised context is ill posed in that the overall objective is less clear. Nevertheless successful unsupervised DR techniques such as principal component analysis (PCA) exist—PCA has the pragmatic objective of transforming the data into a reduced number of dimensions that still captures most of the variation in the data. While one-class classification falls somewhere between the supervised and unsupervised learning categories, supervised DR techniques appear not to be applicable at all for one-class classification because of the absence of a second class label in the training data. In this paper we evaluate the use of a number of up-to-date unsupervised DR techniques for one-class classification and we show that techniques based on *cluster coherence* and *locality preservation* are effective.

Keywords One class classification · Dimensionality reduction · Feature selection · Feature transformation · Principal component analysis · Locality preservation · Cluster coherence

1 Introduction

In recent years, the traditional distinction in machine learning between supervised and unsupervised techniques has been blurred due to the emergence of real-world problems that sit

S. D. Villalba (✉) · P. Cunningham
Machine Learning Group, School of Computer Science and Informatics, University College Dublin, Dublin, Ireland
e-mail: Santiago.Villalba@ucd.ie

P. Cunningham
e-mail: Padraig.Cunningham@ucd.ie

somewhere between these two extremes. In supervised classification problems, *discriminating* classifiers are trained using positive and negative examples. However, for a number of practical problems, counter-examples are either rare, entirely unavailable or statistically unrepresentative. Such problems include industrial process control, text classification and analysis of chemical spectra.

One-class classifiers (OCCs) have emerged as a set of techniques for situations where labelled data exists for only one of the classes in a classification problem. For instance, in industrial inspection tasks, abundant data may only exist describing the process operating correctly. It is difficult to gather training data describing the myriad of ways the system might operate incorrectly. A related problem is where negative examples exist, but their distribution cannot be characterised. For example, it is reasonable to provide characteristic examples of family pictures but impossible to provide examples of pictures that are "typical" of non-family pictures. One-class classifiers are emerging as a solution, which *characterises* the target class, to distinguish it from all other classes.

In practice, one-class problems are typically of high dimension so DR is an important pre-processing step. Indeed the evaluation presented by Manevitz and Yousef (2001) shows that one-class Support Vector Machine (SVM) performance is quite sensitive to the number of features used. This contrasts with two-class SVMs which are generally considered to be robust to high data dimensionality. This provides additional justification for DR in OCC construction. However, the absence of counter-examples means it is difficult to identify a feature subset that encodes a *discriminating* description of the concept.

In this paper we review a range of unsupervised DR techniques and evaluate their performance on a number of OCC problems. We find that DR based on *locality preservation* and *cluster coherence* principles seem particularly promising for OCC. However locality preservation is more likely to be effective when there are no irrelevant features in the full feature set; i.e. locality in the original space must be meaningful. One remarkable finding from the evaluation is the bad performance of PCA on many of the datasets. It appears that for some domains PCA will be at least as effective for dimension reduction as the more complex alternative techniques, but for others it will damage the performance. That there is "no silver bullet" is not surprising as the requirements of dimension reduction for the different datasets vary and the different biases of the different approaches are appropriate in some situations but not others.

In the next section we provide an overview of OCC and describe the OCC techniques included in the evaluation. In Sect. 3 we describe the DR techniques considered in the evaluation—the evaluation is presented in Sect. 4. The paper concludes with a summary and some proposals for further research.

2 One-class classifiers

Traditionally machine learning tasks are divided into supervised and unsupervised categories. Roughly speaking, in unsupervised learning we are provided with a dataset (set of examples describing a real world concept) and the objective is to uncover some structure in the data. In supervised learning we are provided with a dataset where the information to be modeled is explicitly stated in the form of a label (a "class" label in the case of so called "classification problems") and the task is to predict the label for new (as yet unseen) examples.

One-class classification, also referred to as novelty or outlier detection, is sometimes thought of as a weaker form of unsupervised learning. The task is still to classify, but the only information we are given about the training examples is that they belong to the same

class, typically called the "positive" or "target" class. The task here is to accept or reject unseen examples depending on their similarity to the known positive examples. OCC approaches consequently can operate with very few, or no, negative training examples. In other words, one-class learning handles the "no-counter-example" and "imbalanced-data" problems by considering only positive examples. When unlabeled examples and/or a number of negative examples are available for training, several OCC techniques can also use them to fine-tune their performance.

Despite the lack of formal foundations for the one-class problem (Yaniv and Nisenson 2006), there is a doubtless increasing interest in both the methods and techniques available and their practical applicability. Several recent surveys (Tax 2001; Marsland 2003; Markou and Singh 2003a,b; Hodge and Austin 2004; Juszczak 2006) cover to a greater or lesser extent the various alternatives. In the current study we have chosen four different OCCs: support vector data description (SVDD), a k-nearest neighbours approach, a k-means clustering approach and a Gaussian model. All of these are available in the Data Description toolbox (Tax 2007), an open source Matlab library of one-class classification tools.

Support vector data description (Tax 2001; Tax and Duin 2004): The SVDD learns the hypersphere, defined by a center a and a radius R, that encloses (almost) all the training set while covering as little volume as possible. It employs the kernel trick (Schölkopf and Smola 2001, p. 34) for learning more flexible boundaries, and the solution is found by solving a convex quadratic optimization problem analogous to the one found in SVMs.

Clustering (*k-means*): Another approach to one-class classification is that of learning clusters, modeling the target class as a reduced set of cluster prototypes or centers onto which new examples are projected. Examples of clustering methods that can be used are the self organizing map, learning vector quantization or k-means (the one we choose here, Tax 2001; Juszczak 2006). When a new example is to be classified, its distance to the nearest prototype is used to score the extent to which it is an outlier.

Lazy learning (*k-nearest-neighbours*): The nearest neighbour approach can be used for constructing one-class classifiers. The training data is stored and an *outlierness* criterion is calculated for new examples based on their nearest neighbours, i.e. their position relative to the seen examples. Several criteria have been proposed to measure the outlierness of an example (Harmeling et al. 2006; Rieck and Laskov 2007). Here we use γ (Harmeling et al. 2006) which is the average of the distances to the k nearest neighbours.

Parametric density estimation (*Gaussian model*) (Tax 2001; Juszczak 2006): The Gaussian model is a simple parametric one-class classifier which models the training data under the assumption that it comes from a unimodal multivariate normal distribution. These assumptions fit a lot of natural processes, but when they are violated this model introduces a large bias. The mean and the covariance matrix are estimated using a maximum likelihood approach. To avoid possible numerical stability problems associated with the computation of the determinant of the covariance matrix, more frequent in the high dimensional spaces we work in, no normalization factor is calculated and just a plain Mahalanobis distance is used as the resemblance criterion.

We selected these OCC strategies and not others because of their conceptual simplicity and their well established properties. One characteristic that all the classifiers under consideration share is some notion of locality and compactness. Locality is a relative concept that can be defined in several ways, for example by only taking into account the interactions with a fixed number of neighbours (nearest neighbours in k-NN and the nearest prototype in k-means) or those in a small region of the input space (Gaussian kernels with a small width). Generally speaking we talk of local models, models that cover just a small volume centered at concrete points and dismissing any further considerations, as opposed to global ones, those that span

the whole space. As we will note in the next section, this notions of locality are seriously influenced by high dimensions.

3 Dimension reduction techniques

Learning gets harder very quickly as the dimensionality increases, not only because of the possible presence of noise and redundancy in the data but for other reasons as well. In high dimensional spaces the Gaussian kernel suffers from several theoretical problems (François et al. 2005; Verleysen and François 2005). The empty space phenomenon tell us that to cover the whole space we need a number of samples that grows exponentially with the dimensionality (fortunately, in practice, we won't need to cover the whole space). The curse of dimensionality implies that in order to learn successfully, we would need a number of training examples that also grows exponentially with the dimensionality. The "concentration of measure" phenomenon seems to render distance measures not relevant to whatever concept is to be learnt as the dimension of the data increases (Ledoux 2001). So, for theoretical and practical reasons, it is necessary to study methods for combating high dimensionally in the one-class setting.

Research on dimension reduction has itself two dimensions. The first design decision is whether to select a subset of the existing features or to transform to a new reduced set of features. The other dimension in which DR strategies differ is the question of whether the learning process is supervised or unsupervised (see Fig. 1). For OCC problems both feature selection and feature transformation strategies are relevant. However, given that labelled data is only available for one class, it seems that supervised DR techniques cannot be directly applied to OCC problems.

In supervised learning the objective of DR is to optimize the performance of the final system, that is, minimize the classification error. However, in one-class classification performance estimation is difficult because the absence of counterexamples makes the estimation of the false positive rate hard and assumption-based. This makes it difficult also to tune the bias of the classifier and the best strategy to address this problem depends on the specifics of the data available.

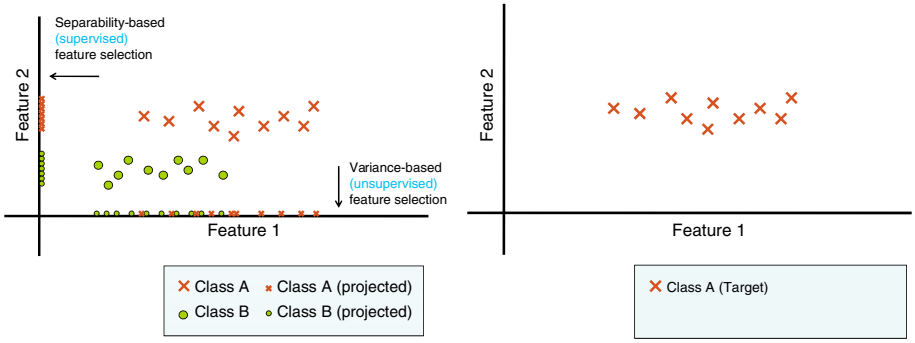

Fig. 1 The graph on the left shows the strength of supervised feature selection arising from the potential to discover features that separate classes. By contrast feature selection based on variance may render data *less* separable. Then in the one-class scenario on the right, no direct separability information is present in the training data

A sensible approach is to try to synchronize the assumptions of both DR and classification. In our evaluation we consider four DR techniques. The first two are the classical principal component analysis (PCA) and the Q-α algorithm presented by Wolf and Shashua (2005). The final two, locality preserving projections (LPP) and Laplacian score (LS) are explicitly based on the principle of *locality preservation* and these are described in Sect. 3.1. We believe that locality preservation is of particular relevance to DR in the OCC domain because, usually, one class classifiers rely on local neighbourhood relationships (see Sect. 2). The first three techniques are in the family of spectral methods, where the low dimensional representations are derived from the eigenvectors (spectra) of specially constructed matrices (Saul et al. 2006, p. 294), while LS is closely related to LPP.

Principal component analysis (*PCA*): PCA (Jolliffe 2002) is the most commonly used technique for unsupervised dimensionality reduction. It aims at finding the linear projections that best capture the variability of the data. In this study we use the common approach of keeping those directions that explains most of the variance. In Tax and Muller (2003) and in Sect. 4 it is shown that retaining these high variance dimensions is not always optimal for one-class classification, so a minor components analysis (use the smallest variance directions) can be better under some circumstances.

The Q-α algorithm: A well motivated criterion of cluster quality is cluster coherence, in graph theoretic terms this is expressed by the notion of objects within clusters being well connected and individual clusters being weakly linked. The whole area of spectral clustering captures these ideas in a well founded family of clustering algorithms based on the task of minimising the *graph-cut* between clusters (Ng et al. 2001).

The principles of spectral clustering have been extended by Wolf and Shashua (2005) to produce the Q-α algorithm that simultaneously performs feature subset selection and discovers a good partition of the data. As with spectral clustering, the fundamental data structure is the affinity matrix $\mathbf{A}$ where each entry $\mathbf{A}_{ij}$ captures the similarity (typically as a dot-product) between data points i and j. In order to facilitate feature selection the affinity matrix for Q-α is expressed as $\mathbf{A}_\alpha = \sum_{i=1}^{p} \alpha_i \mathbf{m}_i \mathbf{m}_i^\mathsf{T}$ where $\mathbf{m}_i$ is the ith feature vector in the data matrix that has been normalised to be centered on 0 and be of unit L_2 norm (this is the set of values in the data set for feature i). $\mathbf{m}_i \mathbf{m}_i^\mathsf{T}$ is the *outer*-product of $\mathbf{m}_i$ with itself. α is the weight vector for the p features—ultimately the objective is for some of these weight terms to be set to 0.

In spectral clustering $\mathbf{Q}$ is an $n \times k$ matrix composed of the k eigenvectors of $\mathbf{A}$ corresponding to the largest k eigenvalues (λ_i). Wolf and Shashua show that the relevance of a feature subset as defined by the weight vector α can be quantified by:

$$Rel(\alpha) = trace(\mathbf{Q}^\mathsf{T} \mathbf{A}_\alpha^\mathsf{T} \mathbf{A}_\alpha \mathbf{Q}) = \sum_{i=1}^{k} \lambda_i$$

They show that feature selection and clustering can be performed as a single process by optimising:

$$\max_{\mathbf{Q}\alpha} trace(\mathbf{Q}^\mathsf{T} \mathbf{A}_\alpha^\mathsf{T} \mathbf{A}_\alpha \mathbf{Q})$$

subject to $\alpha^\mathsf{T}\alpha = 1$ and $\mathbf{Q}^\mathsf{T}\mathbf{Q} = \mathbf{I}$.

Wolf and Shashua show that this optimization problem can be approached by solving two inter-linked eigenvalue problems that produce solutions for α and $\mathbf{Q}$. They demonstrate that a process of iteratively solving for α then fixing α and solving for $\mathbf{Q}$ will converge to a local maximum of the energy function (induced cluster coherence). They also show that the

process has the convenient property that the α_i weights are biased to be positive and sparse, i.e. many of them will be zero.

So the Q-α algorithm performs feature selection in the spirit of spectral clustering, i.e. it discovers a feature subset that will support a partitioning of the data where clusters are well separated according to a graph-cut criterion.

3.1 Locality preservation

Locality preservation in dimensionality reduction techniques refers to the aim of keeping neighbourhood properties only, e.g. objects that are close in the input space should also be close in the reduced space. Several linear and nonlinear techniques exploiting this criterion have recently been proposed (Saul et al. 2006). For the OCC problem it is rational to think that a locality preserving dimension reduction technique would be more practical in some cases than a global based one. Locality and density are frequently used in the OCC literature and both are present in the locality preservation bias.

Locality preserving projections (LPP): The idea behind LPP is that of finding subspaces which preserve the *local structure* in the data (He and Niyogi 2003; He 2005). Given a matrix $\mathbf{A}$ (symmetric, positive, invertible and usually sparse) which captures information about the relationships between the data points, for example the similarity in a neighbourhood, LPP finds the optimal linear embedding that respects the structure present in that matrix. LPP preserves cluster structures when clustering is based on locality, such as in the k-means algorithm, which is an attractive quality when used together with cluster analysis based OCCs. The details of LPP are described in Algorithm 1.

Algorithm 1: LPP computation (He and Niyogi 2003)

Construct the adjacency graph: let $\boldsymbol{S}$ be the training set and G denote a graph with $|\boldsymbol{S}|$ nodes. We put an edge between nodes i and j if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are "close". There are two variations:

- ε-neighbourhoods (parameter $\varepsilon \in \mathbb{R}$). Nodes i and j are connected if $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 < \varepsilon$ where the norm is the usual Euclidean norm in $\mathbb{R}^{d_X}$.
- k nearest neighbours (parameter $k \in \mathbb{N}$). Nodes i and j are connected if i is among the k-nearest neighbours of j or vice-versa.

Choose the weights for the graph edges: Here, as well, we have two variations for weighting the edges. $\mathbf{A}$ is a sparse symmetric $|\boldsymbol{S}| \times |\boldsymbol{S}|$ matrix with $\mathbf{A}_{ij}$ having the weights of the edge joining vertices i and j, and 0 if there is no such edge.

- Heat kernel (parameter $t \in \mathbb{R}$). When nodes i and j are connected put $\mathbf{A}_{ij} = e^{\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{t}}$
- Simple minded (no parameter). When nodes i and j are connected, put $\mathbf{A}_{ij} = 1$.

Eigenmaps: Compute the eigenvectors and eigenvalues for the generalized eigenvector problem:

$$\boldsymbol{S}\mathbf{L}\boldsymbol{S}^{\top}\boldsymbol{a} = \lambda \boldsymbol{S}\mathbf{D}\boldsymbol{S}^{\top}\boldsymbol{a} \tag{1}$$

where $\mathbf{D}$ is a diagonal matrix whose entries are column sums of $\mathbf{A}$, $\mathbf{D}_{ii} = \sum_j \mathbf{A}_{ji}$. $\mathbf{L} = \mathbf{D} - \mathbf{A}$ is the Laplacian matrix.

The embedding is defined by the bottom eigenvectors in the solution of Eq. 1. The construction of the weighted graph in the first and second steps of Algorithm 1 can be accomplished using a variety of criteria. This aspect is quite useful as it provides a mechanism to bring external information to bear on the problem.

Laplacian score for feature selection (*LS*): The same criterion of locality preservation found in LPP can be applied in the feature selection context, where the merit of each feature is measured according to its locality preservation power (He et al. 2005).

As with Q-α, there is no explicit enumeration of the feature subsets. Rather a nearest neighbour based graph is constructed from the training set and analysed to rank each feature individually, without taking into account further interactions between them. The first two steps of the algorithm are identical to those of LPP (Algorithm 1). For ranking each feature, its *Laplacian score* is computed. For the i-th feature we define:

$$\widetilde{\mathbf{m}}_i = \mathbf{m}_i - \frac{\mathbf{m}_i^\top \mathbf{D1}}{\mathbf{1}^\top \mathbf{D1}}\mathbf{1}$$

where $\mathbf{1} = [1, \ldots, 1]^\top$

The Laplacian score (LS_i) for the i-th feature is:

$$LS_i = \frac{\widetilde{\mathbf{m}}_i^\top \mathbf{L}\widetilde{\mathbf{m}}_i}{\widetilde{\mathbf{m}}_i^\top \mathbf{D}\widetilde{\mathbf{m}}_i} \tag{2}$$

In Eq. 2 the numerator measures to which extent the ith feature preserves the structure present in the graph—with smaller values corresponding to better features. The denominator measures the variance of the feature. For a feature to be selected it must have low LS, which implies high variance (more representative power) and locality, as defined by the graph, preservation properties.

4 Evaluation

In order to evaluate the performance of the different DR/OCC pairs we used the labeled datasets summarized in Table 1. They are high dimensional, low sample size datasets that come from different domains: biomedical (Bronchiolitis, Arrhythmia, TIS-5% and Leukemia), chemical spectral analysis (85Drugs), fault detection (DelftPump5x3Noisy), text classification (fbis, tr12, oh5 and re0), image classification (indoor-outdoor-flowers-leaves), digit recognition (mfeat-pixel) and mine detection (sonar). We preprocessed them with normalisation and, when applicable, missing value substitution. We also subsampled the TIS dataset so that the experiments could be run in reasonable time.

4.1 Baseline evaluation

In addition to the techniques described in Sect. 3 and the scenario of no feature selection, we add two dimension reduction methods for the purpose of comparison. We evaluate a random feature selection method to provide a baseline—the algorithms under consideration are not effective if they cannot improve on this. We also consider a ranking of the features using information gain over the original labeled datasets (this is "cheating"). In the same way that no dimension reduction and random feature selection provide a *floor* for the evaluation, this cheating feature selection provides a *ceiling* for expected performance.

It is interesting to note that these ceiling and floor proposals are breached in some circumstances (see Tables 2 and 3). Thinking that information gain will provide an optimal solution for all problems is naive. Get a more accurate ceiling is a supervised dimensionality reduction problem that goes beyond the aims of this paper. Analogously, the random approach could do as well as, or better, than more principled approaches; but that case is improbable so high

Table 1 Summary of the datasets used in the evaluation

Dataset	n	d	Target Class (n#)	Source
Bronchiolitis	118	22	1-Day (37)	Doyle et al. (2006) [a]
Arrhythmia	452	279	Normal (245)	Asuncion and Newman (2007)[b]
TIS-5%	668	927	TIS (178)	Liu and Wong (2003)[c]
Leukemia	72	7,129	ALL (47)	Golub et al. (1999)[d]
85Drugs	85	510	Cocaine (49)	Ryder (2002)
DelftPump5x3Noisy	899	64	Normal (216)	Ypma (2001)[e]
fbis	2,463	2,000	100 (92)	Han and Karypis (2000)[f]
oh5	918	3,012	Anticoag (74)	Han and Karypis (2000)[f]
re0	1,504	2,886	bop (92)	Han and Karypis (2000)[f]
tr12	313	5,804	100 (93)	Han and Karypis (2000)[f]
iofl	240	285	Ind (59), Lea (60)	–[a]
mfeat-pixel	2,000	240	7 (200), 8 (200)	Asuncion and Newman (2007)[b]
sonar	208	60	Mines (111)	Asuncion and Newman (2007)[b]

[a] http://mlg.ucd.ie/datasets
[b] http://archive.ics.uci.edu/beta/
[c] http://research.i2r.a-star.edu.sg/rp/SequenceData/TIS.html
[d] http://www.upo.es/eps/bigs/datasets.html
[e] http://www-ict.ewi.tudelft.nl/%7Edavidt/occ/547/oc_547.html
[f] http://prdownloads.sourceforge.net/weka/19MclassTextWc.zip?download

chances are that it will perform bad. Averaging over several runs would account for those improbable cases, but we are not really interested on how random feature selection performs at this stage. Keeping to just one the number of random feature selection runs retains the "real flavor" of randomness.

4.2 Parameter selection

One difficulty in assessing the performance of the combination of OCC and DR techniques is the model selection (parameter tuning) required for the different techniques. We followed a simple approach of fixing the values of the parameters to reasonable values; for instance, $k = 6$ is set as the number of clusters for k-means and neighbours for k-NN. With SVDD we use a Gaussian kernel and the rest of the parameters are left to the dd_tools default values. The classification threshold is set in the training process so that 90% of training examples are accepted (i.e. we consider that 10% of the training examples are negatives).

For the parameters of the dimension reduction techniques, when applicable, we follow the principle of selecting those used in the counterpart classifiers. For example, the same value of k in the k-nearest neighbour is used when constructing the adjacency graph for LPP and LS, and the value of k in the k-means algorithm is set to the target number of clusters for Q-α. In both LPP and LS the "simple minded" weighting approach is followed. No further model selection is done and we also fix the rest of the parameters to a priori selected default values. The choice of target dimensionality is also an important issue. In this case we just explored all possibilities, from dimensionality 1 to 60, and higher dimensionalities at regular intervals up to the original dimensionality or to the maximum defined by the feature transformation embedding.

Selecting generic parameter settings is useful to the purpose of evaluation, but ignores that there may be dependencies between the DR techniques and the OCC methods. The same parameters cannot be optimal in all the diverse problems we use as test bed. Unfortunately

Table 2 The results on text classification (a–d), image classification (e–f) and digit recognition (g–h) datasets

	No DR	IG	Random	Q-α	LS	LPP	PCA
(a) fbis, 100							
Gauss	50.0(2000)	*76.7(19)*	**61.3(50)***	53.8(2)	57.3(26)	51.3(8)	50.6(6)
k-Means	55.8(2000)	*74.1(14)*	**56.9(1400)**	54.9(1800)	56.2(1800)	51.3(8)	51.6(6)
k-NN	54.2(2000)	*73.6(14)*	56.0(400)	54.0(2)	**56.2(6)**	51.6(8)	52.1(3)
SVDD	**58.5(2000)**	*73.4(14)*	57.4(1800)	58.3(9)	57.9(1800)	50.0(26)	49.7(2)
(b) oh5, Anticoagulants							
Gauss	50.0(3012)	*85.3(24)*	**59.0(45)**	50.7(903)	50.0(2409)	50.0(9)	50.0(25)
k-Means	50.1(3012)	*81.2(31)*	**56.3(19)**	53.7(2409)	49.4(1204)	50.0(9)	51.0(1)
k-NN	48.1(3012)	*78.6(18)*	55.2(15)	**56.3(602)**	49.6(1506)	50.0(9)	50.0(25)
SVDD	52.9(3012)	*89.0(33)*	55.0(13)	**63.0(3)***	51.0(2710)	50.0(9)	50.0(25)
(c) re0, bop							
Gauss	50.0(2886)	*72.1(11)*	**60.9(38)**	52.8(57)	51.5(60)	45.9(1)	50.0(24)
k-Means	52.3(2886)	*64.7(11)*	**57.3(35)**	53.6(2308)	52.0(2020)	47.5(18)	54.7(1)
k-NN	50.2(2886)	*56.6(5)*	**55.9(31)**	52.1(2308)	50.8(1443)	48.8(19)	50.8(1)
SVDD	56.2(2886)	*80.4(32)*	59.1(865)	**75.1(7)***	54.7(2597)	47.4(16)	52.7(2)
(d) tr12, 100							
Gauss	50.0(5804)	*79.2(38)*	**55.6(57)**	52.6(10)	52.7(580)	50.2(8)	48.0(3)
k-Means	48.2(5804)	*71.4(6)*	52.9(44)	51.3(3)	51.8(26)	**56.4(1)***	49.2(9)
k-NN	48.8(5804)	*68.6(7)*	51.1(47)	**51.5(5)**	51.3(21)	50.5(5)	48.2(3)
SVDD	51.6(5804)	*86.1(1)*	**52.4(5223)**	49.7(4)	51.9(34)	50.5(5)	48.5(9)
(e) iofl, Indoor							
Gauss	50.0(285)	*78.3(114)*	78.0(29)	76.1(21)	**86.3(39)***	76.3(19)	78.3(39)
k-Means	73.5(285)	*78.3(228)*	75.0(57)	76.2(199)	74.3(228)	76.3(22)	**76.7(42)**
k-NN	69.5(285)	*68.9(256)*	69.8(171)	70.3(199)	69.2(256)	**74.1(22)**	68.2(51)
SVDD	71.9(285)	*89.6(199)*	69.7(60)	70.2(256)	70.8(256)	**72.0(13)**	70.6(29)
(f) iofl, Leaves							
Gauss	50.0(285)	*70.0(1)*	71.1(12)	71.1(55)	**73.1(51)**	70.8(9)	70.3(18)
k-Means	69.7(285)	*73.9(171)*	**72.5(85)**	69.4(256)	71.9(256)	70.8(16)	65.8(51)
k-NN	70.3(285)	*72.5(228)*	**72.5(57)**	70.3(256)	68.9(256)	71.1(13)	65.3(52)
SVDD	69.2(285)	*71.9(142)*	71.1(85)	69.2(256)	68.1(256)	**74.2(8)***	65.3(53)
(g) mfeat-pixel, 7							
Gauss	50.5(240)	*89.6(38)*	91.4(16)	91.6(15)	85.7(21)	80.0(5)	**97.6(37)**
k-Means	93.0(240)	*93.9(120)*	94.1(40)	94.2(216)	93.8(192)	79.3(2)	**96.5(60)**
k-NN	95.3(240)	*95.0(192)*	95.1(72)	95.5(168)	95.3(192)	62.4(2)	**97.8(72)***
SVDD	50.0(240)	*87.6(19)*	90.0(11)	**91.5(22)**	82.2(12)	75.3(5)	85.1(2)
(h) mfeat-pixel, 8							
Gauss	50.0(240)	*78.5(39)*	85.0(31)	84.4(31)	79.0(21)	77.4(9)	**94.0(57)***
k-Means	88.9(240)	*90.0(192)*	91.2(168)	91.8(216)	91.1(168)	71.0(11)	**91.9(120)**
k-NN	92.6(240)	*92.3(216)*	93.0(96)	93.2(192)	93.1(168)	53.3(5)	**93.3(144)**
SVDD	50.0(240)	*82.1(9)*	74.3(6)	**81.1(13)**	75.0(12)	74.6(11)	61.1(2)

For each OCC/DR pair we report the balanced accuracy rate for the winning dimensionality (in brackets). The results for the supervised dimension reduction based on information gain (IG) are shown in italics, as this is cheating. The winning unsupervised dimension reduction techniques for each classifier are shown in boldface. The overall winner, resolving ties by giving the victory to the lowest dimensionality, is highlighted with an asterisk*. It is clear that dimension reduction is effective in most cases, even if done at random

there is no clear criterion (loss function) on which to base the model selection. There exist techniques to optimize the parameters of one class classifiers [e.g. simplicity-versus-consistency (Tax and Muller 2004), minimum-volume (Tax 2001; Juszczak 2007)]. Moreover sometimes the model selection can come without a considerable extra effort (e.g. optimizing k for k-NN using leave-one-out density estimation or selecting the target dimensionality by using some form of the spectral gap). However, optimizing the parameters for both steps

Table 3 The results on biomedical (a–d) and other datasets (e–g)

	No DR	IG	Random	Q-α	LS	LPP	PCA
(a) Bronchiolitis, 1-Day							
Gauss	63.7(22)	*72.1(13)*	67.7(13)	72.6(7)	72.6(16)	**73.9(9)**	73.4(17)
k-Means	67.6(22)	*72.5(18)*	68.7(17)	74.0(16)	**75.5(9)***	66.8(11)	71.9(19)
k-NN	65.6(22)	*70.0(16)*	65.6(22)	65.7(21)	**69.0(11)**	68.6(11)	65.1(20)
SVDD	65.5(22)	*71.6(1)*	**68.4(16)**	67.0(16)	67.6(20)	60.3(10)	67.2(16)
(b) Arrhythmia, normal							
Gauss	55.0(279)	*77.6(57)*	69.4(111)	70.0(83)	68.9(83)	62.6(2)	**70.5(111)**
k-Means	67.8(279)	*76.3(34)*	69.1(33)	**71.0(167)***	69.9(139)	58.1(2)	68.4(167)
k-NN	67.2(279)	*76.9(35)*	67.3(167)	68.3(167)	**69.3(111)**	58.7(1)	66.1(195)
SVDD	66.1(279)	*75.2(31)*	67.1(139)	**70.5(111)**	68.1(167)	61.3(3)	68.0(167)
(c) TIS-5%, TIS							
Gauss	53.9(927)	*82.7(3)*	53.9(834)	**64.0(20)**	53.9(556)	50.0(4)	50.0(2)
k-Means	44.7(927)	*76.7(3)*	49.8(1)	50.2(24)	50.7(10)	**52.5(1)**	52.6(1)
k-NN	45.5(927)	*80.9(5)*	49.7(1)	**52.1(4)**	51.7(10)	50.6(2)	51.1(1)
SVDD	39.9(927)	*82.5(2)*	49.6(1)	**76.0(1)***	60.8(1)	50.1(3)	50.7(2)
(d) Leukemia, ALL							
Gauss	50.0(7129)	*93.6(4)*	**66.7(16)**	66.0(37)	66.0(10)	50.0(9)	52.9(1)
k-Means	63.2(7129)	*91.6(1)*	68.4(45)	63.2(2851)	**71.2(21)***	59.1(4)	50.9(9)
k-NN	50.7(7129)	*95.7(7)*	54.7(24)	53.6(2)	**62.7(19)**	50.0(14)	50.0(25)
SVDD	50.0(7129)	*93.7(17)*	68.1(712)	**70.5(712)**	61.9(6)	57.5(3)	66.3(6)
(e) 85Drugs, Cocaine							
Gauss	83.3(510)	*83.6(357)*	83.6(255)	81.9(459)	81.9(459)	**87.3(3)***	81.1(37)
k-Means	80.0(510)	*81.7(357)*	79.3(357)	**82.8(255)**	81.7(255)	80.9(17)	81.1(19)
k-NN	69.5(510)	*76.5(35)*	69.5(59)	66.8(459)	68.1(58)	**79.5(14)**	68.1(6)
SVDD	63.3(510)	*65.0(50)*	67.0(25)	65.4(1)	65.0(7)	**82.6(4)**	64.3(42)
(f) DelftPump5x3Noisy, Normal							
Gauss	65.1(64)	*74.2(32)*	65.0(60)	64.4(60)	**79.6(25)**	67.2(50)	55.1(60)
k-Means	53.5(64)	*64.9(4)*	55.4(49)	55.5(23)	59.0(17)	**63.8(48)**	51.8(2)
k-NN	54.5(64)	*70.2(8)*	55.3(39)	57.9(25)	**81.8(16)***	67.3(14)	57.3(5)
SVDD	52.8(64)	*57.6(9)*	54.3(31)	56.4(45)	55.2(18)	**64.2(45)**	50.5(59)
(g) Sonar, Mines							
Gauss	62.4(60)	*65.7(39)*	66.8(43)	69.0(21)	**70.6(37)***	64.0(24)	65.4(48)
k-Means	56.0(60)	*63.0(31)*	59.9(40)	**67.0(23)**	65.7(41)	66.0(33)	60.3(51)
k-NN	52.8(60)	*54.7(30)*	56.1(37)	61.5(19)	65.5(31)	**66.1(9)**	57.7(5)
SVDD	60.3(60)	*68.9(17)*	61.1(28)	62.2(46)	62.1(50)	**64.1(32)**	59.4(59)

For each OCC/DR pair we report the balanced accuracy rate for the winning dimensionality (in brackets). The results for the supervised dimension reduction based on information gain (IG) are shown in italics, as this is cheating. The winning unsupervised dimension reduction techniques for each classifier are shown in boldface. The overall winner, resolving ties by giving the victory to the lowest dimensionality, is highlighted with an asterisk*. We appreciate that locality preservation and cluster coherence can be very effective

would add too many dimensions to our already high-dimensional research problem. Since we are not looking for the best result, but just trying to gain insight into the interactions of the DR/OCC processes, we try to be fair by applying the same fixed conditions in all cases.

4.3 Results

The results are shown in Tables 2 and 3. The class distributions in most of the problems are unbalanced, so we use the balanced accuracy rate (BAR), estimated by stratified 10-fold cross validation, to measure the performance. The BAR is defined as the average of the true positive rate (sensitivity) and true negative rate (specificity). The figures shown

are those obtained by the winning target dimensionality in each case (in parenthesis); this selection a posteriori of the optimum target dimensionality could not be done in a realistic one-class problem, but here it allows us to assess the optimal performance that would be expected.

Obviously the distribution of datasets conditions the shape of the overall results. In just a few of them feature transformation is more suitable than feature selection. Usually feature selection is better for datasets containing informative features while it fails in datasets with highly correlated features (like mfeat), where on the contrary PCA excels. Moreover, we report results on four datasets that belong to the same type of problem: text classification using bag of words to represent the documents; we also used the same dataset changing the target class (mfeat-pixel and iofl). We try in this way to gain insight on the consistency of the good or bad behaviour of DR techniques over the same kind of problems and how important is the "distribution" of the classes we are modelling.

The excellent average performance of the supervised technique (IG) demonstrates that when fed with proper, discriminative, representations, one-class techniques work well. This can be appreciated, for example, in the case of the text classification problems. These also show the intricacies of dimensionality reduction; random feature selection is a good contender there and the structure found by the rest of the techniques is of little use. However the combination of Q-α and SVDD, a recurrent pair on high scores, provide reasonably good results on oh5 and, specially, re0; we will come back to this result later.

In the feature selection arena Q-α is very promising. In problems where feature selection is known to be essential, such as TIS (Liu and Wong 2003), it gives high scores to features proven relevant in the supervised classification setting, which also yields to improvements in the one-class case. Also the combined locality preservation and variance accentuation bias renders LS a very competitive contender; actually it is the technique that gets the overall victory more times (5) over a diverse range of datasets (iofl-Indoor, Bronchiolitis, Leukemia, DelftPump and sonar).

Usually mixing SVDD with LS and PCA is suboptimal. As we will see in Sect. 4.4, spreading the data is not a good idea when seeking for minimum volume, because in this way we force SVDD to create loose boundaries. This contrasts with what happens with Q-α; when modeling the class clusters correctly the selected features induce a small number of compact clusters that SVDD can describe more easily. See Tables 2 (b, c, g, h) and 3 (b, c, d).

In the case of several datasets like Arrhythmia or Leukemia, the locality preserving principle of LPP is not competitive with the rest of the unsupervised criteria. This is due to both numerical issues caused by the low sample size and to the presence of a lot of irrelevant features in the full feature set, which renders locality in that space inappropriate. However, in the case of the 85Drugs and the DelftPump datasets locality preservation is the clear winning bias. Data coming from spectra or multiple sensors usually have high redundancy and low irrelevancy (Verleysen and François 2005), which is exploited by the LPP and LS criteria to provide highly informative low dimensional representations.

In Table 4 we show a summary of the relative performance of the DR techniques across the different classifiers. In Appendix A further information can be found. In considering this summary it must be remembered that we are measuring across an arbitrary selection of domains and datasets; for example, the number of text datasets included has a strong influence on these results, favoring the random feature selection. Even if usually we will be only interested on the best performance for each problem, these figures do provide insight on which criteria work well together as well as on the consistency of the improvements achieved. For example, Q-α is the best match for k-means and with LS it is also a good partner of k-NN. This is not surprising since all those techniques pursue similar goals.

Table 4 Pairwise rankings for the dimensionality reduction techniques on a per classifier basis (a–d) and across all the classifiers (e)

	Dominance	Win	Loss
(a) Gauss			
IG	45	67	22
Random	28	57	29
LS	20	52	32
Q-α	14	51	37
PCA	−16	35	51
LPP	−27	29	56
No DR	−64	10	74
(b) k-Means			
IG	63	76	13
Q-α	19	54	35
LS	7	48	41
Random	4	47	43
PCA	−8	41	49
LPP	−30	30	60
No DR	−55	17	72
(c) k-NN			
IG	45	67	22
Q-α	21	55	34
LS	20	54	34
Random	3	45	42
LPP	−16	36	52
PCA	−33	27	60
No DR	−40	23	63
(d) SVDD			
IG	75	82	7
Q-α	27	58	31
Random	4	47	43
LS	−3	43	46
LPP	−13	38	51
No DR	−41	24	65
PCA	−49	20	69
(e) Total			
IG	228	292	64
Q-α	81	218	137
LS	44	197	153
Random	39	196	157
LPP	−86	133	219
PCA	−106	123	229
No DR	−200	74	274

For each dataset and classifier, each pair of DR techniques are compared and the win/tie/loss result is recorded. On average synchronising the goals of the DR and the classifier results in more consistent improvements

We want to highlight that the winning classifier of the review is the Gaussian. It gets the best overall performance in several of the datasets and ranks high in most of them. This could be due to the key advantage that it does not require model selection, an issue that is important for the rest of the classifiers. However, it is also the case that simplicity and natural models work very well when a correct representation for the inputs is used (Holte 1993).

In Fig. 2 we show the evolution of the sensitivity/specificity tradeoff for the Gaussian classifier across four datasets. The same tradeoff is shown in Fig. 3, this time fixing the dataset (DelftPump5x3Noisy) and varying the classifiers. Sensitivity and specificity are closely related to the dimensionality of the input space and it is clear that the selection of the optimal number of dimensions is critical. How to do it in practice, when we won't have an objective loss function to drive the selection, is a challenging problem. Practical approaches that

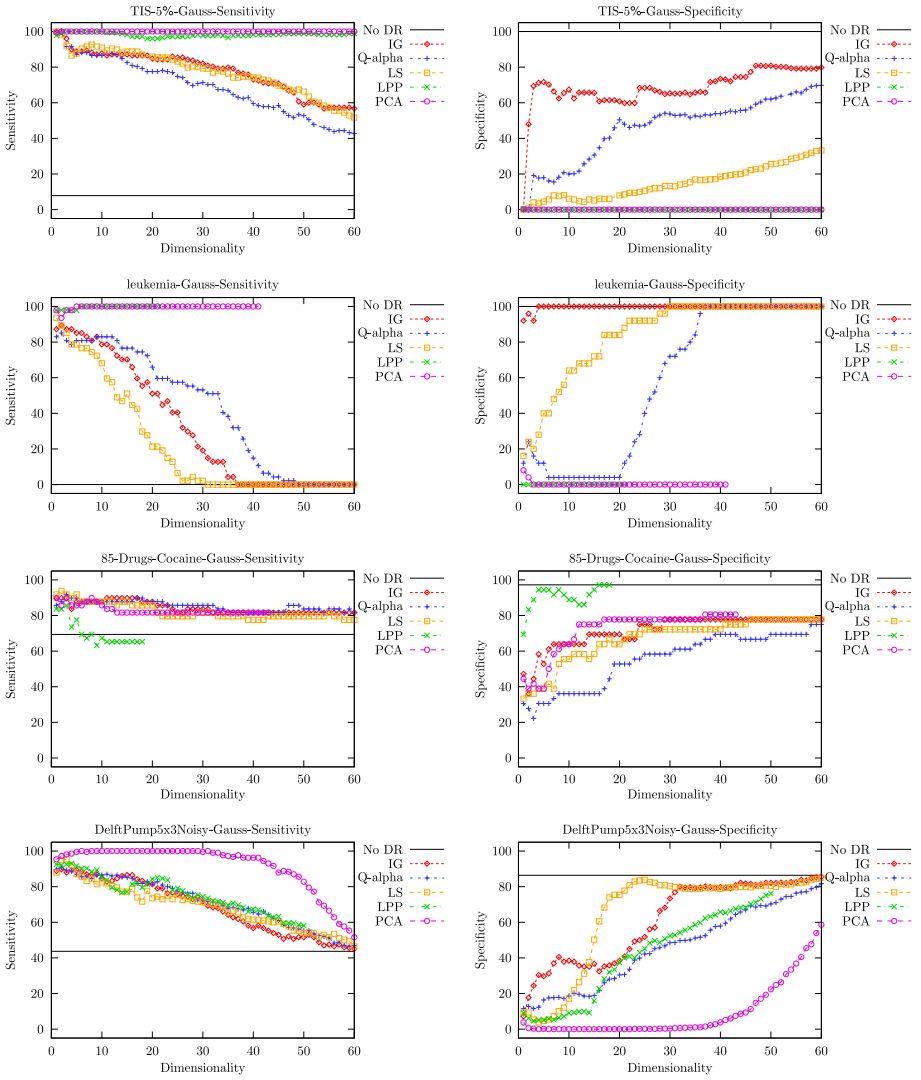

Fig. 2 Evolution of the sensitivity (left) versus specificity (right) tradeoff for the Gaussian model across four datasets: TIS-5%, leukemia, 85Drugs, DelftPump5x3Noisy. Increasing target dimensionality up to 60. The general rule is that the higher the dimensionality, the better the specificity and the worse the sensitivity

estimates this as a by-product of the dimensionality reduction technique appear to us as the most sensible ones.

Empirically it usually holds that the more the dimensionality is reduced, the better the sensitivity and the worse the specificity. When the dimensionality is high the descriptions are inaccurate and so the classifiers become accept-all or, more often, reject-all machines. Although usually the best tradeoff point would be the desired, this effect can be used for cost sensitive applications which seek better performance on one of the sides of the error, whether sensitivity or specificity.

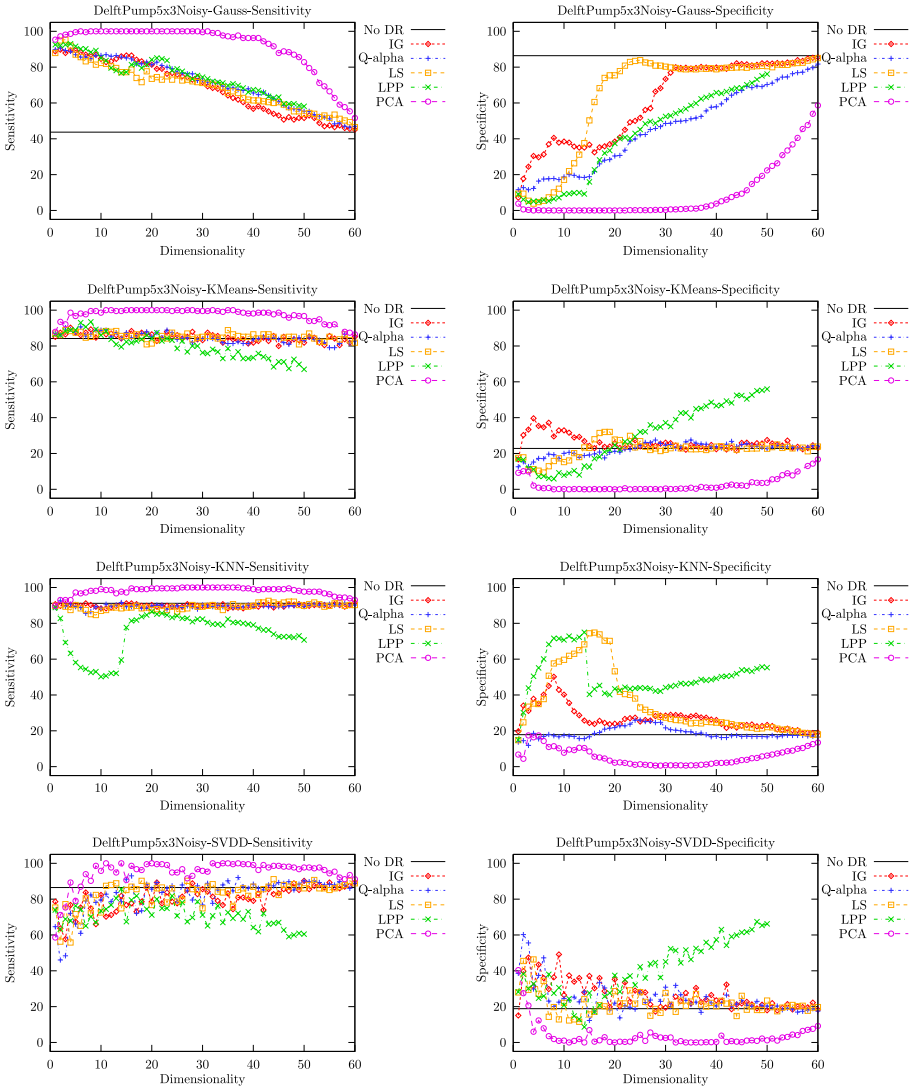

Fig. 3 Evolution of the sensitivity (left) versus specificity (right) tradeoff in the DelftPump5x3Noisy dataset for the different classifiers: Gauss, k-means, k-NN and SVDD. Increasing the target dimensionality from 1 up to 60

4.4 Further results

DR in the OCC setting is a very ill-defined problem. There are two related problems (a) a lack of an estimatable loss function (ELF) and (b) a lack of knowledge of the actual distribution of the negative examples in the input space (KAD). ELF renders the problem harder because it makes it impossible to use cross validation techniques for parameter setting based on loss minimisation. KAD force us to take into account all possible distributions, making conservative but possibly suboptimal approaches the most sensible ones.

The KAD problem is our concern in this last part of our evaluation. In order to investigate if in practice the presence of negative examples can also be beneficial for the application of unsupervised DR techniques, we conducted a different experiment. This time we allowed the unsupervised DR techniques to see the negative data. The whole process resembles now the *cheating* supervised dimensionality reduction applied in Sect. 4.3. The difference is that in this case the unsupervised techniques have access to a more complete distribution of data but not to class labels. This new setting is useful for relaxed OCC problems where we actually have negative or unlabeled examples at training time, even if they are not fully representative of the whole negatives space. Examples can be found in the information retrieval domain, represented in this evaluation by the text and iofl datasets. We show these results in Table 5.

If we use a proper model for the problem at hand, dimensionality reduction can help the classifier. This is obvious, but the way they can help differs in the two cases we compare. In short, with access to only the positive data we can only aspire to find *packing* representations—structures that makes the target class occupy as little volume as possible. When allowed to see data for both positives and negatives, rather than packing representations we can aspire to find *discriminative* ones—structures that specifically separate the two classes. Both cases are helpful for OCCs. The former because, in the absence of further knowledge of the negatives, packing is the best we can do to account for all possible distributions of negative data. The later is also useful, because we can account for the particularities of the actual negatives of which we have knowledge.

Table 5 highlights that PCA works much better when we allow it to see the negatives. There are two reasons for this. One is technical: for low sample sizes the matrices to be decomposed by PCA (and LPP) become singular, so the solution to the eigen-problem becomes unstable and the directions found noisy. Obviously this problem gets attenuated when using more data. For example, in fbis the average size of the training set in each of the 10 folds is of 83

Table 5 Best BAR achieved by the Gaussian classifier in two cases: computing the DR using only positives (L from "legal") or using also the negatives (*C* from *"cheating"*, in italics)

	IG	Q-α		LS		LPP		PCA	
	C	L	*C*	L	*C*	L	*C*	L	*C*
fbis	*76.7*	53.8	***65.0***	57.3	***64.1***	51.3	***58.9***	50.6	***72.7***
oh5	*85.3*	50.7	***58.5***	50.0	***53.1***	50.0	***56.1***	50.0	***72.7***
re0	*72.1*	52.8	***64.4***	51.5	***59.8***	45.9	***49.8***	50.0	***64.2***
tr12	*79.2*	52.6	***64.2***	52.7	***56.6***	50.2	***70.8***	48.0	***65.4***
iofl-Indoor	*78.3*	**76.1**	*75.5*	**86.3**	*79.2*	**76.3**	*62.5*	78.3	***91.3***
iofl-Leaves	*70.0*	71.1	***77.8***	**73.1**	*68.9*	**70.8**	*53.1*	70.3	***78.9***
mfeat-pixel-7	*89.6*	**91.6**	*89.6*	85.7	***87.8***	80.0	***93.2***	**97.6**	*93.8*
mfeat-pixel-8	*78.5*	**84.4**	*80.6*	79.0	***79.7***	77.4	***92.9***	**94.0**	*93.7*
Bronchiolitis	*72.1*	**72.6**	*68.5*	**72.6**	*70.9*	**73.9**	*68.5*	**73.4**	*70.9*
Arrhythmia	*77.6*	70.0	*70.0*	68.9	***69.5***	62.6	***68.7***	70.6	***76.3***
TIS-5%	*82.7*	64.0	***81.5***	53.9	***63.7***	50.0	***53.9***	50.0	***76.8***
Leukemia	*93.6*	**66.0**	*61.3*	66.0	***70.3***	50.0	***85.6***	52.9	***91.7***
85Drugs	*83.6*	**81.9**	*77.9*	**81.9**	*79.3*	**87.3**	*71.3*	81.1	*81.1*
DelftPump5x3Noisy	*74.2*	**64.4**	*64.2*	**79.6**	*77.8*	67.2	***73.3***	55.1	***76.3***
Sonar	*65.7*	69.0	***69.8***	**70.6**	*69.5*	64.0	***66.5***	65.4	***65.9***

For each case, the winning result is highlighted with boldface. For Q-α, LS and LPP the differences presumably depend on whether the model best align to the positive-class or to the whole-space structures, and whether those structures are useful to describe or to discriminate. For PCA the results are way better in the cheating setting, presumably due to a packing versus spreading dilemma which clearly favors packing when seeing only positive data and spreading when it has access to both positives and negatives

examples when we only keep positives while it is of 2,217 when keeping both positives and negatives. Often the solution with more data will be numerically better.

The second reason is what makes the application of PCA a bad idea in a substantial number of one-class problems. After all what PCA does is to find decorrelated dimensions in which the data variance is large. That is, we are finding dimensions where the data has a large spread. Theoretically spreading the data has nothing to do with finding discriminative directions. However, based on geometrical intuitions, and supported by the results presented in Table 5, we can distinguish two different scenarios when forecasting the effectiveness of PCA—if it has access to positives only or if it can see both positives and negatives.

In the pure one-class setting, with no negatives at all at training time, spreading the data is a bad idea. Because of our total ignorance of the negatives, the approach should be to maximize the chance that, whatever is its distribution, we will accept as few of them as possible. This is achieved by projections that make the positive data occupy as little space as possible (packing), which in PCA corresponds to those explaining less variance (Tax and Muller 2003). See Fig. 4 for an explanatory example in low dimensions.

On the other hand, PCA can be useful more often when having access to negatives. This conjecture is based on this observation: in the real world, we will usually face types of classification problems where there will be class separability in at least some subspace. Often separability comes together with high variability between the classes and so, with large spread in the whole data. If projecting into those discriminative subspaces will spread the data as a side effect, in practice we can take the reverse path and find high variability subspaces with the hope that they will lead to class separability.

For example, Gaussianity is a common data generating process (that is the reason why it is also called Normality). In a real world classification problem we could have a Gaussian generating each class, and these Gaussians would differ only in their means. The directions in which the data objects are separable, where the Gaussians do not overlap, will probably account for more variability than the directions in which each single class varies the most. Therefore, by projecting the data onto those directions of high "global variability", a discriminative representation would emerge. A toy example is shown in Fig. 5.

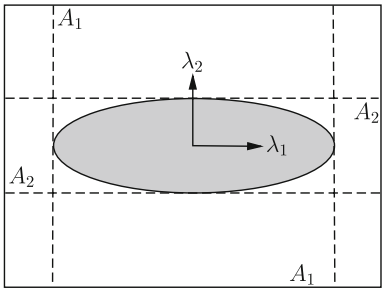

Fig. 4 An example for a two-dimensional classification problem. The positives are distributed in the ellipsoidal area. PCA will find directions λ_1 and λ_2, where λ_1 captures more variance and is ranked as the first principal component. When projecting onto λ_1, negatives in the area marked as A_1 will be classified as positive. When projecting onto λ_2, only those negatives in A_2 will be misclassified. Since we actually don't know where the negatives live in the space, selecting λ_2 seems a more sensible option. We call λ_2 a "packing dimension" [the coincidence of the term with the fractal packing dimension (Falconer 1990) is deliberate]. *Source*: Adapted from Tax and Muller (2003)

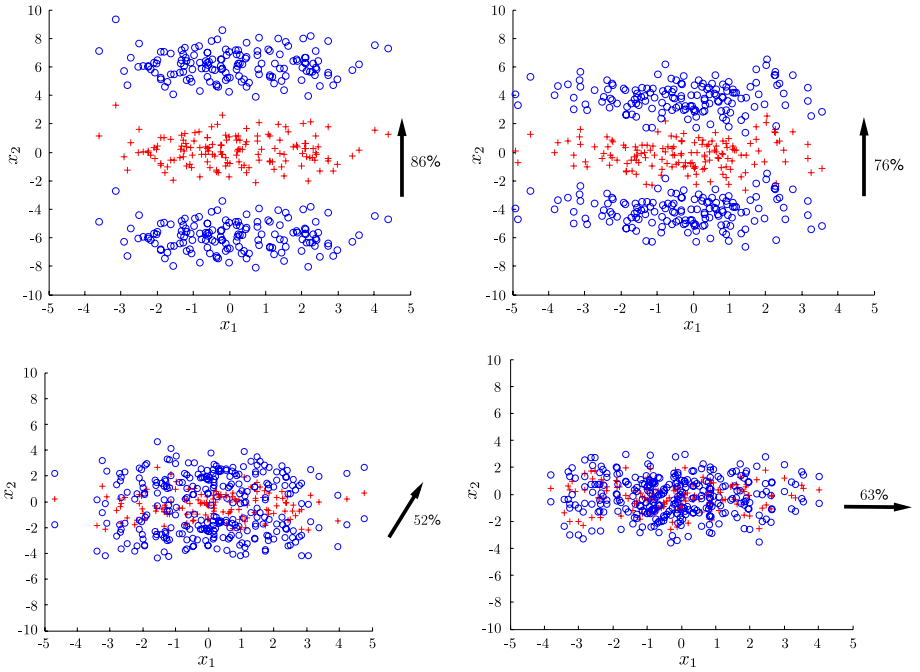

Fig. 5 PCA over an artificial two-dimensional toy example. We generate three mirroring data clouds by sampling from Gaussian distributions with diagonal covariance, the variance in x_1 ("horizontal dimension") is three times that in x_2 ("vertical dimension"), and the means differ only in x_2. We label the central cloud as the positives examples and the upper and lower clouds as the negatives, where the total number of positives and negatives is the same. When computing PCA only with the positive data, the first principal component is x_1, accounting for a 75% of the variance. This is clearly a bad option. On the right side of each plot we indicate the direction of the first principal component found by using both positives and negatives, labelled with the amount of variance it accounts for. We move the negative clouds so that they get closer and, eventually, overlap the positive cloud. In this case PCA finds "the right direction" until it is no longer possible to do so because both classes overlap

5 Conclusions

This paper reports progress in research on the applicability of DR techniques (specifically techniques from unsupervised learning) for OCC problems. We have demonstrated the potential improvements to be had by applying carefully selected DR techniques prior to one-class classification.

All the techniques for dimensionality reduction that we have evaluated seem to deserve a place, at least in some circumstances, in the one-class practitioners toolbox. And since the number of one-class practitioners is increasing it is necessary to gain insight on which combinations of DR technique and OCCs are the most appropriate for the problem at hand. It is also important to develop specific DR tools aimed at the one-class problem. As it is the case in other fields of machine learning, there is no such a thing as a silver bullet. In addition, it is clear that there are some problems where the specific characteristics of all LPP, LS, Q-α and PCA are effective. General guidelines can be inferred from studies like this one, but in general elevated performances can only be achieved by a combination of domain specific knowledge and data analysis skills.

The results presented discourage the application of PCA when we only have access to positives at training time. This is because accentuating the variability within the target class is often a bad idea for one-class classification. Without any knowledge of the actual negatives distribution, we need to allow that all types of distributions are possible. The best strategy is then to pack the positives as much as we can. When considering PCA in practice the question of which components are being used, whether the principal, the minor or a mix of both, should be tackled. In specific problems, such as when the input space features are highly correlated, choosing the high variance dimensions will still produce an elevated performance. Another not so obvious hint is to let PCA see as much data as possible, being positives or not; the rationale behind this suggestion is that there will be a good number of real problems where high variability between the classes arises naturally in subspaces that are relevant for classification.

As already stated, *locality preservation* seems an appropriate criterion for OCC tasks but it contains the implication that none or very few of the input features are irrelevant; they may be just redundant. Laying aside the specific details of the way locality preservation is modelled in LPP and LS, the key issue here is how meaningful the distance functions used to define it are. We encounter the paradoxical situation that for reducing the dimensionality of the data one needs to rely on distance measures which, most probably, are not meaningful in the original high dimensional space. How to learn a proper metric from the data itself instead of imposing a pre-specified one is an active research field in several areas of classification. For one-class classification, once again, the current techniques are not directly applicable because they use information from both sides of the classification boundary. However, related techniques could lead to useful one-class metric learning techniques.

Model selection is another tricky issue in this context that we have not addressed in the current study. Therefore the results can be regarded as unfairly biased for those techniques for which parameter tuning can have a dramatic impact on performance. Obviously this is an aspect that must be taken into account when seeking for maximum performance in real applications, so further investigation on which criteria are to be used and on the algorithmic details on how to conduct parallel and interlaced model selection for both DR and OCC is necessary. Also finding principled means for automatical selection of the threshold value for the resemblance function and of the optimal target dimensionality is a difficult, if not unsolvable, challenge.

The evaluation suggests that there is much to be gained by applying supervised dimensionality reduction techniques together with OCCs. How to accommodate unlabelled and/or actual negatives to relax or tackle specific instances of the OCC problem are related lines of research. Casting the pure OCC problem as a supervised one is possible by assuming a generation process to create artificial negative examples (e.g. sampling from the uniform distribution). This approach is supported by theoretical and practical studies (Fan et al. 2004; Steinwart et al. 2005; Abe et al. 2006; Yaniv and Nisenson 2006; Scott and Nowak 2006). Related heuristics have been used in OCC for tasks such as model selection by volume estimation (Tax and Duin 2002); however the curse of dimensionality renders volume estimation by sampling and counting unfeasible for even moderate dimensional problems because it requires an exhaustive coverage of too much space. Our bet here is that, on the contrary, this approach can be useful for dimensionality reduction even in high dimensional spaces, provided that the generated sample quality is good in terms of diversity and divergence from the specific feature interactions in the positive data; this could be achieved with much smaller artificial sample sizes.

Acknowledgements This Research is supported by Enterprise Ireland Commercialisation Fund Grant No. CFTD/05/222 and by Science Foundation Ireland Grant No. 05/IN.1/I24.

Appendix A: Win–tie–loss analysis

Table 6 Pairwise win/tie/loss analysis for the dimensionality reduction techniques on a per classifier basis (a–d) and across all the classifiers (e)

	IG	Random	Q-α	LS	LPP	PCA
(a) Gauss						
No DR	0,0,15	1,1,13	2,0,13	1,2,12	2,2,11	4,1,10
IG		10,1,4	10,0,5	9,0,6	12,0,3	11,0,4
Random			9,1,5	9,1,5	12,0,3	10,0,5
Q-α				7,1,7	11,0,4	10,0,5
LS					12,1,2	10,1,4
LPP						6,2,7
(b) k-Means						
No DR	0,0,15	1,0,14	2,1,12	2,0,13	8,0,7	4,0,11
IG		13,0,2	10,0,5	11,1,3	14,0,1	13,0,2
Random			6,0,9	8,0,7	9,0,6	8,0,7
Q-α				8,0,7	10,0,5	10,0,5
LS					9,0,6	9,0,6
LPP						5,0,10
(c) k-NN						
No DR	3,0,12	1,2,12	2,1,12	3,1,11	6,0,9	8,0,7
IG		10,1,4	11,0,4	10,0,5	12,0,3	12,0,3
Random			5,0,10	5,0,10	9,0,6	10,0,5
Q-α				8,0,7	9,0,6	12,0,3
LS					10,0,5	11,1,3
LPP						7,2,6
(d) SVDD						
No DR	0,0,15	2,0,13	3,1,11	5,0,10	6,0,9	8,0,7
IG		13,0,2	13,0,2	14,1,0	12,0,3	15,0,0
Random			4,0,11	8,0,7	8,0,7	12,0,3
Q-α				12,0,3	9,0,6	13,0,2
LS					10,0,5	13,0,2
LPP						8,1,6
(e) Totals						
No DR	3,0,57	5,3,52	9,3,48	11,3,46	22,2,36	24,1,35
IG		46,2,12	44,0,16	44,2,14	50,0,10	51,0,9
Random			24,1,35	30,1,29	38,0,22	40,0,20
Q-α				35,1,24	39,0,21	45,0,15
LS					41,1,18	43,2,15
LPP						26,5,29

Table 7 Pairwise win/tie/loss analysis for the classifier techniques on a per dimensionality reduction basis (a–g) and across all the dimensionality reduction techniques (h)

	k-Means	*k*-NN	SVDD		Dominance	Win	Loss
(a) No DR							
Gauss	5,0,10	6,0,9	5,2,8	*k*-Means	13	29	16
k-Means		9,0,6	10,0,5	*k*-NN	3	24	21
k-NN			9,0,6	SVDD	−5	19	24
				Gauss	−11	16	27
(b) IG							
Gauss	11,0,4	11,0,4	7,0,8	Gauss	13	29	16
k-Means		9,0,6	8,0,7	SVDD	−1	22	23
k-NN			8,0,7	*k*-Means	−3	21	24
				k-NN	−9	18	27
(c) Random							
Gauss	10,0,5	12,0,3	12,1,2	Gauss	24	34	10
k-Means		12,1,2	12,0,3	*k*-Means	14	29	15
k-NN			9,0,6	*k*-NN	−16	14	30
				SVDD	−22	11	33
(d) Q-α							
Gauss	6,0,9	11,0,4	9,0,6	Gauss	7	26	19
k-Means		8,0,7	9,0,6	*k*-Means	7	26	19
k-NN			7,0,8	SVDD	−5	20	25
				k-NN	−9	18	27
(e) LS							
Gauss	9,0,6	11,0,4	11,0,4	Gauss	17	31	14
k-Means		10,0,5	10,0,5	*k*-Means	7	26	19
k-NN			9,0,6	*k*-NN	−9	18	27
				SVDD	−15	15	30
(f) LPP							
Gauss	6,2,7	6,2,7	8,1,6	*k*-Means	5	23	18
k-Means		7,1,7	9,1,5	*k*-NN	2	21	19
k-NN			7,2,6	Gauss	0	20	20
				SVDD	−7	17	24
(g) PCA							
Gauss	9,1,5	8,1,6	10,1,4	*k*-Means	16	30	14
k-Means		11,0,4	14,0,1	Gauss	12	27	15
k-NN			6,2,7	*k*-NN	−10	16	26
				SVDD	−18	12	30
(h) Total							
Gauss	56,3,46	65,3,37	62,5,38	Gauss	62	183	121
k-Means		66,2,37	72,1,32	*k*-Means	59	184	125
k-NN			55,4,46	*k*-NN	−48	129	177
				SVDD	−73	116	189

References

Abe N, Zadrozny B, Langford J (2006) Outlier detection by active learning. In: 12th ACM SIGKDD international conference on knowledge discovery and data mining, August 20–23 2006, Philadelphia, USA

Asuncion A, Newman D (2007) UCI machine learning repository. http://www.ics.uci.edu/~mlearn/MLRepository.html, Accessed 28 Aug 2008

Doyle D, Cunningham P, Walsh P (2006) An evaluation of the usefulness of explanation in a case-based reasoning system for decision support in Bronchiolitis treatment. Comput Intell 22(3-4):269–281

Falconer K (1990) Fractal geometry: mathematical foundations and applications. Wiley, Chichester

Fan W, Miller M, Stolfo S, Lee W, Chan P (2004) Using artificial anomalies to detect unknown and known network intrusions. Knowl Inf Syst 6(5):507–527
François D, Wertz V, Verleysen M (2005) About the locality of kernels in high-dimensional spaces. In: ASMDA, international symposium on applied stochastic models and data analysis, May 17–20 2005, Brest, France
Golub T, Slonim D, Tamayo P, Huard C, Gaasenbeek M, Mesirov J, Coller H, Loh M, Downing J, Caligiuri M, Bloomfield C, Lander E (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286(5439):531–537
Han E, Karypis G (2000) Centroid-based document classification: analysis and experimental results. In: PKDD: European conference on the principles of data mining and knowledge discovery, September 13–16 2000, Lyon, France
Harmeling S, Dornhege G, Tax DMJ, Meinecke F, Muller KR (2006) From outliers to prototypes: ordering data. Neurocomputing 69(13-15):1608–1618
He X (2005) Locality preserving projections. PhD thesis, University of Chicago
He X, Niyogi P (2003) Locality preserving projections. In: NIPS: advances in neural information processing systems, December 9–11 2003, Vancouver, Canada
He X, Cai D, Niyogi P (2005) Laplacian score for feature selection. In: NIPS: advances in neural information processing systems, December 8–10 2005, Vancouver, Canada
Hodge V, Austin J (2004) A survey of outlier detection methodologies. Artif Intell Rev V22(2):85–126
Holte RC (1993) Very simple classification rules perform well on most commonly used datasets. Mach Learn 11(1):63–90
Jolliffe IT (2002) Principal component analysis. Springer, New York
Juszczak P (2006) Learning to recognise, a study on one-class classification and active learning. PhD thesis, Delft University of Technology
Juszczak P (2007) Volume-based model selection for one-class classifiers that consist of a set of spheres. In: ICONIP: international conference on neural information processing, November 13–16 2007, Kitakyushu, Japan
Ledoux M (2001) The concentration of measure phenomenon. American Mathematical Society
Liu H, Wong L (2003) Data mining tools for biological sequences. J Bioinf Comput Biol 1(1):139–167
Manevitz LM, Yousef M (2001) One-class SVMs for document classification. J Mach Learn Res 2:139–154
Markou M, Singh S (2003a) Novelty detection: a review—part 1: statistical approaches. Signal Process 83(12):2481–2497
Markou M, Singh S (2003b) Novelty detection: a review—part 2: neural network based approaches. Signal Process 83(12):2499–2521
Marsland S (2003) Novelty detection in learning systems. Neural Comput Surv 3:157–195
Ng AY, Jordan MI, Weiss Y (2001) On spectral clustering: analysis and an algorithm. In: NIPS: advances in neural information processing systems, December 3–6 2001, Vancouver, Canada
Rieck K, Laskov P (2007) Language models for detection of unknown attacks in network traffic. J Comput Virol 2(4):243–256
Ryder AG (2002) Classification of narcotics in solid mixtures using principal component analysis and raman spectroscopy. J Forensic Sci 47(2):275–284
Saul LK, Weinberger KQ, Ham JH, Sha F, Lee DD (2006) Spectral methods for dimensionality reduction. In: Schölkopf B, Chapelle O, Zien A (eds) Semisupervised learning. The MIT Press, Chap. 16
Schölkopf B, Smola AJ (2001) Learning with kernels: support vector machines, regularization, optimization, and beyond (adaptive computation and machine learning). The MIT Press
Scott CD, Nowak RD (2006) Learning minimum volume sets. J Mach Learn Res 7:665–704
Steinwart I, Hush D, Scovel C (2005) A classification framework for anomaly detection. J Mach Learn Res 6:211–232
Tax DMJ (2001) One-class classification. Concept learning in the absence of counterexamples. PhD thesis, Delft University of Technology
Tax DMJ (2007) DDtools, the data description toolbox for Matlab
Tax DMJ, Duin RPW (2002) Uniform object generation for optimizing one-class classifiers. J Mach Learn Res 2:155–173
Tax DMJ, Duin RPW (2004) Support vector data description. J Mach Learn Res 54(1):45–66
Tax DMJ, Muller KR (2003) Feature extraction for one-class classification. In: ICANN/ICONIP: joint international conference on artificial neural networks and neural information processing, June 26–29 2003, Istambul, Turkey
Tax DMJ, Muller KR (2004) A consistency-based model selection for one-class classification. In: ICPR: international conference of pattern recognition, August 23–26 2004, Cambridge, England

Verleysen M, François D (2005) The curse of dimensionality in data mining and time series prediction. In: IWANN: international work-conference on artificial neural networks (invited talk), June 8–10 2005, Vilanova i la Geltru, Spain

Wolf L, Shashua A (2005) Feature selection for unsupervised and supervised inference: The emergence of sparsity in a weight-based approach. J Mach Learn Res 6:1855–1887

Yaniv RE, Nisenson M (2006) Optimal single-class classification strategies. In: NIPS: advances in neural information processing systems, December 4–7 2006, Vancouver, Canada

Ypma A (2001) Learning methods for machine vibration analysis and health monitoring. PhD thesis, Pattern Recognition Group, Department of Applied Physics, Delft University of Technology