The Knowledge Engineering Review, page 1 of 30. © Cambridge University Press, 2014
doi:10.1017/S026988891300043X

One-class classification: taxonomy of study and review of techniques

SHEHROZ S. KHAN[1] and MICHAEL G. MADDEN[2]

[1]*David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, Canada N2L 3G1; e-mail: shehroz@gmail.com;*
[2]*College of Engineering and Informatics, National University of Ireland, Galway, Republic of Ireland; e-mail: michael.madden@nuigalway.ie*

Abstract

One-class classification (OCC) algorithms aim to build classification models when the negative class is either absent, poorly sampled or not well defined. This unique situation constrains the learning of efficient classifiers by defining class boundary just with the knowledge of positive class. The OCC problem has been considered and applied under many research themes, such as outlier/novelty detection and concept learning. In this paper, we present a unified view of the general problem of OCC by presenting a taxonomy of study for OCC problems, which is based on the availability of training data, algorithms used and the application domains applied. We further delve into each of the categories of the proposed taxonomy and present a comprehensive literature review of the OCC algorithms, techniques and methodologies with a focus on their significance, limitations and applications. We conclude our paper by discussing some open research problems in the field of OCC and present our vision for future research.

1 Introduction to one-class classification (OCC)

The traditional multi-class classification paradigm aims to classify an unknown data object into one of several pre-defined categories (two in the simplest case of binary classification). A problem arises when the unknown data object does not belong to any of those categories. Let us assume that we have a training data set comprising of instances of fruits and vegetables. Any binary classifier can be applied to this problem, if an unknown test object (within the domain of fruits and vegetables, e.g. apple or potato) is given for classification. But if the test data object is from an entirely different domain (e.g. a cat from the category animals), the classifier will always classify the cat as either a fruit or a vegetable, which is a wrong result in both the cases. Sometimes the classification task is just not to allocate a test object into predefined categories but to decide if it belongs to a particular class or not. In the above example an apple belongs to class fruits and the cat does not.

In OCC (Tax, 2001; Tax & Duin, 2001b), one of the classes (which we will arbitrarily refer to as the positive or target class) is well characterized by instances in the training data, while the other class (negative or outlier) has either no instances or very few of them, or they do not form a statistically representative sample of the negative concept. To motivate the importance of OCC, let us consider some scenarios. A situation may occur, for instance, where we want to monitor faults in a machine. A classifier should detect when the machine is showing abnormal/faulty behavior. Measurements on the normal operation of the machine (positive class training data) are easy to obtain. On the other hand, most possible faults would not have occurred in reality, hence we may have little or no training data for the negative class. Also, we may not want to wait until such

faults occur as they may involve high cost, machine malfunction or risk to human operators. Another example is the automatic diagnosis of a disease. It is relatively easy to compile positive data (all patients who are known to have a 'common' disease) but negative data may be difficult to obtain since other patients in the database cannot be assumed to be negative cases if they have never been tested, and such tests can be expensive. Alternatively, if the disease is 'rare' it is difficult to collect positive samples until a sufficiently large group has contracted that disease, which is an unsatisfactory approach. As another example, a traditional binary classifier for text or web pages requires arduous pre-processing to collect negative training examples. For example, in order to construct a *homepage* classifier (Yu *et al*., 2002), sample of homepages (positive training examples) and a sample of non-homepages (negative training examples) need to be gleaned. In this situation, collection of negative training examples is challenging because it may either result in improper sampling of positive and negative classes or it may introduce subjective biases.

The outline of the rest of this paper paper is as follows. Section 2 compares OCC with multi-class classification and discusses the performance measures employed for OCC algorithms. Section 3 provides an overview of related reviews of OCC. In Section 4, we propose a taxonomy for the study of OCC algorithms and present a comprehensive review of the current state of the art and significant research contributions under the branches of the proposed taxonomy. Section 5 concludes our presentation with a discussion on some open research problems and our vision for the future of research in OCC.

2 One-class classification vs. multi-class classification

In a conventional multi-class classification problem, data from two (or more) classes are available and the decision boundary is supported by the presence of data objects from each class. Most conventional classifiers assume more or less equally balanced data classes and do not work well when any class is severely under-sampled or is completely absent. It appears that Minter (1975) was the first to use the term 'single-class classification' four decades ago, in the context of learning Bayes classifier that requires only labeled data from the 'class of interest'. Much later, Moya *et al.* (1993) originate the term *One-Class Classification* in their research work. Different researchers have used other terms such as *Outlier Detection*[1] (Ritter & Gallegos, 1997), *Novelty Detection*[2] (Bishop, 1994), *Concept Learning* (Japkowicz, 1999) or *Single Class Classification* (Munroe & Madden, 2005; Yu, 2005; El-Yaniv & Nisenson, 2007). These terms originate as a result of different applications to which OCC has been applied. Juszczak (2006) defines One-Class Classifiers as *class descriptors that are able to learn restricted domains in a multi-dimensional pattern space using primarily just a positive set of examples.*

As observed by Tax (2001), the problems that are encountered in the conventional classification problems, such as the estimation of the classification error, measuring the complexity of a solution, the curse of dimensionality, the generalization of the classification method also appear in OCC and sometimes become even more prominent. As stated earlier, in OCC tasks either the negative data objects are absent or available in limited amount, so only one side of the classification boundary can be determined using only positive data (or some negatives). This makes the problem of OCC harder than the problem of conventional two-class classification. The task in OCC is to define a classification boundary around the positive class, such that it accepts as many objects as possible from the positive class, while it minimizes the chance of accepting the outlier objects. In OCC, since only one side of the boundary can be determined, it is hard to decide on the basis of just one-class how tightly the boundary should fit in each of the directions around the data. It is also harder to decide which features should be used to find the best separation of the positive and outlier class objects.

[1] Readers are advised to refer to detailed literature survey on outlier detection by Chandola *et al.* (2009).

[2] Readers are advised to refer to detailed literature survey on novelty detection by Markou and Singh (2003a, 2003b).

Table 1 Confusion matrix for OCC

	Object from target class	Object from outlier class
Classified as a target object	True positive, T^+	False positive, F^+
Classified as an outlier object	False negative, F^-	True negative, T^-

Source: Tax (2001).

2.1 Measuring classification performance of one-class classifiers

As mentioned in the work of Tax (2001), a confusion matrix (see Table 1) can be constructed to compute the classification performance of one-class classifiers. To estimate the true error (as computed for multi-class classifiers), the complete probability density of both the classes should be known. In the case of OCC, the probability density of only the positive class is known. This means that only the number of positive class objects that are not accepted by the one-class classifier (i.e. the false negatives, F^-) can be minimized. In the absence of examples and sample distribution from outlier class objects, it is not possible to estimate the number of outliers objects that will be accepted by the one-class classifier (the false positives, F^+). Furthermore, it can be noted that since $T^+ + F^- = 1$ and $T^- + F^+ = 1$, the main complication in OCC is that only T^+ and F^- can be estimated and nothing is known about F^+ and T^-. Therefore, a limited amount of outlier class data is required to estimate the performance and generalize the classification accuracy of a one-class classifier. However, during testing if the outlier class is not presented in a reasonable proportion, then the actual accuracy values of the one-class classifier could be manipulated and they may not be the true representative of the metric; this point is explored in detail by Glavin and Madden (2009). In such imbalanced data set scenarios, several other performance metrics can also be useful, such as *f*-score, geometric mean, etc. (Nguyen *et al*., 2009).

3 Related Review Work in OCC

In recent years, there has been a considerable amount of research work carried out in the field of OCC. Researchers have proposed several OCC algorithms to deal with various classification problems. Mazhelis (2006) presents a review of OCC algorithms and analyzed its suitability in the context of mobile-masquerader detection. In that paper, Mazhelis proposes a taxonomy of one-class classifiers classification techniques based on

1. The internal model used by classifier (density, reconstruction or boundary based),
2. the type of data (numeric or symbolic), and
3. the ability of classifiers to take into account temporal relations among feature (yes or no).

Mazhelis's survey of OCC describes a lot of algorithms and techniques; however, it covers a sub-spectrum of the problems in the field of OCC. As we describe in subsequent sections, OCC techniques have been developed and used by various researchers by different names in different contexts. The survey presented by Mazhelis (2006) proposes a taxonomy suitable to evaluate the applicability of OCC to the specific application domain of mobile-masquerader detection.

Brew *et al*. (2007) present a review of several OCC algorithms along with Gaussian Mixture models for the Speaker Verification problem. Their main work revolves around front-end processing and feature extraction from speech data, and speaker and imposter modeling and using them in OCC framework. Kennedy *et al*. (2009) discuss some issues related to OCC and presents a review of several OCC approaches that includes statistical, neural networks and support vector machine-based methods. They also discuss the importance of including non-target data for building OCC models and use this study for developing credit-scoring system to identify good and bad creditors. Bergamini *et al*. (2009) present a brief overview of OCC algorithms for the

biometric applications. They identified two broad categories for OCC development as density approaches and boundary approaches. Bartkowiak (2011) presents survey of research on Anomaly, Outlier and OCC. The research survey is mostly focussed on research and their applications of OCC for detecting unknown behavior. Khan and Madden (2009) present a short survey of the recent trends in the field of OCC, wherein they present a taxonomy for the study of OCC methods. Their taxonomy is based on availability of training data, methodology used and application domains applied.

This publication is an extension of the work of Khan and Madden (2009), and is more comprehensive, in-depth and detailed. The survey in this publication identifies some important research areas, raises several open questions in the study of OCC and discusses significant contributions made by researchers (see Section 4). In this work, we neither restrict the review of literature pertaining to OCC to a particular application domain, nor to specific algorithms that are dependent on type of the data or model. Our aim is to cover as many algorithms, designs, contexts and applications where OCC has been applied in multiple ways (as shown by examples in Section 1). This publication does not intend to duplicate or re-state previous review work; little of the research work presented here may be found in the past surveys of Mazhelis (2006), Brew *et al.* (2007), Kennedy *et al.* (2009), Bergamini *et al.* (2009) or Bartkowiak (2011). Moreover, this publication encompasses a broader definition of OCC than many.

4 Proposed taxonomy

Based on the research work carried out in the field of OCC using different algorithms, methodologies and application domains, we present a unified approach to OCC by proposing a taxonomy for the study of OCC problems. The taxonomy is divided into three broad categories (see Figure 1):

1. *Availability of Training Data*: learning with positive data only or learning with positive and unlabeled data and/or some amount of outlier samples.
2. *Methodology Used*: algorithms based on One-class Support Vector Machines (OSVMs) or methodologies based on algorithms other than OSVMs.
3. *Application Domain*: OCC applied in the field of text/document classification or in other application domains.

The proposed categories are not mutually exclusive, so there may be some overlapping among the research carried out in each of these categories. However, they cover almost all of the major research conducted by using the concept of OCC in various contexts and application domains. The key contributions in most OCC research fall into one of the above-mentioned categories. In the subsequent subsections, we will consider each of these categories in detail.

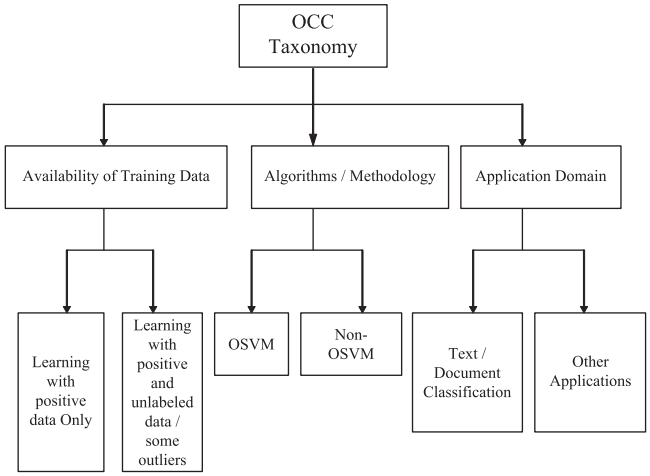

Figure 1 Our taxonomy for the study of OCC techniques. OCC=one-class classification.

4.1 Category 1: Availability of training data

Th availability of training data plays a pivotal role in any OCC algorithm. Researchers have studied OCC extensively under three broad categories:

a Learning with positive examples only.
b Learning with positive examples and some amount of poorly sampled negative examples or artificially generated outliers.
c Learning with positive and unlabeled data.

Category (c) has been a matter of much research interest among the text/document classification community (Lee & Liu, 2003; Li & Liu, 2003; Liu *et al*., 2003) that will be discussed in detail in Section 4.3.1.

Tax and Duin (1999a, 1999b) and Schölkopf *et al.* (1999b) have developed various algorithms based on support vector machines to tackle the problem of OCC using positive examples only; for a detailed discussion on them, refer to Section 4.2.1. The main idea behind these strategies is to construct a decision boundary around the positive data so as to differentiate them from the outlier/negative data.

For many learning tasks, labeled examples are rare while numerous unlabeled examples are easily available. The problem of learning with the help of unlabeled data given a small set of labelled examples was studied by Blum and Mitchell (1998) by using the concept of co-training. The co-training approach can be applied when a data set has natural separation of their features and classifiers are built incrementally on them. Blum and Mitchell demonstrate the use of co-training methods to train the classifiers in the application of text classification. Under the assumptions that each set of the features is sufficient for classification, and the feature sets of each instance are conditionally independent given the class, they provide PAC (Probably Approximately Correct) learning (Valiant, 1984) guarantees on learning from labelled and unlabeled data and prove that unlabeled examples can boost accuracy. Denis (1998) was the first to conduct a theoretical study of PAC learning from positive and unlabeled data. Denis proved that many concept classes, specifically those that are learnable from statistical queries, can be efficiently learned in a PAC framework using positive and unlabeled data. However, the trade-off is a considerable increase in the number of examples needed to achieve learning, although it remains polynomial in size. De Comité *et al.* (1999) give evidence with both theoretical and empirical arguments that positive examples and unlabeled examples can boost accuracy of many machine learning algorithms. They note that the learning with positive and unlabeled data is possible when the weight of the target concept (i.e. the ratio of positive examples) is known by the learner, which in turn can be estimated from a small set of labelled examples. Muggleton (2001) presents a theoretical study in the Bayesian framework where the distribution of functions and examples are assumed to be known. Liu *et al.* (2002) extend Muggleton's result to the noisy case; they present sample complexity results for learning by maximizing the number of unlabeled examples labelled as negative while constraining the classifier to label all the positive examples correctly. Further details on the research carried out on training classifiers with labelled positive and unlabeled data is presented in Section 4.3.1.

4.2 Category 2: Algorithms used

Most of the major OCC algorithms development can be classified under two broad categories, as has been done either using

- OSVMs or
- Non-OSVMs methods (including various flavours of neural networks, decision trees, nearest neighbors and others).

It may appear at first that this category presents a biased view, by classifying algorithms according to whether or not they are based on OSVM vs. Non-OSVM. However, we have found that the advancements, applications, significance and difference that OSVM-based algorithms

have shown opens it up as a separate research area in its own right. Nonetheless, Non-OSVM-based OCC algorithms have also been used for tackling specific research problems; the details are presented in the following sub-sections.

4.2.1 OSVM

Tax and Duin (1999a, 1999b) seek to solve the problem of OCC by distinguishing the positive class from all other possible data objects in the pattern space. They constructed a hyper-sphere around the positive class data that encompasses almost all points in the data set with the minimum radius. This method is called the Support Vector Data Description (SVDD; see Figure 2). The SVDD classifier rejects a given test point as outlier if it falls outside the hyper-sphere. However, SVDD can reject some fraction of positively labelled data when the volume of the hyper-sphere decreases. The hyper-sphere model of the SVDD can be made more flexible by introducing kernel functions. Tax (2001) considers Polynomial and a Gaussian kernel and found that the Gaussian kernel works better for most data sets considered (Figure 3). Tax uses different values for the width of the kernel. The larger the width of the kernel, the fewer support vectors are selected and the description becomes more spherical. Also, using the Gaussian kernel instead of the Polynomial kernel results in tighter descriptions, but it requires more data to support more flexible boundary. Tax's method becomes inefficient when the data set has high dimension. It also does not work well

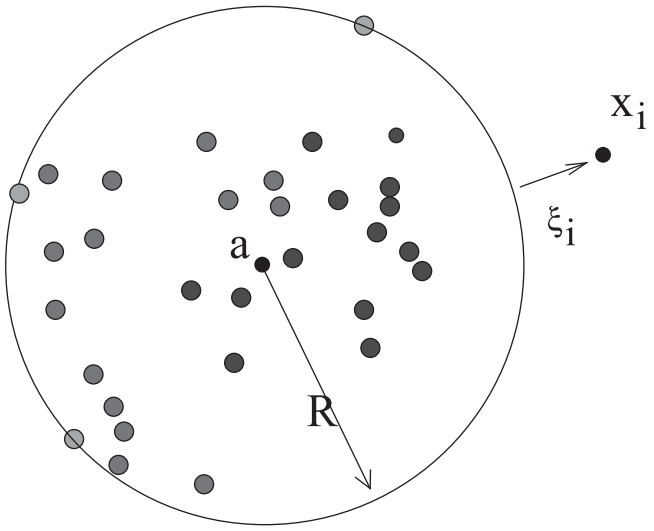

Figure 2 The hyper-sphere containing the target data, with center a and radius R. Three objects are on the boundary are the support vectors. One object x_i is outlier and has $\xi > 0$. *Source*: Tax (2001).

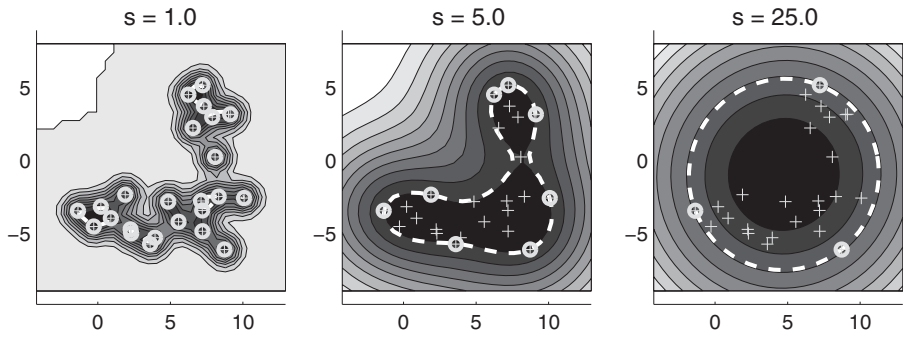

Figure 3 Data description trained on a banana-shaped data set. The kernel is a Gaussian kernel with different width sizes *s*. Support vectors are indicated by the solid circles; the dashed line is the description boundary. *Source*: Tax (2001).

when large variations in density exist among the positive-class objects; in such case, it starts rejecting the low-density target points as outliers. Tax (2001) demonstrates the usefulness of the approach on machine fault diagnostic data and handwritten digit data.

Tax and Duin (2001b) propose a sophisticated method that uses artificially generated outliers to optimize the OSVM parameters in order to balance between over-fitting and under-fitting. The fraction of the outliers accepted by the classifier is an estimate of the volume of the feature space covered by the classifier. To compute the error without the use of outlier examples, they uniformly generate artificial outliers in and around the target class. If a hyper-cube is used, then in high dimensional feature space it becomes infeasible. In that case, the outlier objects generated from a hyper-cube will have very low probability to be accepted by the classifier. The volume in which the artificial outliers are generated has to fit as tightly as possible around the target class. To make this procedure applicable in high dimensional feature spaces, Tax and Duin propose to generate outliers uniformly in a hyper-sphere. This is done by transforming objects generated from a Gaussian distribution. Their experiments suggest that the procedure to artificially generate outliers in a hyper-sphere is feasible for up to 30 dimensions.

Schölkopf *et al.* (2000, 1999a) present an alternative approach to SVDD. In their method, they construct a hyper-plane instead of a hyper-sphere around the data, such that this hyper-plane is maximally distant from the origin and can separate the regions that contain no data. They propose to use a binary function that returns $+1$ in 'small' region containing the data and -1 elsewhere. They introduce a variable that controls the effect of outliers, that is, the hardness or softness of the boundary around the data. Schölkopf *et al.* (1999a) suggest the use of different kernels, corresponding to a variety of non-linear estimators. In practical implementations, the method of Schölkopf *et al.* and the SVDD method of Tax and Duin (2001b) operate comparably and both perform best when the Gaussian kernel is used. As mentioned by Campbell and Bennett (2001), the origin plays a crucial role in the methods of both Schölkopf *et al.* and Tax and Duin, which is a drawback since the origin effectively acts as a prior for where the abnormal class instances are assumed to lie; this is termed as the problem of origin. Schölkopf *et al.* (1999a) have tested their method on both synthetic and real-world data, including the US Postal Services data set of handwritten digits. Their experiments show that the algorithm indeed extracts data objects that are difficult to be assigned to their respective classes and a number of outliers were in fact identified.

Manevitz and Yousef (2001) investigate the use of OSVM for information retrieval. Their paper proposes a different version of the OSVM than that proposed by Schölkopf *et al.* (1999b); the method of Manevitz and Yousef is based on identifying outlier data that is representative of the second class. The idea of their methodology is to work first in the feature space, and assume that not only the origin is member of the outlier class, but also all data points close to the origin are considered as noise or outliers (see Figure 4). Geometrically speaking, the vectors lying on standard sub-spaces of small dimension, that is, axes, faces, etc., are to be treated as outliers. Hence, if a vector has few non-zero entries, then this indicates that the data object shares very few items with the chosen feature subset of the database and will be treated as an outlier. Linear, sigmoid, polynomial and radial basis kernels were used in their work. Manevitz and Yousef (2001) evaluate the results on the Reuters data set[3] using the 10 most frequent categories. Their results are generally somewhat worse than the OSVM (Schölkopf *et al.*, 1999a). However, they observe that when the number of categories are increased, their version of OSVM obtains better results. Li *et al.* (2003) present an improved version of the approach of Schölkopf *et al.* (1999a) for detecting anomaly in an intrusion detection system, with higher accuracy. Their idea is to consider all points 'close enough' to the origin as outliers and not just the origin as the member of second class (see Figure 5). Zhao *et al.* (2005) use this method for customer churn prediction for the wireless industry data. They investigate the performance of different kernel functions for this version of

[3] `http://www.daviddlewis.com/resources/testcollections/reuters21578/` (Accessed January 2012).

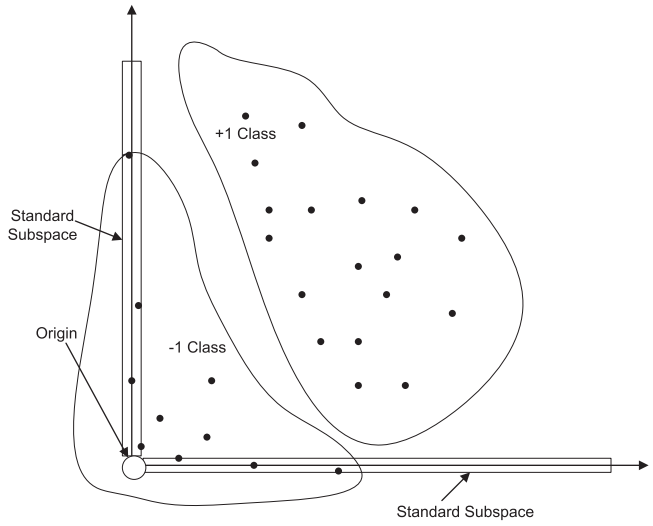

Figure 4 Outlier SVM Classifier. The origin and small subspaces are the original members of the second class. *Source*: Manevitz and Yousef (2001). SVM=Support Vector Machines.

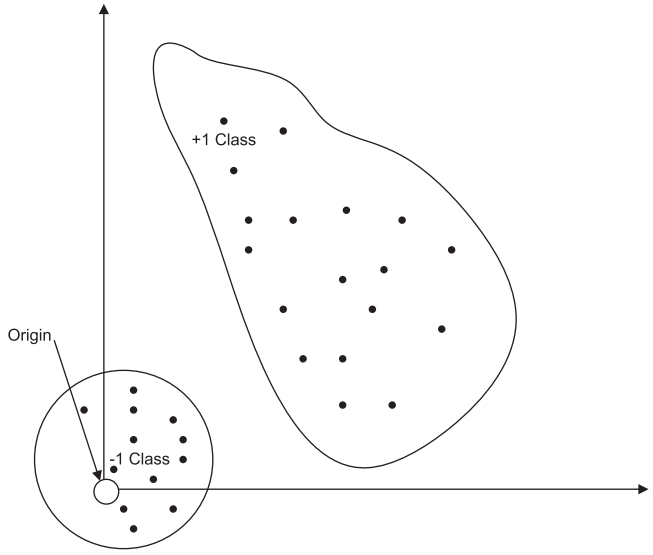

Figure 5 Improved OSVM. *Source*: Li *et al.* (2003). OSVM=One-class Support Vector Machines.

OSVM, and show that the Gaussian kernel function can detect more churners than the Polynomial and Linear kernel.

An extension to the work of Tax and Duin (1999a, 1999b) and Schölkopf *et al.* (2000) is proposed by Campbell and Bennett (2001). They present a kernel OCC algorithm that uses linear programming techniques instead of quadratic programming. They construct a surface in the input space that envelopes the data, such that the data points within this surface are considered targets and outside it are regarded as outliers. In the feature space, this problem condenses to finding a hyper-plane, which is pulled onto the data points and the margin remains either positive or zero. To fit the hyper-plane as tightly as possible, the mean value of the output of the function is minimized. To accommodate outliers, a soft margin is introduced around the hyper-plane. Their algorithm avoids the problem of the origin (stated earlier) by attracting the hyper-plane toward the center of data distribution rather than by repelling it away from a point outside the

data distribution. In their work, different kernels are used to create hyper-planes and they show that the Radial Basis Function kernel can produce closed boundaries in input space while other kernels may not (Campbell and Bennett, 2001). A drawback of their method is that it is highly dependent on the choice of kernel width parameter, σ. However, if the data size is large and contains some outliers then σ can be estimated. They show their results on artificial data set, Biomedical Data and Condition Monitoring data for machine fault diagnosis.

Yang and Madden (2007) apply particle swarm optimization to calibrate the parameters of OSVM (Schölkopf *et al.*, 2000) and find experimentally that their method either matches or surpass the performance of OSVM with parameters optimized using grid search method, while using lower CPU time. Tian and Gu (2010) propose a refinement to the Schölkopf's OSVM model (Schölkopf *et al.*, 2000) by searching optimal parameters using particle swarm optimization algorithm (Kennedy & Eberhart, 1995) and improving the original decision function with a boundary movement. Their experiments show that after adjusting the threshold, the final decision function gives a higher detection rate and a lower rejection rate.

Luo *et al.* (2007) extends the work of Tax and Duin (2001b) to propose a cost-sensitive OSVM algorithm called Frequency-Based SVDD (F-SVDD) and Write-Related SVDD (WS-SVDD) for intrusion detection problem. The SVDD method gives equal cost to classification errors, whereas F-SVDD gives higher cost to frequent short sequences occurring during system calls and WS-SVDD gives different costs to different system calls. Their experiments suggest that giving different cost or importance to system users than to processes results in higher performance in intrusion detection than SVDD. Yang *et al.* (2010b) propose a neighborhood-based OSVM method for fMRI (functional magnetic resonance imaging) data, where the objective is to classify individauls as having schizophrenia or not, based on fMRI scans of their brains. In their formulation of OSVM, they assume the neighborhood consistency hypothesis used by Chen *et al.* (2002). By integrating it with OSVM, they compute primal values which denote distance between points and hyper-plane in kernel space. For each voxel in an fMRI image, a new decision value is computed using the primal values and their neighbors. If this decision value is greater than a given threshold then it is regarded as activated voxel, otherwise it is regarded as non-activated. Their experiments on various brain fMRI data sets show that it gives more stable results than K-Means and fuzzy K-Means clustering algorithms.

Yu (2005) proposes a OCC algorithm with support vector machines (SVMs) using positive and unlabeled data and without labelled negative data and discuss some of the limitations of other OSVM-based OCC algorithms (Manevitz & Yousef, 2001; Tax & Duin, 2001b). In assessing the performance of OSVMs under a scenario of learning with unlabeled data and no negative examples, Yu comments that to induce an accurate class boundary around the positive data set, OSVM requires a larger amount of training data. The support vectors in such a case come only from positive examples and cannot create a proper class boundary, which also leads to either over-fitting or under-fitting of the data (See Figure 6). Yu notes that when the numbers of support vectors in OSVM were increased, it overfits the data rather than being more accurate. Yu (2003) presents an OCC algorithm called Mapping Convergence (MC) to induce accurate class boundary around the positive data set in the presence of unlabeled data and without negative examples. The algorithm has two phases: mapping and convergence. In the first phase, a weak classifier (e.g. Rocchio, 1971) is used to extract strong negatives (those that are far from the class boundary of the positive data) from the unlabeled data. In the second phase, a base classifier (e.g. SVM) is used iteratively to maximize the margin between positive and strong negatives for better approximation of the class boundary. Yu (2003) also presents another algorithm called Support Vector Mapping Convergence (SVMC) that works faster than the MC algorithm. At every iteration, SVMC only uses minimal data so that the accuracy of class boundary is not degraded and the training time of SVM is also saved. However, the final class boundary is slightly less accurate than the one obtained by employing MC. They show that MC and SVMC perform better than other OSVM algorithm and can generate accurate boundaries comparable to standard SVM with fully labelled data.

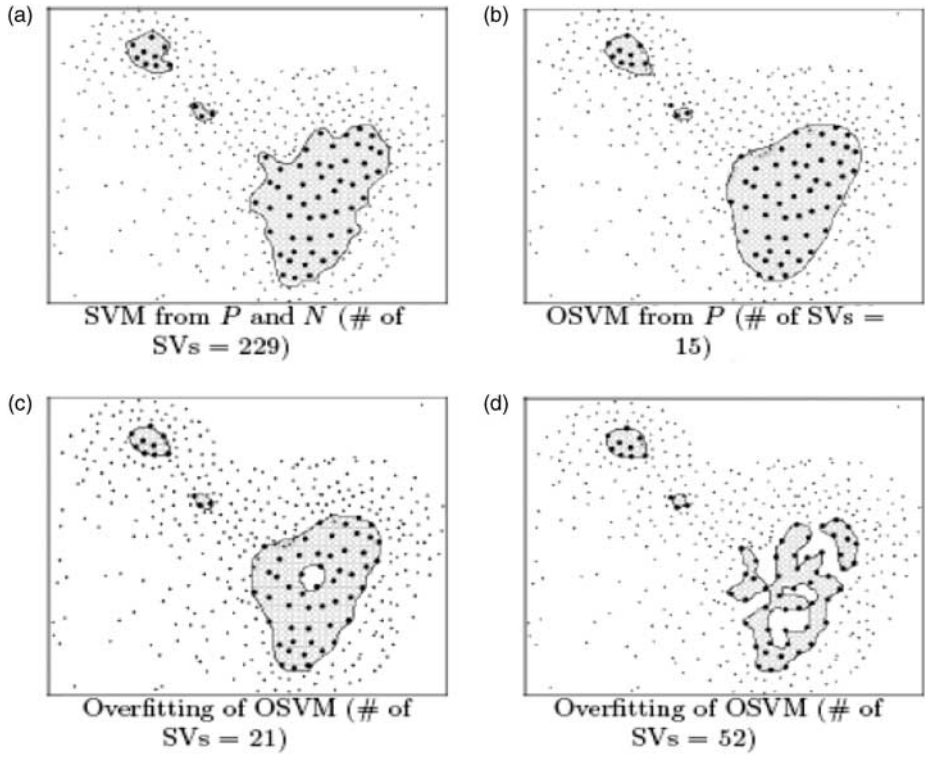

Figure 6 Boundaries of SVM and OSVM on a synthetic data set: big dots: positive data, small dots: negative data. *Source*: Yu (2005). SVM=Support Vector Machines.

Hao (2008) incorporates the concept of fuzzy set theory into OSVM (Schölkopf *et al.*, 1999a) in order to deal with the problem that standard OSVM can be sensitive to outliers and noise. The main idea of Hao's method is that different training data objects may contribute differently to the classification, which can be estimated using fuzzy membership. Two versions of the fuzzy one-class classifier are proposed: (i) a crisp hyper-plane is constructed to separate target class from origin and fuzzy membership is associated to training data, where noisy and outlier data can be given low fuzzy membership values, (ii) a fuzzy hyper-plane is constructed to discriminate target class from others. Hao uses Gaussian kernels with different parameter settings and test their method on handwritten digits problem. Another fuzzy OSVM classifier is proposed by Choi and Kim (2004) for generating visually salient and semantically important video segments. Their algorithm gives different weights to the importance measure of the video segments and then estimates their support. They demonstrate the performance of their algorithm on several synthesized data sets and different types of videos.

4.2.2 One-class classifiers other than OSVMs

In this section, we review some major OCC algorithms that are based on one-class ensembles, neural networks, decision trees, nearest neighbors, Bayesian classifiers and other methods.

4.2.2.1 One-class classifier ensemble. As in traditional multi-class classification problems, one one-class classifier might not capture all characteristics of the data. However, using just the best classifier and discarding the classifiers with poorer performance might waste valuable information (Wolpert, 1992). To improve the performance of different classifiers, which may differ in complexity or in the underlying training algorithm used to construct them, an ensemble of classifiers is a viable solution. This may serve to increase the performance and also the robustness of the classification (Sharkey & Sharkey, 1995). Classifiers are commonly ensembled to provide a combined decision by averaging the estimated posterior probabilities. This simple algorithm is

known to give good results for multi-class problems (Tanigushi & Tresp, 1997). In the case of one-class classifiers, the situation is different. One-class classifiers cannot directly provide posterior probabilities for target (positive class) objects, because accurate information on the distribution of the outlier data is not available, as has been discussed earlier in this paper. In most cases, by assuming that the outliers are uniformly distributed, the posterior probability can be estimated. Tax (2001) mentions that in some OCC methods, distance is estimated instead of probability, and if there exists a combination of distance and probability outputs, they should be standardized before they can be combined. Having done so, the same types of combining rules as in conventional classification ensembles can be used. Tax and Duin (2001a) investigate the influence of the feature sets, their inter-dependence and the type of one-class classifiers for the best choice of combination rules. They use a Normal density and a mixture of Gaussians and the Parzen density estimation (Bishop, 1994) as two types of one-class classifiers. They use four models, the SVDD (Tax & Duin, 1999b), K-means clustering, K-center method (Ypma & Duin, 1998) and an auto-encoder neural network (Japkowicz, 1999). In their experiments, the Parzen density estimator emerges as the best individual one-class classifier on the handwritten digit pixel data set[4]. Tax (2001) shows that combining classifiers trained on different feature spaces is useful. In their experiments, the product combination rule gives the best results while the mean combination rule suffers from the fact that the area covered by the target set tends to be overestimated.

Lai *et al*. (2002) study combining one-class classifier for image database retrieval and show that combining SVDD-based classifiers improve the retrieval precision. Juszczak and Duin (2004) extend combining one-class classifierd for classifying missing data. Their idea is to form an ensemble of one-class classifiers trained on each feature or each pre-selected group of features, or to compute a dissimilarity representation from features. The ensemble is able to predict missing feature values based on the remaining classifiers. As compared with standard methods, their method is more flexible, since it requires significantly fewer classifiers and does not require re-training of the system whenever missing feature values occur. Juszczak and Duin (2004) also show that their method is robust to small sample size problems due to splitting the classification problem into several smaller ones. They compare the performance of their proposed ensemble method with standard methods used with missing features values problem on several UCI data sets (Bache & Lichman, 2013).

Ban and Abe (2006) address the problem of building a multi-class classifier based on an ensemble of one-class classifiers by studying two kinds of one-class classifiers, namely, SVDD (Tax & Duin, 1999b) and Kernel Principal Component Analysis (Schölkopf *et al*., 1998). They construct a minimum-distance-based classifier from an ensemble of one-class classifiers that is trained from each class and assigns a test data object to a given class based on its prototype distance. Their method gives comparable performance to that of SVMs on some benchmark data sets; however, it is heavily dependent on the algorithm parameters. They also comment that their process could lead to faster training and better generalization performance provided appropriate parameters are chosen.

Pękalska *et al*. (2004) use the proximity of target object to its class as a 'dissimilarity representation' (DR) and show that the discriminative properties of various DRs can be enhanced by combining them properly. They use three types of one-class classifier, namely Nearest Neighbor, Generalized Mean Class and Linear Programming Dissimilarity Data Description. They make two types of ensembles: (i) combine different DR from individual one-class classifiers into one representation after proper scaling using fixed rules, for example average, product and train single one-class classifier based on this information, (ii) combine different DR of training objects over several base classifiers using majority voting rule. Their results show that both methods perform significantly better than the OCC trained with a single representation. Nanni (2006) studies combining several one-class classifiers using the random subspace method (Ho, 1998) for the problem of online signature verification. Nanni's method generates new training sets by selecting features randomly and then employing OCC on them; the final

[4] `ftp://ftp.ics.uci.edu/pub/machine-learning-databases/mfeat/` (Accessed August 2013).

results of these classifiers are combined through the max rule. Nanni uses several one-class classifiers: Gaussian model description, Mixture of Gaussian Descriptions, Nearest Neighbor Method Description, PCA Description (PCAD), Linear Programming Description (LPD), SVDD and Parzen Window Classifier. It is shown that fusion of various classifiers can reduce the error and the best fusion method is the combination of LPD and PCAD. Cheplygina and Tax (2011) propose to apply pruning to random sub-spaces of one-class classifiers. Their results show that pruned ensembles give better and more stable performance than the complete ensemble. Bergamini *et al.* (2008) present the use of OSVM for biometric fusion where very low false acceptance rates are required. They suggest a fusion system that may be employed at the level of feature selection, score-matching or decision-making. A normalization step is used before combining scores from different classifiers. Bergamini *et al.* use z-score normalization, min–max normalization, and column norm normalization, and classifiers are combined using various ensemble rules. They find that min–max normalization with a weighted sum gives the best results on NIST Biometric Scores Set. Gesù and Bosco (2007) present an ensemble method of combining one-class fuzzy KNN classifiers. Their classifier combining method is based on a genetic algorithm optimization procedure by using different similarity measures. Gesù and Bosco test their method on two categorical data sets and show that whenever the optimal parameters are found, fuzzy combination of one-class classifiers may improve the overall recognition rate.

Bagging (Breiman, 1996) is an ensemble method (for multi-class classification problems) that combines multiple classifiers on re-sampled data to improve classification accuracy. Tu *et al.* (2006) extends the one-class information bottleneck method for information retrieval by introducing bagging ensemble learning. The proposed ensemble emphasizes different parts of the data and results from different parameter settings are aggregated to give a final ranking, and the experimental results show improvements in image retrieval applications. Shieh and Kamm (2009) propose an ensemble method for combining OSVM (Tax and Duin, 1999a) using bagging. However, bagging is useful when the classifiers are unstable and small changes in the training data can cause large changes in the classifier outputs (Bauer & Kohavi, 1999). OSVM is not an unstable classifier as its estimated boundary always encloses the positive class, therefore directly applying bagging on OSVM is not useful. Shieh and Kamm (2009) propose a kernel density estimation method to give weights to the training data objects, such that the outliers get the least weights and the positive class members get higher weights for creating bootstrap samples. Their experiments on synthetic and real data sets show that bagging OSVMs achieve higher true positive rate. They also mention that such a method is useful in application where low false positive rates are required, such as disease diagnosis.

Boosting methods are widely used in traditional classification problems (Dietterich, 2000) for their high accuracy and ease of implementation. Rätsch *et al.* (2002) propose a boosting-like OCC algorithm based on a technique called barrier optimization (Luenberger, 1984). They also show, through an equivalence of mathematical programs, that a support vector algorithm can be translated into an equivalent boosting-like algorithm and vice versa. It has been pointed out by Schapire *et al.* (1998) that boosting and SVMs are 'essentially the same' except for the way they measure the margin or the way they optimize their weight vector: SVMs use the l_2-norm to implicitly compute scalar products in feature space with the help of kernel trick, whereas boosting employs the l_1-norm to perform computation explicitly in the feature space. Schapire *et al.* comment that SVMs can be thought of as a 'boosting' approach in high dimensional feature space spanned by the base hypotheses. Rätsch *et al.* (2002) exemplify this translation procedure for a new algorithm called the one-class leveraging. Building on barrier methods, a function is returned which is a convex combination of the base hypotheses that leads to the detection of outliers. They comment that the prior knowledge that is used by boosting algorithms for the choice of weak learners can be used in OCC and show the usefulness of their results on artificially generated toy data and the US Postal Service database of handwritten characters.

4.2.2.2 Neural networks. de Ridder *et al.* (1998) conduct an experimental comparison of various OCC algorithms. They compare a number of unsupervised methods from classical pattern recognition to several variations of a standard shared weight supervised neural network (Cun *et al.*, 1989) and

show that adding a hidden layer with radial basis function improves performance. Manevitz and Yousef (2000a) show that a simple neural network can be trained to filter documents when only positive information is available. They design a *bottleneck* filter that uses a basic feed-forward neural network that can incorporate the restriction of availability of only positive examples. They chose three level network with m input neurons, m output neurons and k hidden neurons, where $k < m$. The network is trained using a standard back-propagation algorithm (Rumelhart & McClelland, 1986) to learn the identity function on the positive examples. The idea is that while the bottleneck prevents learning the full identity function on m-space, the identity on the small set of examples is in fact learnable. The set of vectors for which the network acts as the identity function is more like a sub-space, which is similar to the trained set. For testing a given vector, it is shown to the network and if the result is the identity, the vector is deemed interesting (i.e. positive class) otherwise it is deemed an outlier. Manevitz and Yousef (2000b) apply the auto-associator neural network to document classification problem. During training, they check the performance values of the test set at different levels of error. The training process is stopped at the point where the performance starts a steep decline. A secondary analysis is then performed to determine an optimal threshold. Manevitz and Yousef test the method and compare it with a number of competing approaches (i.e. Neural Network, Naïve Bayes, Nearest Neighbor and Prototype algorithm) and conclude that it outperforms them.

Skabar (2003) describes how to learn a classifier based on feed-forward neural network using positive examples and corpus of unlabeled data containing both positive and negative examples. In a conventional feed-forward binary neural network classifier, positive examples are labelled as 1 and negative examples as 0. The output of the network represents the probability that an unknown example belongs to the target class, with a threshold of 0.5 typically used to decide which class an unknown sample belongs to. However, in this case, since unlabeled data can contain some unlabeled positive examples, the output of the trained neural network may be less than or equal to the actual probability that an example belongs to the positive class. If it is assumed that the labelled positive examples adequately represent the positive concept, it can be hypothesized that the neural network will be able to draw a class boundary between negative and positive examples. Skabar shows the application of the technique to the prediction of mineral deposit location.

4.2.2.3 Decision trees. Several researchers have used decision tress to classify positive samples from a corpus of unlabeled examples. De Comité *et al.* (1999) present experimental results showing that positive examples and unlabeled data can efficiently boost accuracy of the statistical query learning algorithms for monotone conjunctions in the presence of classification noise and present experimental results for decision tree induction. They modify standard C4.5 algorithm (Quinlan, 1993) to get an algorithm that uses unlabeled and positive data and show the relevance of their method on UCI data sets. Letouzey *et al.* (2000) design an algorithm which is based on positive statistical queries (estimates for probabilities over the set of positive instances) and instance statistical queries (estimates for probabilities over the instance space). The algorithm guesses the weight of the target concept, that is, the ratio of positive instances in the instance space and then uses a hypothesis testing algorithm. They show that the algorithm can be estimated in polynomial time and is learnable from positive statistical queries and instance statistical queries only. Then, they design a decision tree induction algorithm, called POSC4.5, using only positive and unlabeled data and present experimental results on UCI data sets that are comparable to the C4.5 algorithm. Yu (2005) comments that such rule learning methods are simple and efficient for learning nominal features but are tricky to use for data with continuous features, high dimensions, or sparse instance spaces. Li and Zhang (2008) perform bagging ensemble on POSC4.5 and classify test samples using the majority voting rule. Their result on UCI data sets shows that the classification accuracy and robustness of POSC4.5 could be improved by applying their technique. Based on very fast decision trees (VFDT; Domingos & Hulten, 2000) and POSC4.5, Li *et al.* (2009) propose a one-class VFDT for data streams with applications to credit fraud detection and intrusion detection. They state that by using the proposed algorithm, even if 80% of the data is unlabeled,

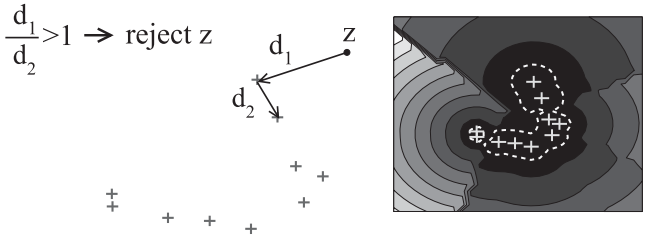

Figure 7 The nearest neighbor data description. *Source*: Tax (2001).

the performance of one-class VFDT is very close to the standard VFDT algorithm. Désir *et al.* (2012) propose a one-class Random Forest algorithm that internally conjoins bagging and random feature selection (RFS) for decision trees. They note that the number of artificial outliers to be generated to convert an one-class classifier to a binary classifier can be exponential with respect to the size of the feature space and availability of positive data objects. They propose a novel algorithm to generate outliers in small feature spaces by combining RFS and Random Subspace methods. Their method aims at generating more artificial outliers in the regions where the target data objects are sparsely populated and less in areas where the density of target data objects is high. They compare their method against OSVM, Gaussian Estimator, Parzen Windows and Mixture of Gaussian Models on an image medical data set and two UCI data sets and show that it performs equally well or better than the other algorithms.

4.2.2.4 Nearest neighbors. Tax (2001) presents a one-class Nearest Neighbor method, called Nearest Neighbor Description (*NN-d*), where a test object z is accepted as a member of target class provided that its local density is greater than or equal to the local density of its nearest neighbor in the training set. The first nearest neighbor is used for the local density estimation (1-*NN*). The following acceptance function is used:

$$f_{NN^{tr}}(z) = I\left(\frac{\|z - NN^{tr}(z)\|}{\|NN^{tr}(z) - NN^{tr}(NN^{tr}(z))\|}\right) \tag{1}$$

which shows that the distance from object z to its nearest neighbor in the training set NN^{tr} (z) is compared with the distance from this nearest neighbor NN^{tr} (z) to its nearest neighbor (see Figure 7).

The *NN-d* has several predefined choices to tune various parameters. Different numbers of nearest neighbors can be considered; however, increasing the number of neighbors will decrease the local sensitivity of the method, but it will make the method less sensitive to noise. Instead of 1-*NN*, the distance to the kth nearest neighbor or the average of the k distances to the first k neighbors can also be used. The value of the threshold (default 1.0) can be changed to either higher or lower values to change the detection sensitivity of the classifier. Tax and Duin (2000) proposes a nearest neighbor method capable of finding data boundaries when the sample size is very low. The boundary thus constructed can be used to detect the targets and outliers. However, this method has the disadvantage that it relies on the individual positions of the objects in the target set. Their method seems to be useful in situations where the data is distributed in subspaces. They test the technique on both real and artificial data and find it to be useful when very small amounts of training data exist (fewer than five samples per feature).

Datta (1997) modify the standard nearest neighbor algorithm that is appropriate to learn a single class or the positive class. Their modified algorithm, *NNPC* (nearest neighbor positive class), takes examples from only one class as input. *NNPC* learns a constant δ, which is the maximum distance a test example can be from any learned example and still be considered a member of the positive class. Any test data object that has a distance greater than δ from any training data object will not be considered a member of the positive class. The variable δ is calculated by

$$\delta = Max\{\forall x Min\{\forall y \neq x dist(x, y)\}\} \tag{2}$$

where x and y are two examples of the positive class, and Euclidean distance, $dist\ (x, y)$ is used as the distance function. Datta also experimented with another similar modification which involves learning a vector $\delta_1, \ldots, \delta_n$, where δ_i is the threshold for the ith example. This modification records the distance to the closest example for each example. δ_i is calculated by

$$\delta_i = Min\{\forall y \neq x_i dist(x_i, y)\} \tag{3}$$

where x_i is the ith training example. To classify a test example the same classification rule as above is used.

Munroe and Madden (2005) extend the idea of one-class KNN to tackle the recognition of vehicles using a set of features extracted from their frontal view, and present results showing high accuracy in classification. They compare their results with multi-class classification methods and comment that it is not reasonable to draw direct comparisons between the results of the multi-class and single-class classifiers, because the use of training and testing data sets and the underlying assumptions are quite different. They also note that the performance of multi-class classifier could be made arbitrarily worse by adding those vehicle types to the test set that do not appear in the training set. Since one-class classifiers can represent the concept "none of the above", their performance should not deteriorate in these conditions. Cabral *et al.* (2007) propose a one-class nearest neighbor data description using the concept of structural risk minimization. k-Nearest Neighbors (kNN) suffers from the limitation of having to store all training samples as prototypes that would be used to classify an unseen sample. Their paper is based on the idea of removing redundant samples from the training set, thereby obtaining a compact representation aiming at improving generalization performance of the classifier. The results on artificial and UCI data sets show improved performance than the *NN-d* classifiers and also achieved considerable reduction in number of stored prototypes. Cabral *et al.* (2009) present another approach where not only 1-*NN* is considered but all of the *k*-nearest neighbors, to arrive at a decision based on majority voting. In their experiments on artificial data, biomedical data[5] and data from the UCI repository, they observe that the *k-NN* version of their classifier outperforms the 1-*NN* and is better than *NN-d* algorithms. Gesù *et al.* (2008) present a one-class KNN and test it on synthetic data that simulates microarray data for the identification of nucleosomes and linker regions across DNA. A decision rule is presented to classify an unknown sample X as

$$X = \begin{cases} 1, & \text{if} |y \in T_p \quad \text{such that} \quad \delta(y, x) \leq \varphi| \geq K \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where T_P is the training set for the data object P representing positive instance, δ is the dissimilarity function between data objects and $j = 1$ means that x is positive. The meaning of the above rule is that if there are at least K data objects in T_P with dissimilarity from x no more than φ, then x is classified as a positive data object, otherwise it is classified as an outlier. This *kNN* model depends on parameters K and φ and their values are chosen by using optimization methods. Their results have shown good recognition rate on synthetic data for nucleosome and linker regions across DNA.

de Haro-Garca *et al.* (2009) use one-class KNN along with other one-class classifiers for identifying plant/pathogen sequences, and present a comparison of results. They find that these methods are suitable owing to the fact that genomic sequences of plant are easy to obtain in comparison to pathogens, and they build one-class classifiers based only on information from the sequences of the plant. One-class kNN classifiers are used by Glavin and Madden (2009) to study the effect of unexpected outliers that might arise in classification (in their case, they considered the task of classifying spectroscopic data). According to the authors, unexpected outliers can be defined as those outliers that do not come from the same distribution of data as the postive cases or outlier cases in the training data set. Their experiments show that the one-class kNN method is more reliable for the task of detecting such outliers than a similar binary kNN classifier. The 'kernel approach' has been used by various researchers to implement different flavours of NN-based classifier for multi-class classification. Khan (2010) extends this idea and propose two variants of one-class nearest neighbor classifiers. These variants use a kernel as a distance metric instead of Euclidean diistance for identification of

[5] `http://lib.stat.cmu.edu/datasets/` (Accessed January 2012).

chlorinated solvents in the absence of non-chlorinated solvents. The first method finds the neighborhood for nearest k neighbors, while the second method finds j localized neighborhoods for each of k neighbors. Popular kernels like the polynomial kernel (of degree 1 and 2), RBF and spectroscopic kernels (Spectral Linear Kernel (Howley, 2007) and Weighted Spectral Linear Kernel (Madden & Howley, 2008)) are used in their experiments. Their kernelized *kNN* methods perform better than standard *NN-d* and 1-*NN* methods and that a one-class classifier with RBF Kernel as distance metric performs better when compared with other kernels.

4.2.2.5 Bayesian classifiers. Datta (1997) suggests a method to learn a Naïve Bayes (NB) classifier from samples of positive class data only. Traditional NB attempts to find the probability of a class given an unlabeled data object $p(C_i|A_1=v_1 \,\&\, \ldots \,\&\, A_n=v_n)$. By assuming that the attributes are independent and applying Bayes' theorem the previous calculation is proportional to:

$$\left[\prod_{j}^{attributes} p(A_j = v|C_i)\right] p(C_i) \tag{5}$$

where A_j is an attribute, v is a value of the attribute, C_i is a class and the probabilities are estimated using the training examples. When only the positive class is available, the calculation of $p(C_i)$ from the above equation cannot be done correctly. Therefore, Datta (1997) modifies NB to learn in a single class situation and calls their modification *NBPC* (Naïve Bayes Positive Class) that uses the probabilities of the attribute-values. *NBPC* computes a threshold t as

$$t = Min\left[\forall x \prod_{j}^{attributes} p(A_j = v_i)\right] \tag{6}$$

where $A_j = v_i$ is the attribute value for the example x and $p(A_j = v_i)$ is the probability of the attribute's ith value. The probabilities for the different values of attribute A_j is normalized by the probability of the most frequently occurring v. During classification, if for the test example $\prod_j^{attributes}(A_j = v_i) \geq t$, then the test example is predicted as a member of the positive class. Datta tests the above positive class algorithms on various data sets taken from UCI repository and conclude that *NNPC* (discussed in previous section) and *NBPC* have classification accuracy (both precision and recall values) close to C4.5's value, although C4.5 decision trees are learned from all classes, whereas each of the one-class classifiers is learned using only one class. Wang and Stolfo (2003) use a simple one-class NB method that uses only positive samples, for masquerade detection in a network. The idea is to generate a user's profile (u) using UNIX commands (c) and compute the conditional probability $p(c|u)$ for user u's self profile. For the non-self profile, they assume that each command has a random probability $\frac{1}{m}$. For testing a sample d, they compare the ratios $p(d|self)$ and $p(d|non\text{-}self)$. The larger the value of this ratio, it is more likely that the command d has come from user u.

4.2.2.6 Other methods. Wang *et al.* (2004) investigate several OCC methods in the context of Human–Robot interaction for image classification into faces and non-faces. Some of the important non-standard methods used in their study are Gaussian Data Description, k-Means, Principal Component Analysis and Linear Programming (Pękalska *et al.*, 2002), as well as the SVDD. They study the performance of these OCC methods on an image recognition data set and observe that SVDD attains better performance in comparison to the other OCC methods studied, due to its flexibility relative to the other methods, which use very strict models of separation, such as planar shapes. They also investigate the effect of varying the number of features and remark that more features do not always guarantee better results, because with an increase in the number of features, more training data are needed to reliably estimate the class models. Ercil and Buke (2002) report a different technique to tackle the OCC problem, based on fitting an implicit polynomial surface to the point cloud of features, to model the target class in order to separate it from the outliers. They show the utility of their method for the problem of defect classification, where there are often plentiful samples for the non-defective class but only very few samples for various defective classes. They use an implicit polynomial fitting technique and show a considerable improvement in

the classification rate, in addition to having the advantage of requiring data only from non-defective motors in the learning stage. A fuzzy one-class classifier is proposed by Bosco and Pinello (2009) for identifying patterns of signals embedded in a noisy background. They employed a Multi-Layer Model (Gesù *et al*., 2009) as a data processing step and then use their proposed fuzzy one-class classifier for testing on microarray data. Their results show that integrating these two methods could improve the overall classification results. Juszczak *et al*. (2009) propose a one-class classifier that is built on the minimum spanning tree of only the target class. The method is based on a graph representation of the target class to capture the underlying structure of the data. This method performs distance-based classification as it computes the distance from a test object to its closest edge. They show that their method performs well when the data size is small and has high dimensions. Seguì *et al*. (2010) suggest that the performance of the method of Juszczak *et al*. is reduced in the presence of outliers in the target class. To circumvent this problem, they present two bagging ensemble approaches that are aimed to reduce the influence of outliers in the target training data and show improved performance on both real and artificially contaminated data. Hempstalk *et al*. (2008) present a OCC method that combines a density estimator to draw a reference distribution and a standard model for estimating class probability for the target class. They use the reference distribution to generate artificial data for the second class and decompose this problem as a standard two-class learning problem and show that their results on UCI data set and typist data set are comparable with standard OSVM (Schölkopf *et al*., 2000), with the advantage of not having to specify a target rejection rate at the time of training. Silva and Willett (2009) present a high dimension anomaly detection method that uses limited amount of unlabeled data. Their algorithm uses a variational Expectation-Maximization (EM) method on the hyper-graph domains that allows edges to connect more than two vertices simultaneously. The resulting estimate can be used to compute degree of anomalousness based on a false-positive rate. The proposed algorithm is linear in the number of training data objects and does not require parameter tuning. They compare the method with OSVM and another *K*-point entropic graph method on synthetic data and the very high dimensional Enron email database, and conclude that their method outperforms both of the other methods.

4.3 Category 3: Application domain applied

4.3.1 Text/document classification

Traditional text classification techniques require an appropriate distribution of positive and negative examples to build a classifier; thus they are not suitable for the problem of OCC. It is of course possible to manually label some negative examples, though depending on the application domain, this may be a labor-intensive and time-consuming task. However, the core problem remains, that it is difficult or impossible to compile a set of negative samples that provides a comprehensive characterization of everything that is 'not' the target concept, as is assumed by a conventional binary classifier. It is a common practice to build text classifiers using positive and unlabeled examples[6] (in *semi-supervised* setting[7]) as collecting unlabeled samples is relatively easy and fast in many text or Web page domains (Nigam *et al*., 2000; Liu *et al*., 2002). In this section, we will discuss some of the algorithms that exploit this methodology with application to text classification.

The ability to build classifiers without negative training data is useful in a scenario if one needs to extract positive documents from many text collections or sources. Nigam *et al*. (2000) show that accuracy of text classifiers can be improved by adding small amount of labelled training data to a large available pool of unlabeled data. They introduce an algorithm based on EM and NB. The central idea of their algorithm is to train the NB classifier based on the labelled documents and then probabilistically label the unlabeled documents and repeat this process till it converges.

[6] Readers are advised to refer to survey paper by Zhang and Zuo (2008)

[7] Readers are advised to refer to survey paper on semi-supervised learning by Zhu (2005)

To improve the performance of the algorithm, they propose two variants that give weights to modulate the amount of unlabeled data, and use mixture components per class. They show that using this type of methodology can result in reduction of error by factor of up to 30%. Liu *et al.* (2003) study the problem of learning from positive and unlabeled data and suggest that many algorithms that build text classifiers are based on two steps:

1. Identifying a set of reliable/strong negative documents from the unlabeled set. In this step, Spy-EM (Liu *et al.*, 2002) uses a Spy technique, PEBL (Yu *et al.*, 2004) uses a technique called 1-DNF (Yu *et al.*, 2002), and Roc-SVM (Liu *et al.*, 2003) uses the Rocchio algorithm (Rocchio, 1971).
2. Building a set of classifiers by iteratively applying a classification algorithm and then selecting a good classifier from the set. In this step, Spy-EM uses the Expectation Maximization (EM) algorithm with a NB classifier, while PEBL and Roc-SVM use SVM. Both Spy-EM and Roc-SVM have same methods for selecting the final classifier. PEBL simply uses the last classifier at convergence, which can be a poor choice.

These two steps together work in an iterative manner to increase the number of unlabeled examples that are classified as negative, while at the same time maintain the correct classification of positive examples. It was shown theoretically by Yu *et al.* (2002) that if the sample size is large enough, maximizing the number of unlabeled examples classified as negative while constraining the positive examples to be correctly classified will give a good classifier. Liu *et al.* (2003) introduce two new methods, one for Step 1 (i.e. the NB) and one for Step 2 (i.e. SVM alone) and perform an evaluation of all 16 possible combinations of methods for Step 1 and Step 2 that were discussed above. They develop a benchmarking system called LPU (Learning from Positive and Unlabeled Data)[8] and propose an approach based on a biased formulation of SVM that allows noise (or error) in positive examples. Their experiments on Reuters and Usenet articles suggest that the biased-SVM approach outperforms all existing two-step techniques.

Yu *et al.* (2003) explore SVMC (Yu, 2003) (for detail on this technique refer to Section 4.2.1) for performing text classification without labelled negative data. They use Reuters and WebKb[9] corpora for text classification and compare their method against six other methods: Simple MC, OSVM, Standard SVM trained with positive examples and unlabeled documents substituted for negative documents, Spy-EM, NB with Negative Noise and Ideal SVM trained from completely labelled documents. They conclude that with a reasonable number of positive documents, the MC algorithm gives the best performance among all the methods they considered. Their analysis show that when the positive training data is not under-sampled, SVMC significantly outperforms other methods because SVMC tries to exploit the natural gap between positive and negative documents in the feature space, which eventually helps to improve the generalization performance. Peng *et al.* (2006) present a text classifier from positive and unlabeled documents based on Genetic Algorithms (GA) by adopting a two stage strategy (as discussed above). Firstly, reliable negative documents are identified by an improved 1-DNF algorithm. Secondly, a set of classifiers are built by iteratively applying the SVM algorithm on training data. They then discuss an approach to evaluate the weighted vote of all classifiers generated in the iteration steps, to construct the final classifier based on a GA. They comment that the GA evolving process can discover the best combination of the weights. Their experiments are performed on the Reuters data set and compared against PEBL and OSVM and it is shown that the GA based classification performs better.

Koppel and Schler (2004) study the *Authorship Verification* problem where only examples of writings of a single author is given and the task is to determine if given piece of text *is* or *is not* written by this author. Traditional approaches to text classification cannot be applied directly to this kind of classification problem. Hence, they present a new technique called 'unmasking' in

[8] `http://www.cs.uic.edu/ liub/LPU/LPU-download.html` [Accessed: Jan-2012]

[9] `http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/` [Accessed: Jan-2012]

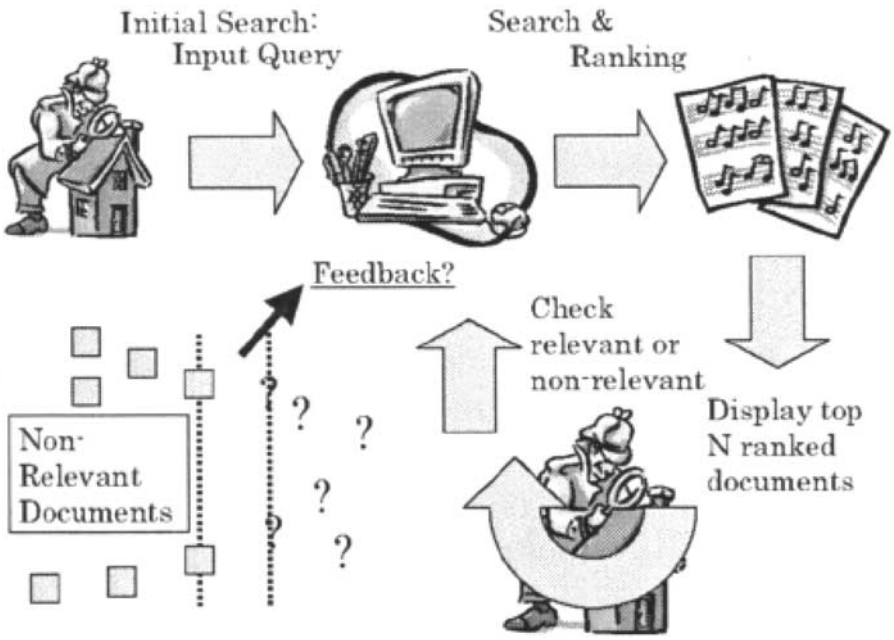

Figure 8 Outline of a problem in the relevance feedback documents retrieval. *Source*: Onoda *et al.* (2005).

which features that are most useful for distinguishing between books A and B are iteratively removed and the speed with which cross-validation accuracy degrades is gauged as more features are removed. The main hypothesis is that if books A and B are written by the same author, then no matter what differences there may be between them (of genres, themes etc), the overall essence or regularity in writing style can be captured by only a relatively small number of features. For testing the algorithm, they consider a collection of twenty-one 19th century English books written by 10 different authors and spanning a variety of genres and obtain overall accuracy of 95.7% with errors almost equally distributed between false positives and false negatives. Onoda *et al.* (2005) report a document retrieval method using non-relevant documents. Users rarely provide a precise query vector to retrieve desired documents in the first iteration. In subsequent iterations, the user evaluates whether the retrieved documents are relevant or not, and correspondingly the query vector is modified in order to reduce the difference between the query vector and documents evaluated as relevant by the user. This method is called relevance feedback. The relevance feedback needs a set of relevant and non-relevant documents to work usefully. However, sometimes the initial retrieved documents that are presented to a user do not include relevant documents. In such a scenario, traditional approaches for relevance feedback document retrieval systems do not work well, because the system needs relevant and non relevant documents to construct a binary classifier (see Figure 8). To solve this problem, Onoda *et al.* propose a feedback method using information from non-relevant documents only, called non-relevance feedback document retrieval. The design of non-relevance feedback document retrieval is based on OSVM (Schölkopf *et al.*, 1999b). Their proposed method selects documents that are discriminated as not non-relevant and that are near the discriminant hyper-plane between non-relevant document and relevant documents. They compare the proposed approach with conventional relevance feedback methods and vector space model without feedback and show that it consistently gives better performance as compared to other methods. Pan *et al.* (2008) extend the concept of classifying positive examples with unlabeled samples in the Collaborative Filtering (CF) application. In CF, the positive data is gathered based on user interaction with the web like news items recommendation or bookmarking pages, etc. However, due to ambiguous interpretations, limited knowledge or lack of interest of users, the collection of valid negative data may be hampered. Sometime negative and unlabeled positive data are severely mixed up and it becomes difficult to discern them. Manually labeling

negative data is not only intractable considering the size of the web but also will be poorly sampled. Traditional CF algorithms either label negative data, or assume missing data are negative. Both of these approaches have an inherent problem of being expensive and biased to the recommendation results. Pan *et al.* (2008) propose two approaches to one-class CF to handle the negative sparse data to balance the extent to which to treat missing values as negative examples. Their first approach is based on weighted low rank approximation (Srebro & Jaakkola, 2003) that works on the idea of providing different weights to error terms of both positive and negative examples in the objective function. Their second approach is based on sampling missing values as negative examples. They perform experiments on real world data from social bookmarking site del.icio.us and Yahoo News data set and show that their method outperforms other state of the art CF algorithms.

Denis *et al.* (2002) introduce a NB algorithm and shows its feasibility for learning from positive and unlabeled documents. The key step in their method is the estimation of word probabilities for the negative class, because negative examples were not available. This limitation can be overcome by assuming an estimate of the positive class probability (the ratio of positive documents in the set of all documents). In practical situations, the positive class probability can be empirically estimated or provided by domain knowledge. Their results on WebKB data set show that error rates of NB classifiers obtained from p positive data objects, *PNB* (Positive Naïve Bayes), trained with enough unlabeled examples are lower than error rates of NB classifiers obtained from p labeled documents. Denis *et al.* (2003) consider situations where only a small set of positive data is available together with unlabeled data. Constructing an accurate classifier in these situations may fail because of the shortage of properly sampled data. However, learning in this scenario may still be possible using the co-training framework (Blum & Mitchell, 1998) that looks for two feature views over the data. They propose a Positive Naïve Co-Training algorithm, *PNCT*, that takes a small pool of positive documents as its seed. *PNCT* first incrementally builds NB classifiers from positive and unlabeled documents over each of the two views by using PNB. Along the co-training steps, self-labelled positive examples and self-labelled negative examples are added to the training sets. A base algorithm is also proposed which is a variant of *PNB* that is able to use these self-labelled examples. The experiments on the WebKB data set show that co-training algorithms lead to significant improvement of classifiers, even when the initial seed is only composed of positive documents. Calvo *et al.* (2007) extend the *PNB* classification idea to build more complex Bayesian classifiers in the absence of negative samples when only positive and unlabeled data are present. A positive tree augmented NB (*PTAN*) in a positive-unlabeled scenario is proposed along with the use of a Beta distribution to model the *a priori* probability of the positive class, and it is applied to *PNB* and *PTAN*. The experiments suggest that when the predicting attributes are not conditionally independent, *PTAN* performs better than *PNB*. The proposed Bayesian approach to estimating *apriori* probability of the positive class also improves the performance of *PNB* and *PTAN*. He *et al.* (2010) improves the *PNB* by selecting the value of the prior probability of the positive class on the validation set using a performance measure that can be estimated from positive and unlabeled examples. Their experiments suggest that the proposed algorithm performs well even without the user specifying the prior probability of the positive class.

Pan *et al.* (2010) propose two variants of a nearest neighbor classifier for classification of uncertain data under learning from positive and unlabeled scenario. The method outperforms the *NN-d* (Tax & Duin, 2000) and OCC method by Hempstalk *et al.* (2008). Elkan and Noto (2008) show that if a classifier is trained using labelled and unlabeled data then its predicted probabilities differ from true probabilities only by a constant factor. They test their method on the application of identifying protein records and show it performs better in comparison to the standard biased SVM method (Liu *et al.*, 2003). Zhang *et al.* (2008) propose a OCC method for the classification of text streams with concept drift. In situations where text streams withe large volumes of documents arrive at high speed, it is difficult to label all of them and few positive samples are labelled. They propose a stacked ensemble approach and compare it against other window-based approaches and demonstrate its better performance. Blanchard *et al.* (2010) present a different outlook on learning

with positive and unlabeled data that develops general solution to this problem by a surrogate problem related to Neyman-Pearson classification, which is a binary classification problem subject to a constraint on false-positive rate while minimizing the false-negative rate. It is to be noted that in their problem formulation, outliers are not assumed to be rare but that special case is also discussed. They perform theoretical analysis to deduce generalization error bounds, consistency and rates of convergence for novelty detection and show that this approach optimally adapts to unknown novelty distributions, whereas the traditional methods assume a fixed uniform distribution. Their experiments compare the proposed method with OSVM on several data sets[10] and find comparable performance. Learning from positive and unlabeled data has been studied in other domains as well apart from text classification such as facial expression recognition (Cohen *et al*., 2003), gene regulation networks (Cerulo *et al*., 2010), etc.

4.3.2 Other Application Domains

In this subsection we will highlight some of the other applications of OCC methods that may not necessarily employ learning from positive and unlabeled data. Some of these areas are: Handwriting Detection (Schölkopf *et al*., 2000; Tax, 2001; Tax & Duin, 2001b; Hempstalk *et al*., 2008); Information Retrieval (Manevitz & Yousef, 2001); Missing Data/Data Correction (Juszczak & Duin, 2004; Xiaomu *et al*., 2008); Image Database Retrieval (Chen *et al*., 2001; Tax & Duin, 2001b; Gondra *et al*., 2004; Seo, 2007); Face/Object Recognition Applications (Wang *et al*., 2004; Bicego *et al*., 2005; Zeng *et al*., 2006), Remote Sensing (Li *et al*., 2011); Stream Mining (Zhang *et al*., 2008, 2010; Liu *et al*., 2011); Chemometrics and Spectroscopy (Xu & Brereton, 2007; Glavin & Madden, 2009; Hao *et al*., 2010; Khan, 2010; Kittiwachana *et al*., 2010); Biometrics (Bergamini *et al*., 2009, 2008); Assistive Technologies (Yang *et al*., 2010a; Khan *et al*., 2012b); Time Series Analysis (Sachs *et al*., 2006; Nguyen *et al*., 2011); Disease Detection (Cohen *et al*., 2004; Zhang *et al*., 2011); Medical Analysis (Zhou *et al*., 2005; Gardner *et al*., 2006); Bioinformatics (Spinosa & Ferreira de Carvalho, 2004; Ferreira de Carvalho, 2005; Alashwal *et al*., 2006; Wang *et al*., 2006; Yousef *et al*., 2008); Steganalysis (Lyu & Farid, 2004; Rodriguez *et al*., 2007); Spam Detection (Schneider, 2004; Sun *et al*., 2005; Wu *et al*., 2005); Audio Surveillance, Sound & Speaker Classification (Brew *et al*., 2007; Rabaoui *et al*., 2007, 2008); Ship Detection (Tang & Yang, 2005); Vehicle Recognition (Munroe & Madden, 2005); Collision Detection (Quinlan *et al*., 2003); Anomaly Detection (Nguyen, 2002; Li *et al*., 2003; Tran *et al*., 2004; Yilmazel *et al*., 2005; Perdisci *et al*., 2006; Zhang *et al*., 2007); Intrusion Detection (Evangelista *et al*., 2005; Giacinto *et al*., 2005; Luo *et al*., 2007); Credit Scoring (Kennedy *et al*., 2009); Yeast Regulation Prediction (Kowalczyk & Raskutti, 2002); Customer Churn Detection (Zhao *et al*., 2005); Relevant Sentence Extraction (Kruengkrai & Jaruskulchai, 2003); Machine Vibration Analysis (Tax *et al*., 1999); Machine Fault Detection (Ercil & Buke, 2002; Tax & Duin, 2004; Sarmiento *et al*., 2005; Shin *et al*., 2005); and Recommendation Tasks (Yasutoshi, 2006). Compression neural networks for OCC have been used to detect mineral deposits (Skabar, 2003) and for fMRI Analysis (Hardoon & Manevitz, 2005a, 2005b). One-class Fuzzy ART networks have been explored to classify cancerous cells (Murshed *et al*., 1996).

5 Conclusions and open research questions

The goal of OCC algorithms is to induce generalized classifiers when only one class (the target or positive class) is well characterized by the training data and the negative or outlier class is either absent, poorly sampled or the negative concept is not well defined. The limited availability of data makes the problem of OCC more challenging and interesting. The research in the field of OCC encompasses several research themes developed over time. In this paper, we have presented a unified view on the general problem of OCC and presented a taxonomy for the study of OCC problems. We have observed that the research carried out in OCC can be broadly represented by

[10] `http://www.csie.ntu.edu.tw/ cjlin/libsvmtools/datasets/` (Accessed July 2013).

three different categories or areas of study, which depends upon the availability of training data, classification algorithms used and the application domain investigated. Based on the categories under the proposed taxonomy, we have presented a comprehensive literature survey of current state-of-the-art and significant research work in the field of OCC, discussing the techniques used and methodologies employed with a focus on their limitations, importance and applications.

Over the course of several years, new OCC algorithms have emerged and new application areas have been exploited. Although the OCC field is becoming mature, there are still several fundamental problems that are open for research, not only in describing and training classifiers, but also in scaling, controlling errors, handling outliers, using non-representative sets of negative examples, combining classifiers, generating sub-spaces, reducing dimensionality and making a fair comparison of errors with multi-class classification.

In the context of OCC, we believe that classifier ensemble methods need further exploration. Although there exist several bagging models, new techniques based on boosting and random subspace warrant further attention. Random subspace methods with one-class variants of decision trees and nearest neighbor classifiers can be an interesting research direction. The Random oracle Ensemble (Kuncheva & Rodrguez, 2007) has been shown to fare better than standard ensembles for the multi-class classification problems; however, for OCC it has been not explored. We believe that carrying out research on new ensemble methods within the domain of OCC can bring interesting results.

Another point to note here is that in OSVMs, the kernels that have been used mostly are Linear, Polynomial, Gaussian or Sigmoidal. We suggest it would be fruitful to investigate some more innovative forms of kernels, for example, Genetic Kernels (Howley & Madden, 2006) or domain specific kernels, such as Weighted Linear Spectral Kernel (Howley, 2007), that have shown greater potential in standard SVM classification. Moreover, parameter tuning of standard kernels may give biased results, therefore we believe that researchers should focus on efficiently tuning and optimizing kernel parameters. The kernels used as distance metric have shown promising results in the basic form of one-class nearest neighbor classifiers. We believe further research in this direction can show interesting insights into the problem. An important issue that has been largely ignored by OCC researchers is how to handle missing data in the target class and still be able to develop robust one-class classifiers. This kind of scenario makes the OCC problem more difficult. Khan *et al*. (2012a) address the issue of handling missing data in target class by using Bayesian Multiple Imputations (Rubin, 1987) and EM and propose several variants of one-class classifier ensemble that can perform better than traditional method of mean imputation. However, we recommend that research involving these and other advanced data imputation methods can help in building one-classifiers that can handle missingness in the data. Feature selection for one-class classifiers is a difficult problem because it is challenging to model the behavior of features for one class in terms of their discriminatory power. There are some studies in this direction (Villalba & Cunningham, 2007); however, a lack of advanced methods and techniques to handle high dimensional positive class data can still be a bottleneck in learning one-class classifiers.

In the case where abundant unlabeled examples and some positive examples are available, researchers have used many different two-step algorithms, as have been discussed in Section 4.3.1. We believe that a Bayesian Network approach to such OCC problems would be an interesting research area. When learning on only positive data using NB method, an important problem that hampers the flexibility of the model is the estimation of prior probabilities. An important research direction is to put in effort to estimate prior distribution with minimal user intervention. A possible direction in this attempt is to explore mixtures of beta distribution for inferring prior class probability (Calvo, 2008).

Normally OCC is employed to identify anomalies, outliers, unusual or unknown behaviors, and failing to identify such novel observations may be costly in terms of risks associated with health, safety and money. In general, most of the researchers assume equal cost of errors (false alarms and miss alarms), which may not be true in the case of OCC. However, traditional cost-sensitive learning methods (Ling & Sheng, 2010) may not fit here because neither the prior probability of the outlier class nor the associated cost of errors are known. Data-dependent approaches that

deduce cost of errors from the training data objects may not generalize the cost across different domains of even same application area. In the papers we reviewed, only one research paper discusses the cost-sensitive aspect of OCC (Luo *et al*., 2007) and it shows that this area of research is largely unexplored. We believe that approaches based on careful application of Preference Elicitation techniques (Chen & Pu, 2004) can be useful to deduce cost of errors. In terms of data types, most of the research work in OCC is focused on numerical or continuous data, however not much emphasis is given to categorical or mixed data. Similarly, adaptation and development of OCC methods for streaming data analysis and online classification also need more research effort. Many of the OCC algorithms perform density estimation and assume that outliers are uniformly distributed in low density regions, however, if the target data also lies in low density region than such methods may start rejecting positive data objects or if the thresholds are increased than will start accepting outliers. To tackle such scenarios, alternate formulations of OCC is required.

In terms of applications of OCC, we have seen that it has been explored in many diverse fields with very encouraging results. With the emergence of Assistive Technologies to help people with medical conditions, OCC has a potential application to map patients' individual behavior under different medical conditions, which would otherwise be very difficult to handle with multi-class classification approaches. Activity recognition is also a central problem in such domains. The data for these domains is normally captured by sensors that are prone to noise and may miss vital recordings. Multi-class classifiers are difficult to employ in these applications to detect unknown or anomalous behaviors, such as those related to the user/patient or the sensor itself. We believe OCC can play an important role in modeling these kinds of applications. In activity recognition and some computer vision problems, even the collected positive data may contain instances from negative class. Building binary classes on such unclean data is detrimental to the classification accuracy. We believe that employing OCC classifiers as a post-processing to filter out the noise from target data objects can be conducive to build better multi-class classifiers. Finally, we strongly recommend the development of open-source OCC software, tools and benchmark data sets that can be used by researchers to compare and validate results in coherent and systematic way.

This survey provides a unified and in-depth insight into the current study in the field of OCC. Depending upon the data availability, algorithm use and applications domain, appropriate OCC techniques can be applied and improved upon. We hope that the proposed taxonomy along with this survey will provide researchers with a direction to formulate future novel work in this field.

Acknowledgements

The authors are grateful to Dr D.M.J. Tax, Dr L. Manevitz, Dr K. Li, Dr H. Yu and Dr T. Onoda for their kind permission to reproduce figures from their respective papers.

References

Alashwal, H. T., Deris, S. & Othman, R. M. 2006. One-class support vector machines for protein-protein interactions prediction. *International Journal of Biomedical Sciences* **1**(2), 120–127.

Bache, K. & Lichman, M. 2013. UCI machine learning repository. Accessed on July 2013 from `http://archive.ics.uci.edu/ml`.

Ban, T. & Abe, S. 2006. Implementing multi-class classifiers by one-class classification methods. In *International Joint Conference on Neural Networks*, 327–332.

Bartkowiak, A. M. 2011. Anomaly, novelty, one-class classification: a comprehensive introduction. *International Journal of Computer Information Systems and Industrial Management Applications* **3**, 61–71.

Bauer, E. & Kohavi, R. 1999. An empirical comparison of voting classification algorithms: bagging, boosting, and variants. *Machine Learning* **36**, 105–139.

Bergamini, C., Koerich, A. L. & Sabourin, R. 2009. Combining different biometric traits with one-class classification. *Signal Processing* **89**(11), 2117–2127.

Bergamini, C., Oliveira, L. S., Koerich, A. L. & Sabourin, R. 2008. Fusion of biometric systems using one-class classification. In *Proceedings of IEEE International Joint Conference on Neural Networks*, Hong Kong, 1308–1313.

Bicego, M., Grosso, E. & Tistarelli, M. 2005. Face authentication using one-class support vector machines. In *Advances in Biometric Person Authentication: International Workshop on Biometric Recognition Systems, in Conjunction with International Conference On Computer Vision*. Lecture Notes in Computer Science **3781**, 15–22.

Bishop, C. 1994. Novelty detection and neural network validation. In *IEEE Proceedings on Vision, Image and Signal Processing*, **141** of 4, 217–222.

Blanchard, G., Lee, G. & Scott, C. 2010. Semi-supervised novelty detection. *Journal of Machine Learning Research* **11**, 2973–3009.

Blum, A. & Mitchell, T. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of 11th Annual Conference on Computation Learning Theory*, Bartlett, P. L. & Mansour, Y. (eds). ACM Press, 92–100.

Bosco, G. L. & Pinello, L. 2009. A fuzzy one class classifier for multi layer model. *Fuzzy Logic and Application*, Lecture Notes in Computer Science **5571**, 124–131.

Breiman, L. 1996. Bagging predictors. *Machine Learning* **24**, 123–140.

Brew, A., Grimaldi, M. & Cunningham, P. 2007. An evaluation of one-class classification techniques for speaker verification. *Artificial Intelligence Review* **27**(4), 295.

Cabral, G. G., Oliveira, A. L. I. & Cahu, C. B. G. 2007. A novel method for one-class classification based on the nearest neighbor data description and structural risk minimization. In *Proceedings of International Joint Conference on Neural Networks*, Orlando, FL, 1976–1981.

Cabral, G. G., Oliveira, A. L. I. & Cahu, C. B. G. 2009. Combining nearest neighbor data description and structural risk minimization for one-class classification. *Neural Computing and Applications* **18**(2), 175–183.

Calvo, B. 2008. *Positive unlabelled learning with applications in computational biology*. Phd thesis, University of the Basque Country.

Calvo, B., Larrañaga, P. & Lozano, J. A. 2007. Learning bayesian classifiers from positive and unlabeled examples. *Pattern Recognition Letters* **28**(16), 2375–2384.

Campbell, C. & Bennett, K. P. 2001. A linear programming approach to novelty detection. In *Advances in Neural Information Processing*, Leen, T. K., Dietterich, T. D. & Tresp, V. (eds). MIT Press, **14**. Cambridge, MA.

Cerulo, L., Elkan, C. & Ceccarelli, M. 2010. Learning gene regulatory networks from only positive and unlabeled data. *BMC Bioinformatics* **11**, 228.

Chandola, V., Banerjee, A. & Kumar, V. 2009. Anomaly detection – a survey. *ACM Computing Surveys* **41**(3), 15:1–15:58.

Chen, H. F., Yao, D. Z., Becker, S., Zhou, Y., Zeng, M. & Chen, L. 2002. A new method for fMRI data processing: neighborhood independent component correlation algorithm and its preliminary application. *Science in China Series F* **45**(5), 373–382.

Chen, L. & Pu, P. 2004. Survey of Preference Elicitation Methods. Technical report, EPFL.

Chen, Y., Zhou, X. & Huang, T. S. 2001. One-class SVM for learning in image retrieval. In *Proceedings of IEEE International Conference on Image Processing*, Greece.

Cheplygina, V. & Tax, D. M. J. 2011. Pruned random subspace method for one-class classifiers. In *10th International Workshop*, Sansone, C., Kittler, J. & Roli, F. (eds). *MCS 2011*, Lecture Notes in Computer Science **6713**, 96–105. Springer.

Choi, Y. S. & Kim, K. J. 2004. Video summarization using fuzzy one-class support vector machine. In Lecture Notes in Computer Science, Laganà, A., Gavrilova, M. L., Kumar, V., Mun, Y., Tan, C. J. K. & Gervasi, O. (eds). **3043**, 49–56. Springer-Verlag.

Cohen, G., Hilario, M., Sax, H., Hugonnet, S., Pellegrini, C. & Geissbuhler, A. 2004. An application of one-class support vector machines to nosocomial infection detection. In *Proceedings of Medical Informatics*.

Cohen, I., Sebe, N., Gozman, F. G., Cirelo, M. C. & Huang, T. S. 2003. Learning bayesian network classifiers for facial expression recognition both labeled and unlabeled data. In *Proceedings of 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003*, **1**. IEEE, 595–601.

Cun, Y. L., Boser, B., Denker, J.S., Henderson, D., Howard, R. E., Hubbard, W. & Jackel, L. D. 1989. Backpropagation applied to handwritten zip code recognition. *Neural Computation* **1**, 541–551.

Datta, P. 1997. *Characteristic Concept Representations*. PhD thesis, University of California Irvine.

Denis, F. 1998. PAC learning from positive statistical queries. In *Proceedings of the 9th International Conference on Algorithmic Learning Theory*, Richter, M. M., Smith, C. H., Wiehagen, R. & Zeugmann, T. (eds). Springer-Verlag, 112–126.

Denis, F., Gilleron, R. & Tommasi, M. 2002. Text classification from positive and unlabeled examples. In *Proceedings of 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, Annecy, France.

Denis, F., Laurent, A., Gilleron, R. & Tommasi, M. 2003. Text classification and co training from positive and unlabeled examples. In *Proceedings of the ICML Workshop: the Continuum from Labeled Data to Unlabeled Data in Machine Learning and Data Mining*, 80–87, Washington DC, USA.

Désir, C., Bernard, S., Petitjean, C. & Heutte, L. 2012. A random forest based approach for one class classification in medical imaging. In *Machine Learning in Medical Imaging*, Wang, F., Shen, D., Yan, P. & Suzuki, K. (eds). Lecture Notes in Computer Science **7588**, 250–257. Springer.

De Comité, F., Denis, F., Gilleron, R. & Letouzey, F. 1999. Positive and unlabeled examples help learning. In *Proceedings of the 10th International Conference on Algorithmic Learning Theory*, Watanabe, O. & Yokomori, T. (eds). Springer-Verlag, 219–230.

de Haro-Garca, A., Garca-Pedrajas, N., Romero del Castillo, J. A. & Garca-Pedrajas, M. D. 2009. One-class methods for separating plant/pathogen sequences. In *VI Congreso Espaol sobre Metaheursticas, Algoritmos Evolutivos y Bioinspirados*, Malaga, Spain.

de Ridder, D., Tax, D. M. J. & Duin, R. P. W. 1998. An experimental comparison of one-class classification methods. In *Proceedings of the 4th Annual Conference of the Advanced School for Computing and Imaging*, Delft.

Dietterich, T. G. 2000. An experimental comparison of three methods for constructing ensembles of decision trees, bagging, boosting and randomization. *Machine Learning* **40**(2), 139–157.

Domingos, P. & Hulten, G. 2000. Mining high-speed data streams. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Ramakrishnan, R., Stolfo, S. J., Bayardo, R. J. & Parsa, I. (eds). ACM, 71–80.

Elkan, C. & Noto, K. 2008. Learning classifiers from only positive and unlabeled data. In *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Li, Y., Liu, B. & Sarawagi, S. (eds). ACM, 213–220.

El-Yaniv, R. & Nisenson, M. 2006. Optimal single-class classification strategies – google scholar. In *Proceedings of the 2006 NIPS Conference*, Schölkopf, B., Platt, J. C. & Hoffman, T. (eds). **19**. MIT Press, 377–384.

Ercil, A. & Buke, B. 2002. One class classification using implicit polynomial surface fitting. In *Proceedings of the 16th International Conference on Pattern Recognition*, Quebec, Canada, **2**, 152–155.

Evangelista, P. F., Bonnisone, P., Embrechts, M. J. & Szymanski, B. K. 2005. Fuzzy ROC curves for the 1 class SVM: application to intrusion detection. In *Application to Intrusion Detection, 13th European Symposium on Artificial Neural Networks*, Burges.

Ferreira de Carvalho, A. C. P. L. 2005. Combining one-class classifiers for robust novelty detection in gene expression data. In *Brazilian Symposium on Bioinformatics*, 54–64.

Gardner, B., Krieger, A. M., Vachtsevanos, G. & Litt, B. 2006. One-class novelty detection for seizure analysis from intracranial EEG. *Journal of Machine Learning Research* **7**, 1025–1044.

Gesù, V. D. & Bosco, G. L. 2007. Combining one class fuzzy KNN's. *Applications of Fuzzy Sets Theory* Lecture Notes in Computer Science 4578, 152–160.

Gesú, V. D., Bosco, G. L. & Pinello, L. 2008. A one class classifier for signal identification: a biological case study. In *Proceedings of the 12th International Conference on Knowledge-Based Intelligent Information and Engineering Systems*, Lovrek, I., Howlett, R. J. & Jain, L. C. (eds). **5179**. Springer, 747–754.

Gesù, V. D., Bosco, G. L., Pinello, L., Yuan, G. C. & Corona, D. F. V. 2009. A multi-layer method to study genome-scale positions of nucleosomes. *Genomics* **930**(2), 140–145.

Giacinto, G., Perdisci, R. & Roli, F. 2005. Network intrusion detection by combining one-class classifiers. In *Image Analysis and Processing – ICIAP 2005*, Roli, F. & Vitulano, S. (eds). Lecture Notes in Computer Science **3617**, 58–65. Springer.

Glavin, F. G. & Madden, M. G. 2009. Analysis of the effect of unexpected outliers in the classification of spectroscopy data. In *Artificial Intelligence and Cognitive Science 2009*, Dublin.

Gondra, I., Heisterkamp, D. R. & Peng, J. 2004. Improving image retrieval performance by inter-query learning with one-class support vector machines. *Neural Computation and Applications* **13**, 130–139.

Hao, P. Y. 2008. Fuzzy one-class support vector machines. *Fuzzy Sets and Systems* **159**, 2317–2336.

Hao, L., Wen, Z. J., Sheng, C. Q., Rong, C. J. & Ping, Z. 2010. Identification of egg freshness using near infrared spectroscopy and one class support vector machine algorithm. *Spectroscopy and Spectral Analysis* **30**(4), 929–932.

Hardoon, D. R. & Manevitz, L. M. 2005a. fMRI analysis via one-class machine learning techniques. In *Proceedings of 19th International Joint Conference on Aritifical Intelligence*, Edinburgh, UK, 1604–1606.

Hardoon, D. R. & Manevitz, L. M. 2005b. One-class machine learning approach for fMRI analysis. In *In Postgraduate Research Conference in Electronics, Photonics, Communications and Networks, and Computer Science,* Lancaster.

Hempstalk, K., Frank, E. & Witten, I. H. 2008. One-class classification by combining density and class probability estimation. In *Proceedings of the 12th European Conference on Principles and Practice of Knowledge Discovery in Databases and 19th European Conference on Machine Learning*, Berlin, 505–519.

He, J., Zhang, Y., Li, X. & Wang, Y. 2010. Naive bayes classifier for positive unlabeled learning with uncertainty. In *Proceedings of the 10th SIAM International Conference on Data Mining*, USA, 361–372.

Howley, T. & Madden, M. G. 2006. An evolutionary approach to automatic kernel construction. In *Proceedings of ICANN 2006*, Lecture Notes in Computer Science **4132**, 417–426.

Howley, T. 2007. *Kernel methods for machine learning with applications to the analysis of raman spectra*. Phd thesis, National University of Ireland Galway.

Ho, T. K. 1998. The random subspace method for constructing decision forests. *IEEE Trans. Pattern Analysis and Machine Intelligence*, **200**(8): 832–844.

Japkowicz, N. 1999. *Concept-Learning in the absence of counterexamples: an autoassociation-based approach to classification*. PhD thesis, New Brunswick Rutgers, The State University of New Jersey.

Juszczak, P. 2006. *Learning to Recognise. A study on one-class classification and active learning*. PhD thesis, Delft University of Technology.

Juszczak, P. & Duin, R. P. W. 2004. Combining one-class classifiers to classify missing data. In *Proceedings of the 5th International Workshop MCS*, Roli, F., Kittler, J. & Windeatt, T. (eds). Springer-Verlag, **3077**, 92–101.

Juszczak, P., Tax, D. M. J., Pękalska, E. & Duin, R. P. W. 2009. Minimum spanning tree based one-class classifier. *Neurocomputing* **72**(7–9), 1859–1869.

Kennedy, J. & Eberhart, R. 1995. Particle swarm optimization. In *Proceedings IEEE International Conference on Neural Networks*, Piscataway, NJ, 1942–1948.

Kennedy, K., Mac Namee, B. & Delany, S. J. 2009. Credit scoring: Solving the low default portfolio problem using one-class classification. In *Proceedings of the 20th Irish Conference on Artificial Intelligence and Cognitive Science*, 168–177.

Khan, S. S., Hoey, J. & Lizotte, D. 2012a. Bayesian multiple imputation approaches for one-class classification – springer. In *Proceedings Advances in Artificial Intelligence*, Kosseim, L. & Inkpen, D. (eds). Springer, Toronto, **7310**, 331–336.

Khan, S. S., Karg, M. E., Hoey, J. & Kulic, D. 2012b. Towards the detection of unusual temporal events during activities using HMMs. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, Dey, A. K., Chu, H-H. & Hayes, G. R. (eds). UbiComp '12, ACM, 1075–1084. New York, NY, USA.

Khan, S. S. 2010. *Kernels for One-Class Nearest Neighbour Classification and Comparison of Chemical Spectral Data*. Masters thesis, National University of Ireland Galway.

Khan, S. S. & Madden, M. G. 2009. A survey of recent trends in one class classification. In Lecture Notes in Artificial Intelligence, Coyle, L. & Freyne, J. (eds). **6206**, 181–190, Springer-Verlag.

Kittiwachana, S., Ferreira, D. L. S., Lloyd, G. R., Fido, L. A., Thompson, D. R., Escott, R. E. A. & Brereton, R. G. 2010. One class classifiers for process monitoring illustrated by the application to online HPLC of a continuous process. *Journal of Chemometrics* **24**(3–4), 96–110.

Koppel, M. & Schler, J. 2004. Authorship verification as a one-class classification problem. In *Proceedings of the 21st International Conference on Machine learning*, Brodley, C. E. (ed.). ACM Press, **69**. Alberta, Canada.

Kowalczyk, A. & Raskutti, B. 2002. One class SVM for yeast regulation prediction. In *ACM SIGKDD Explorations Newsletter*, **4**. ACM, 99–100.

Kruengkrai, C. & Jaruskulchai, C. 2003. Using one-class SVMs for relevant sentence extraction. In *International Symposium on Communications and Information Technologies*, Thailand.

Kuncheva, L. I. & Rodrguez, J. J. 2007. Classifier ensembles with a random linear oracle. *IEEE Transactions on Knowledge and Data Engineering* **19**(4), 500–508.

Lai, C., Tax, D. M. J., Duin, R. P. W., Pękalska, E. & Paclk, P. 2002. On combining one-class classifiers for image database retrieval. In *Proceedings of the 3rd International Workshop on Multiple Classifier Systems*, 212–221, Italy.

Lee, W. & Liu, B. 2003. Learning with positive and unlabeled examples using weighted logistic regression. In *Proceedings of the 20th International Conference on Machine Learning*, Washington DC, USA.

Letouzey, F., Denis, F. & Gilleron, R. 2000. Learning from positive and unlabeled examples. In *Proceedings of 11th International Conference on Algorithmic Learning Theory*, Sydney, Australia.

Ling, C. X. & Sheng, V. S. 2010. Cost-sensitive learning. In *Encyclopedia of Machine Learning*, Sammut, C. & Webb, G. I. (eds). Springer, 231–235.

Liu, B., Dai, Y., Li, X., Lee, W. S. & Yu, P. S. 2003. Building text classifiers using positive and unlabeled examples. In *Proceedings of the 3rd IEEE International Conference on Data Mining*, Florida, USA.

Liu, B., Lee, W. S., Yu, P. S. & Li, X. 2002. Partially supervised classification of text documents. In *Proceedings of the 19th International Conference on Machine Learning*. Morgan Kaufmann Publishers Inc, Australia, 387–394.

Liu, B., Xiao, Y., Cao, L. & Yu, P. S. 2011. One-class based uncertain data stream learning. In *SIAM International Conference on Data Mining*, 992–1003, Arizona, USA.

Li, C. & Zhang, Y. 2008. Bagging one-class decision trees. In *Proceedings of 5th International Conference on Fuzzy Systems and Knowledge Discovery*, Shandong, 420–423.

Li, C., Zhang, Y. & Li, X. 2009. OcVFDT: one-class very fast decision tree for one-class classification of data streams. In *Proceedings of the 3rd International Workshop on Knowledge Discovery from Sensor Data*, Omitaomu, O. A., Ganguly, A. R., Gama, J., Vatsavai, R. R., Chawla, N. V. & Gaber, M. M. (eds). SensorKDD '09, ACM, 79–86. New York, NY, USA.

Li, K., Huang, H., Tian, S. & Xu, W. 2003. Improving one-class SVM for anomaly detection. In *Proceedings of the 2nd International Conference on Machine Learning and Cybernetics*, **5**, 3077–3081.

Li, W., Guo, Q. & Elkan, C. 2011. A positive and unlabeled learning algorithm for one-class classification of remote-sensing data. *IEEE Transactions on Geoscience and Remote Sensing* **49**(2), 717–725.

Li, X. & Liu, B. 2003. Learning to classify texts using positive and unlabeled data. In *Proceedings of 18th International Joint Conference on Artificial Intelligence*, 587–594, Mexico.

Luenberger, D. G. 1984. *Linear and Nonlinear Programming*, 2nd ed. Addison-Wesley.

Luo, J., Ding, L., Pan, Z., Ni, G. & Hu, G. 2007. Research on cost-sensitive learning in one-class anomaly detection algorithms. In *Autonomic and Trusted Computing*, Lecture Notes in Computer Science **4610**, 259–268. Springer.

Lyu, S. & Farid, H. 2004. Steganalysis using color wavelet statistics and one class support vector machines. In *Proceedings of SPIE* **5306**, 35–45, San Jose, USA.

Madden, M. G. & Howley, T. 2008. A machine learning application for classification of chemical spectra. In *Proceedings of 28th SGAI International Conference*, Cambridge, UK.

Manevitz, L. M. & Yousef, M. 2000a. Document classification on neural networks using only positive examples. In *Proceedings of 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 304–306, Athens, Greece.

Manevitz, L. M. & Yousef, M. 2000b. Learning from positive data for document classification using neural networks. In *Proceedings of 2nd Bar-Ilan Workshop on Knowledge Discovery and Learning*, Israel.

Manevitz, L. M. & Yousef, M. 2001. One-class SVMs for document classification. *Journal of Machine Learning Research* **2**, 139–154.

Markou, M. & Singh, S. 2003a. Novelty detection: a review – part 1: statistical approaches. *Signal Processing* **83**(12), 2481–2497.

Markou, M. & Singh, S. 2003b. Novelty detection: a review – part 2: neural networks based approaches. *Signal Processing* **83**(12), 2499–2521.

Mazhelis, O. 2006. One-class classifiers: a review and analysis of suitability in the context of mobile-masquerader detection. *South African Computer Journal (SACJ), ARIMA & SACJ Joint Special Issue on Advances in End-User Data-mining Techniques* **36**, 29–48.

Minter, T. C. 1975. Single-class classification. In *Symposium on Machine Processing of Remotely Sensed Data*. IEEE, 2A12–2A15.

Moya, M. R., Koch, M. W. & Hostetler, L. D. 1993. One-class classifier networks for target recognition applications. In *International Neural Network Society*, 797–801, Portland, OR.

Muggleton, S. 2001. Learning from the positive data. *Machine Learning*.

Munroe, D. T. & Madden, M. G. 2005. Multi-class and single-class classification approaches to vehicle model recognition from images. In *Proceedings of Irish Conference on Artificial Intelligence and Cognitive Science*, Portstewart.

Murshed, N., Bortolozzi, F. & Sabourin, R. 1996. Classification of cancerous cells based on the one-class problem approach. In *SPIE Conference on Applications and Science of Artificial Neural Networks II*, **2760**, 487–494, Orlando, USA.

Nanni, L. 2006. Experimental comparison of one-class classifiers for online signature verification. *Neurocomputing* **69**, 869–873.

Nguyen, B. V. 2002. An application of support vector machines to anomaly detection. Technical Report CS681, Ohio University.

Nguyen, H., Abdesselam Bouzerdoum, Giang. & Son, L. Phung 2009. Learning pattern classification tasks with imbalanced data sets. In *Pattern Recognition*, Peng-Yeng Yin (ed.). InTech.

Nguyen, M. N., Li, X. L. & Ng, S. K. 2011. Positive unlabeled learning for time series classification. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, 16–22, Spain.

Nigam, K., McCallum, A., Thrun, S. & Mitchell, T. 2000. Text classification from labeled and unlabeled documents using EM. *Machine Learning* **39**(2/3), 103–134.

Onoda, T., Murata, H. & Yamada, S. 2005. One class support vector machine based non-relevance feedback document retrieval. In *International Joint Conference on Neural Networks 2005*, Montreal, Canada.

Pan, R., Zhou, Y., Cao, B., Liu, N. N., Lukose, R., Scholz, M. & Yang, Q. 2008. One-class collaborative filtering. In *Proceedings of 8th IEEE International Conference on Data Mining*, Italy.

Pan, S., Zhang, Y., Li, X. & Wang, Y. 2010. Nearest neighbor algorithm for positive and unlabeled learning with uncertainty. *Journal of Computer Science and Frontiers* **4**(9), 766–779.

Pękalska, E., Skurichina, M. & Duin, R. P. W. 2004. Combining dissimilarity representations in one-class classifier problems. In *Proceedings Fifth International Workshop MCS 2004*, Springer, **3077**, 122–133.

Pękalska, E., Tax, D. M. J. & Duin, R. P. W. 2002. One-class LP classifiers for dissimilarity representations. In *Advances in Neural Info. Processing Systems*, Becker, S., Thrun, S. & Obermayer, K. (eds). **15**. MIT Press, 761–768, British Columbia, Canada.

Peng, T., Zuo, W. & He, F. 2006. Text classification from positive and unlabeled documents based on GA. In *Proceedings of VECPAR'06*, Brazil.

Perdisci, R., Gu, G. & Lee, W. 2006. Using an ensemble of one-class SVM classifiers to harden payload-based anomaly detection systems. In *Proceedings of the 16th International Conference on Data Mining*. IEEE Computer Society, 488–498.

Quinlan, J. M., Chalup, S. K. & Middleton, R. H. 2003. Application of SVMs for colour classification and collision detection with AIBO robots. *Advances in Neural Information Processing Systems* **16**, 635–642.

Quinlan, J. R. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann.

Rabaoui, A., Davy, M., Rossignol, S. & Ellouze, N. 2008. Using one-class SVMs and wavelets for audio surveillance systems. *IEEE Trans. on Information Forensic and Security* **3**(4), 763–775.

Rabaoui, A., Davy, M., Rossignol, S., Lachiri, Z. & Ellouze, N. 2007. Improved one-class SVM classifier for sounds classification. In *AVSS 2007. IEEE Conference on Advanced Video and Signal Based Surveillance*, 117–122, London.

Rätsch, G., Mika, S., Schölkopf, B. & Müller, K. R. 2002. Constructing boosting algorithms from SVMS: an approach to one-class classification. *IEEE Transaction on Pattern Analysis and Machine Intelligence* **24**(9), 1184–1199.

Ritter, G. & Gallegos, M. 1997. Outliers in statistical pattern recognition and an application to automatic chromosome classification. *Pattern Recognition Letters* **18**, 525–539.

Rocchio, J. 1971. Relevant feedback in information retrieval. In *In The Smart Retrieval System- experiments in automatic document processing*. Englewood Cliffs.

Rodriguez, B. M., Peterson, G. L. & Agaian, S. S. 2007. Steganography anomaly detection using simple one-class classification. In *Proceddings of the SPIE*, **6579**, page 65790E.

Rubin, D. B. 1987. *Multiple Imputation for Non response in Surveys*. John Wiley and Sons, New York.

Rumelhart, D. E. & McClelland, J. L. 1986. *Parallel distributed processing : Exploration in the microstructure of cognition*, volume 1 & 2. MIT Press.

Sachs, A., Thiel, C. & Schwenker, F. 2006. One-class support vector machines for the classification of bioacoustic time series. In *INFOS'06*, Cairo.

Sarmiento, T., Hong, S. J. & May, G. S. 2005. Fault detection in reactive ion etching systems using one-class, support vector machines. In *Advanced Semiconductor Manufacturing Conference and Workshop*, 139–142, Munich.

Schapire, R. E., Feund, Y., Bartlett, P. L. & Lee, W. S. 1998. Boosting the margin: A new explanation for the effectiveness of voting methods. *The Annals of Statistics* **26**(5), 1651–1686.

Schneider, K. M. 2004. Learning to filter junk e-mail from positive and unlabeled examples. In Lecture Notes in Computer Science, Su, K-Y., Tsujii, J., Lee, J-H. & Kwong, O. Y. (eds). **3248**, 426–435. Springer.

Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J. & Williamson, R. C. 1999b. Estimating the support of a high dimensional distribution. Technical Report MSR-TR-99-87, Microsoft Research.

Schölkopf, B., Smola, A. J. & Müller, K. R. 1998. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation* **10**, 1299–1319.

Schölkopf, B., Williamson, R. C., Smola, A. J., Taylor, J. S. & Platt, J.C. 2000. Support vector method for novelty detection. In *Neural Information Processing Systems*, 582–588, CO, USA.

Schölkopf, B., Williamson, R. C., Smola, A. J. & Taylor, J. S. 1999a. SV estimation of a distribution's support. In *Advances in Neural Information Processing Systems*, CO, USA.

Seguì, S., Igual, L. & Vitrià, J. 2010. Weighted bagging for graph based one-class classifiers. In *Multiple Classifier Systems*, Neamat Gayar, Josef Kittler, & Fabio Roli, (eds). Springer, **5997**, 1–10.

Seo, K. 2007. An application of one-class support vector machines in content based image retrieval. *Expert Systems with Applications* **33**(2), 491–498.

Sharkey, A. J. C. & Sharkey, N. E. 1995. How to improve the reliability of artificial neural networks. Technical Report CS-95-11, Department of Computer Science, University of Sheffield.

Shieh, A. D. & Kamm, D. F. 2009. Ensembles of one class support vector machines. In Lecture Notes in Computer Science, Benediktsson, J., Kittler, J. & Roli, F. (eds). **5519**, 181–190. Springer-Verlag.

Shin, H. J., Eom, D. W. & Kim, S. S. 2005. One-class support vector machines: an application in machine fault detection and classification. *Computers and Industrial Engineering* **48**(2), 395–408.

Silva, J. & Willett, R. 2009. Hypergraph-based anomaly detection of high-dimensional co-occurrences. *IEEE transactions on pattern analysis and machine intelligence* **31**(3), 563–569.

Skabar, A. 2003. Single-class classifier learning using neural networks: an application to the prediction of mineral deposits. In *Proceedings of the Second International Conference on Machine Learning and Cybernetics*, **4**, 2127–2132, China.

Spinosa, E. J. & Ferreira de Carvalho, A. C. P. L. 2004. SVMs for novel class detection in bioinformatics. In *Brazilian Workshop on Bioinformatics*, 81–88, Brazil.

Srebro, N. & Jaakkola, T. 2003. Weighted low-rank approximations. In *Proceedings of the 20th International Conference on Machine Learning*, Fawcett, T. & Mishra, N. (eds). AAAI Press, Washington DC, USA, 720–727.

Sun, D., Tran, Q. A., Duan, H. & Zhang, G. 2005. A novel method for Chinese spam detection based on one-class support vector machine. *Journal of Information and Computational Science* **2**(1), 109–114.

Tang, Y. & Yang, Z. 2005. One-class classifier for HFGWR ship detection using similarity-dissimilarity representation. In *Proceedings of the 18th International Conference on Innovations in Applied Artificial Intelligence*, Ali, M. & Esposito, F. (eds). Springer-Verlag, Italy, 432–441.

Tanigushi, M. & Tresp, V. 1997. Averaging regularized estimators. *Neural Computation* **9**, 1163–1178.

Tax, D. M. J. 2001. *One-class Classification*. PhD thesis, Delft University of Technology.

Tax, D. M. J. & Duin, R. P. W. 1999a. Data domain description using support vectors. In *Proceedings of European Sysmposium on Artificial Neural Networks*, Brussels, 251–256.

Tax, D. M. J. & Duin, R. P. W. 1999b. Support vector domain description. *Pattern Recognition Letters* **20**, 1191–1199.

Tax, D. M. J. & Duin, R. P. W. 2000. Data description in subspaces. In *Proceedings of 15th International Conference on Pattern Recognition*, Los Alamitos, 672–675.

Tax, D. M. J. & Duin, R. P. W. 2001a. Combining one class classifiers. In *Proceedings of the 2nd International Workshop on Multiple Classifier Systems*, 299–308, Cambridge, UK.

Tax, D. M. J. & Duin, R. P. W. 2001b. Uniform object generation for optimizing one-class classifiers. *Journal of Machine Learning Research* **2**, 155–173.

Tax, D. M. J. & Duin, R. P. W. 2004. Support vector data description. *Machine Learning* **54**(1), 45–66.

Tax, D. M. J., Ypma, A. & Duin, R. P. W. 1999. Support vector data description applied to machine vibration analysis. In *Proceedings of the 5th Annual Conference of the ASCI*, 398–405, The Netherlands.

Tian, J. & Gu, H. 2010. Anomaly detection combining one-class SVMs and particle swarm optimization algorithms. *Nonlinear Dynamics* **61**(1–2), 303–310.

Tran, Q. A., Duan, H. & Li, X. 2004. One-class support vector machine for anomaly network traffic detection. In *the 2nd Network Research Workshop of the 18th APAN*, Cairns, Australia.

Tu, Y., Li, G. & Dai, H. 2006. Integrating local one-class classifiers for image retrieval. In *Advanced Data Mining and Applications*, X. Li, O. R. Zaïane, & Z. H. Li (eds), Springer, **4093**, 213–222.

Valiant, L. G. 1984. A theory of learnable. *Coomunications of the ACM* **270**(11), 1134–1142.

Villalba, S. D. & Cunningham, P. 2007. An evaluation of dimension reduction techniques for one-class classification. *Artificial Intelligence Review* **270**(4), 273–294.

Wang, C., Ding, C., Meraz, R. F. & Holbrook, S. R. 2006. PSoL: a positive sample only learning algorithm for finding non-coding RNA genes. *BioInformatics* **22**(21), 2590–2596.

Wang, K. & Stolfo, S. J. 2003. One class training for masquerade detection. In *ICDM Workshop on Data Mining for Computer Security*.

Wang, Q. H., Lopes, L. S. & Tax, D. M. J. 2004. Visual object recognition through one-class learning. In *Image Analysis and Recognition,* Campilho, A. C. & Kamel, M. S. (eds). Lecture Notes in Computer Science **3211**, 463–470. Springer.

Wolpert, D. 1992. Stacked generalization. *Neural Networks* **5**, 241–259.

Wu, C. T., Cheng, K. T., Zhu, Q. & Wu, Y. L. 2005. Using visual features for anti spam filtering. In *Proceedings of IEEE International Conference on Image Processing*, **3**, 509–12, Italy.

Xiaomu, S., Guoliang, F. & Rao, M. 2008. SVM-Based data editing for enhanced one-class classification of remotely sensed imagery. *IEEE Geoscience and Remote Sensing Letters* **5**, 189–193.

Xu, Y. & Brereton, R. G. 2007. Automated single-nucleotide polymorphism analysis using fluorescence excitation–emission spectroscopy and one-class classifiers. *Analytical and Bioanalytical Chemistry* **388**(3), 655–664.

Yang, L. & Madden, M. G. 2007. One-class support vector machine calibration using particle swarm optimisation. In *AICS 2007*, Dublin.

Yang, J., Wang, S., Chen, N., Chen, X. & Shi, P. 2010a. Wearable accelerometer based extendable activity recognition system. In *2010 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 3641–3647, Alaska, USA.

Yang, J., Zhong, N., Liang, P., Wang, J., Yao, Y. & Lu, S. 2010b. Brain activation detection by neighborhood one-class SVM. *Cognitive Systems Research* **11**(1), 16–24.

Yasutoshi, Y. 2006. One-class support vector machines for recommendation tasks. In *PKDD*, Ng, W. K., Kitsuregawa, M., Li, J. & Chang, K. (eds). **3918**. Springer, Singapore, 230–239.

Yilmazel, O., Symonenko, S., Balasubramanian, N. & Liddy, E. D. 2005. Leveraging one-class SVM and semantic analysis to detect anomalous content. In *IEEE International Conference on Intelligence and Security Informatics*, Kantor, P. B., Muresan, G., Roberts, F. S., Zeng, D. D., Wang, F-Y., Chen, H. & Merkle, R. C. (eds). **3495**, Springer, 381–388.

Yousef, M., Jung, S., Showe, L.C. & Showe, M.K. 2008. Learning from positives examples when the negative class is undermined – microRNA gene identification. *Algorithms for Molecular Biology* **3**(2), 1–9.

Ypma, A. & Duin, R. P. W. 1998. Support objects for domain approximation. In *Proceedings of the 8th International Conference on Artificial Neural Networks*, Sweden.

Yu, H. 2003. SVMC: single-class classification with support vector machines. In *Proceedings of International Joint Conference on Artificial Intelligence*, 567–572, Mexico.

Yu, H. 2005. Single-class classification with mapping convergence. *Machine Learning* **610**(1), 49–69.

Yu, H., Han, J. & Chang, K. C. C. 2002. PEBL: positive-example based learning for web page classification using SVM. In *Eighth International Conference on Knowledge Discovery and Data Mining*, 239–248. Alberta, Canada.

Yu, H., Han, J. & Chang, K. C. C. 2004. PEBL: web page classification without negative examples. *IEEE Transactions on Knowledge and Data Engineering* **16**(1), 70–81.

Yu, H., Zhai, C. X. & Han, J. 2003. Text classification from positive and unlabeled documents. In *Proceedings of the 12th International Conference on Information and Knowledge Management*, 232–239. Louisiana, USA.

Zeng, Z., Fu, Y., Roisman, G. I., Wen, Z., Hu, Y. & Huang, T. S. 2006. One-class classification for spontaneous facial expression analysis. In *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, 281–286, Southampton, UK.

Zhang, B. & Zuo, W. 2008. Learning from positive and unlabeled examples: a survey. In *2008 International Symposiums on Information Processing ISIP*, 650–654, Russia.

Zhang, D., Cai, L., Wang, Y. & Zhang, L. 2010. A learning algorithm for one-class data stream classification based on ensemble classifier. In *2010 International Conference on Computer Application and System Modeling (ICCASM)*, IEEE, **2**. 596–600.

Zhang, J., Lu, J. & Zhang, G. 2011. Combining one class classification models for avian influenza outbreaks. In *2011 IEEE Symposium on Computational Intelligence in Multicriteria Decision-Making (MDCM)*, 190–196, Paris. IEEE.

Zhang, R., Zhang, S., Muthuraman, S. & Jiang, J. 2007. One class support vector machine for anomaly detection in the communication network performance data. In *Proceedings of the 5th Conference on Applied Electromagnetics, Wireless and Optical Communications*, Spain, 31–37.

Zhang, Y., Li, X. & Orlowska, M. 2008. One-class classification of text streams with concept drift. In *IEEE International Conference on Data Mining Workshops, 2008. ICDMW '08*. IEEE, 116–125, Italy.

Zhao, Y., Li, B., Li, X., Liu, W. & Ren, S. 2005. Customer churn prediction using improved one-class support vector machine. In *Advanced Data Mining and Applications*, Lecture Notes in Computer Science **3584**, 300–306.

Zhou, J., Chan, K. L., Chong, V. F. H. & Krishnan, S. M. 2005. Extraction of brain tumor from MR images using one-class support vector machine. In *Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, Shanghai, China, 1–4.

Zhu, X. 2005. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison.