How to Introduce Expert Feedback in One-Class Support Vector Machines for Anomaly Detection?[☆]

Julien Lesouple[a,*], Cédric Baudoin[b], Marc Spigai[b], Jean-Yves Tourneret[a,c]

[a] *TéSA laboratory, 7 boulevard de la Gare, 31500, Toulouse, France*
[b] *Thales Alenia Space, 26 Avenue Jean François Champollion, 31100, Toulouse, France*
[c] *INP-ENSEEIHT/IRIT, 2 rue Charles Camichel, 31071, Toulouse, France*

Abstract

Anomaly detection consists of detecting elements of a database that are different from the majority of normal data. The majority of anomaly detection algorithms considers unlabeled datasets. However, in some applications, labels associated with a subset of the database (coming for instance from expert feedback) are available providing useful information to design the anomaly detector. This paper studies a semi-supervised anomaly detector based on support vector machines, which takes the best of existing supervised and unsupervised support vector machines algorithms. The proposed algorithm allows the maximum proportion of vectors detected as anomalies and the maximum proportion of errors in the supervised data to be controlled, through two hyperparameters defining these proportions. Simulations conducted on various benchmark datasets show the interest of the proposed semi-supervised anomaly detection method.

Keywords: Machine learning, Semi-supervised learning, Anomaly detection, Support vector machines

1. Introduction

Machine learning (ML) methods have been gaining a huge interest with the computational abilities of modern computers, allowing a lot of data to be processed in a reasonable amount of time. Moreover, various kinds of data are becoming accessible due to various widespread sensors (Internet of things (IOT), mobile phones, satellites, medical devices, etc.) and growing storage capacities. ML algorithms for regression or classification classically use a so-called training set to build a model that is able to predict the outputs associated with other input vectors [1]. A well-known ML method is the support vector machine (SVM) classifier that has shown impressive results in many practical applications [2, Chap. 7], [1, Chap. 7].

[☆] This work was funded by TéSA and Thales Alenia Space.

[*] Corresponding author
Email address: julien.lesouple@tesa.prd.fr (Julien Lesouple)

This paper focuses on specific ML algorithms designed for anomaly detection (AD) [3, 4]. AD consists in detecting data that have not been generated by some normal process. AD techniques can generally handle unlabeled samples isolating some fraction of them that are classified as anomalies, the others being classified as normal data. These techniques are usually based on a cost function, which is estimated from the training dataset and a threshold (corresponding to a given false alarm rate) that needs to be adjusted by the user. Some popular AD techniques are based on the local outlier factor (LOF) [5] and the local outlier probability (LoOP) [6], which compute an anomaly score based on the nearest neighbors of each tested sample. Another popular algorithm is the isolation forest (IF) algorithm [7] evaluating the ability of an abnormal sample to be isolated from the majority of normal samples. The idea behind IF is that anomalies are generally far from normal data, which are gathered in some small subspace of the input space. Training samples are then isolated using random trees until obtaining subsets of cardinal 1 containing each vector of the database. An isolation score is finally computed for each vector of the database by counting the number of nodes required to reach the bottom of the tree for this vector. Methods based on SVMs have also been applied to AD leading to very efficient algorithms. These algorithms include the support vector data description (SVDD) algorithm [8], finding the smallest hypersphere containing a given fraction of the training data, and the one-class support vector machine (OCSVM) [9], finding the hyperplane separating the data from the origin with a maximum margin. Finally, an important class of anomaly detectors are those based on deep learning [10]. These detectors consider feature extraction, e.g., using a deep belief networks [11], learning of feature representations for normal data, e.g., using autoencoders [12], and end-to-end anomaly score learning, e.g., using one-class classification with two deep networks (a novelty detector and another detector enhancing the inlier samples and distorting the outliers) [13].

Anomaly detection can be obviously improved by using labeled data. However, obtaining labeled data representing all normal and abnormal behaviors, is often prohibitively expensive [3]. Therefore, some methods have been designed to process partially labeled datasets, leading to semi-supervised learning [14, 15, 16], with many applications to AD [17, 18, 19, 20]. In particular, extensions of SVMs have been proposed for semi-supervised learning [21], leading to transductive SVMs (TSVMs) [15, Chap. 6], semi-supervised SVMs (S3VMs) [22], and semi-supervised AD [23, 24, 25]. The principle of these methods is to determine the classifier with maximum margin using both labeled and unlabeled data vectors. The main drawback of these methods is that the standard convex SVM problem is transformed into a non-convex and NP-hard problem [16]. This problem can be solved by exploiting the fact that the majority of the data vectors are assumed to belong to the same class (containing normal vectors).

This paper studies a new semi-supervised algorithm for AD. This algorithm reduces to the state-of-the-art unsupervised SVM AD for unlabeled training data and to the supervised SVM classifier for fully labeled data. However, it can also be applied to partially labeled data leading to an interesting semi-supervised AD method. The proposed formulation is slightly different from the one introduced in [26], allowing the hyperparameters to be adjusted more easily. It has also some similarities with the approach of [25], except that it is more user-friendly since its hyperparameters have a physical meaning. Section 2 summarizes some important results on SVMs for supervised classification and for unsupervised and semi-supervised AD. Section 3 introduces the proposed AD method whose theoretical properties are detailed in Section 3.2. Section 4 evaluates the performance of the proposed AD method via simulations conducted on various datasets. Conclusions are reported in Section 5. Proofs of the various results can be found in the appendices.

2. State of the art

SVMs [2, Chap. 7], [1, Chap. 7] are powerful tools to determine data-driven decision functions for classification. This section recalls how SVMs can be used for supervised binary classification, unsupervised one-class classification, and semi-supervised AD.

2.1. Supervised SVM

Consider a labeled dataset $\{(\boldsymbol{x}_i, y_i)\}_{i=1,\dots,n}$ where $\boldsymbol{x}_i \in \mathbb{R}^d$ belongs to one of two classes with labels $y_i = \pm 1$. In the nonlinear setting, the idea of SVMs is to find a feature mapping $\Phi : \mathbb{R}^d \to \mathbb{R}^q$, with $q > d$, such that the transformed dataset $\Phi(\boldsymbol{X})$, with $\boldsymbol{X} = (\boldsymbol{x}_1, ..., \boldsymbol{x}_n)^T$, is linearly separable. This framework can be handled by solving the following optimization problem

$$\underset{\boldsymbol{w}\in\mathbb{R}^q, b\in\mathbb{R}}{\arg\min}\ \frac{1}{2}\|\boldsymbol{w}\|_2^2 \tag{1a}$$

$$\text{s.t. } y_i(\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) + b) \geq 1, \quad i = 1, \dots, n. \tag{1b}$$

In order to choose an appropriate mapping Φ ensuring linear separability of the two classes, according to the kernel trick, one can define a kernel k such that

$$\begin{aligned} k : \mathbb{R}^d \times \mathbb{R}^d &\to \mathbb{R} \\ (\boldsymbol{x}_i, \boldsymbol{x}_j) &\mapsto k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Phi(\boldsymbol{x}_i)^T\Phi(\boldsymbol{x}_j). \end{aligned} \tag{2}$$

It can be shown that the solution of problem (1) only depends on a subset of the training vectors, called support vectors, denoted as $\mathcal{S}_{\mathcal{V}}$. Namely,

$$\boldsymbol{w} = \sum_{\boldsymbol{x}_i \in \mathcal{S}_{\mathcal{V}}} \alpha_i y_i \Phi(\boldsymbol{x}_{\boldsymbol{i}}), \tag{3}$$

where $\alpha_i > 0$ for $\boldsymbol{x}_i \in \mathcal{S}_{\mathcal{V}}$. The decision function for a new vector $\boldsymbol{x}$ is then

$$f(\boldsymbol{x}) = \text{sign}\left(\boldsymbol{w}^T\Phi(\boldsymbol{x}) + b\right) \tag{4}$$

$$= \text{sign}\left(\sum_{\boldsymbol{x}_i \in \mathcal{S}_{\mathcal{V}}} \alpha_i y_i k(\boldsymbol{x}_{\boldsymbol{i}}, \boldsymbol{x}) + b\right). \tag{5}$$

There are many ways of constructing kernels for classification [2, Chap. 2.3]. This paper focuses on the well known Gaussian kernel defined as

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{1}{2\sigma^2}\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2\right), \tag{6}$$

which requires to adjust a unique hyperparameter $\sigma \in \mathbb{R}^+$. An advantage of using the Gaussian kernel is that many heuristics have been proposed to estimate the hyperparameter σ for binary classification, see for instance [27].

Problem (1) can be modified to allow some of the training data to violate the constraint (1b). This can be useful when dealing with mislabeled data, or to avoid overfitting. These methods are called soft margin SVMs [2, Chap. 7.5] and have led to C-SVM [28] and ν-SVM [29] classifiers. This paper concentrates on ν-SVM, because it has more interpretable parameters. However, under some specific assumptions, the two methods are known to be equivalent [2, Prop. 7.6]. The C-SVM method considers slack variables $\xi_i \geq 0$ associated with the learning data $\boldsymbol{x}_i$, allowing the constraint (1b) to be violated, i.e.,

$$y_i(\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) + b) \geq 1 - \xi_i, \quad 1 \leq i \leq n, \tag{7}$$

with $\xi_i \geq 0$. When the training vector $\boldsymbol{x}_i$ satisfies the constaint (1b), then $\xi_i = 0$. Conversely, when the constraint (7) is active, i.e., when $y_i(\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) + b) = 1 - \xi_i$ with $\xi_i > 0$, then the corresponding training vector is such that $y_i(\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) + b) < 1$, which allows constraint (1b) to be violated. A slight modification is generally introduced for constraint (7) leading to ν-SVM

$$y_i(\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) + b) \geq \rho - \xi_i, \quad 1 \leq i \leq n, \tag{8}$$

where $\rho \geq 0$. Note that instead of considering a canonical hyperplane (with margin equal to 1), a hyperplane with margin ρ will be preferred, in order to ensure specific properties for parameter ν defining the ν-SVM method described hereafter. To summarize, the C-SVM problem is defined as

$$\underset{\boldsymbol{w}\in\mathbb{R}^q, \boldsymbol{\xi}\in\mathbb{R}^n, b\in\mathbb{R}}{\arg\min} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{i=1}^{n}\xi_i \tag{9a}$$

$$\text{s.t } y_i(\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) + b) \geq 1 - \xi_i, \quad 1 \leq i \leq n \tag{9b}$$

$$\xi_i \geq 0, \quad 1 \leq i \leq n \tag{9c}$$

and the so-called ν-SVM problem is defined as

$$\underset{\boldsymbol{w}\in\mathbb{R}^q, \boldsymbol{\xi}\in\mathbb{R}^n, \rho, b\in\mathbb{R}}{\arg\min} \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \nu\rho + \frac{1}{n}\sum_{i=1}^{n}\xi_i \tag{10a}$$

$$\text{s.t } y_i(\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) + b) \geq \rho - \xi_i, \quad 1 \leq i \leq n \tag{10b}$$

$$\xi_i \geq 0, \quad 1 \leq i \leq n \tag{10c}$$

$$\rho \geq 0. \tag{10d}$$

It can be shown that parameter ν is an upper bound for the fraction of data points that violates the constraints (10b), i.e., the vectors such that $\xi_i > 0$, and a lower bound on the fraction of the number of support vectors, i.e., the data with $\alpha_i > 0$. For more details on soft-margin SVM, the interested readers are invited to consult [30, 31]. Note that on-the-shelf codes are available in the very popular `libsvm` package [32].

2.2. Unsupervised SVM

SVMs have been generalized to one-class SVMs for AD [9]. OCSVM considers n unlabeled training data $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ with a feature mapping Φ and determines the hyperplane separating the data from the origin located at a maximum distance of the origin. The use of slack variables $\xi_i \geq 0$ allows some points to be located in the wrong side of the hyperplane, leading to the following problem

$$\underset{\boldsymbol{w}\in\mathbb{R}^q, \boldsymbol{\xi}\in\mathbb{R}^n, \rho\in\mathbb{R}}{\arg\min} \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \rho + \frac{1}{\nu n}\sum_{i=1}^{n}\xi_i \tag{11a}$$

$$\text{s.t } \boldsymbol{w}^T\Phi(\boldsymbol{x}_i) \geq \rho - \xi_i, \quad 1 \leq i \leq n \tag{11b}$$

$$\xi_i \geq 0, \quad 1 \leq i \leq n. \tag{11c}$$

Note that contrary to Problem (10), no constraint is imposed to ρ, which might even be negative. When using a Gaussian kernel, the heuristics introduced in [33] can be considered to estimate the kernel hyperparameter in Eq. (6), i.e., estimating σ using the median of all pairwise distances between all vectors of the training set. It can be shown [9] that parameter ν is again an upper bound for the fraction of vectors that violate the constraints (11b), i.e., the vectors such that $\xi_i > 0$, and a lower bound for the fraction of support vectors, i.e., with $\alpha_i > 0$.

In the following, SVM models will be introduced to perform AD using partially labeled datasets.

2.3. Semi-Supervised Learning using SVM

This section considers a partially labeled dataset with two classes containing anomalies ($y_i = -1$) and normal data ($y_i = +1$). The dataset is split into two subsets associated with labeled and unlabeled training samples. Without loss of generality, the data are sorted such that the first r vectors correspond to labeled instances ($i = 1, \ldots, r$) whereas the $n-r$ last ones ($i = r+1, \ldots, n$) correspond to unlabeled instances. As claimed in the introduction, the proposed approach borrows some ideas from the S3VM Anomaly Detection (S3VMAD) [25], which is described hereafter. The S3VMAD algorithm is formulated[1] as

$$\begin{aligned}
&\underset{\boldsymbol{w}\in\mathbb{R}^q,\boldsymbol{\xi}\in\mathbb{R}^n,b}{\arg\min} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \frac{n-r}{C_0(n-r+1)}b + \frac{1}{C_2(r+1)}\sum_{i=1}^{r}\xi_i \\
&\qquad + \frac{1}{C_1(n-r+1)}\sum_{i=r+1}^{n}\xi_i && \text{(12a)}\\
&\text{s.t. } y_i(\boldsymbol{w}^T\Phi(\boldsymbol{x}_i)+b) \geq 1-\xi_i,\ \ i=1,\ldots,r && \text{(12b)}\\
&\qquad \boldsymbol{w}^T\Phi(\boldsymbol{x}_i)+b \geq -\xi_i,\ \ i=r+1,\ldots,n && \text{(12c)}\\
&\qquad \xi_i \geq 0,\ \ i=1,\ldots,n && \text{(12d)}
\end{aligned}$$

with $0 < C_i \leq 1$, $i = 0, ..., 2$. This formulation is a trade-off between C-SVM and OCSVM, which were proposed for labeled and unlabeled data respectively. Note that the hyperparameters C_i are not easily interpretable and can be difficult to adjust. Moreover, it enforces a margin equal to 1 for labeled data, and to 0 for unlabeled data.

3. Proposed algorithm

The proposed algorithm uses a ν-SVM approach to process labeled data (taking the best of the given labels) and an OCSVM approach for unlabeled training samples. The resulting approach is referred to as ν-SSVM in the following.

3.1. ν-SSVM formulation

The vector of labels $\boldsymbol{y} \in \mathbb{R}^r$ is extended to $\mathbb{R}^n$ by setting $y_i = 1$ for unlabeled data ($i = r+1, \ldots, n$). This will simplify the notations and yield more compact formulas. The proposed ν-SSVM strategy is formulated as follow

$$\begin{aligned}
&\underset{\boldsymbol{w}\in\mathbb{R}^q,\boldsymbol{\xi}\in\mathbb{R}^n,b,\rho_1,\rho_2\in\mathbb{R}}{\arg\min} \frac{1}{2}\|\boldsymbol{w}\|_2^2 - r\rho_1 - (n-r)(\rho_2-b) + \frac{1}{\nu_1}\sum_{i=1}^{r}\xi_i + \frac{1}{\nu_2}\sum_{i=r+1}^{n}\xi_i && \text{(13a)}\\
&\text{s.t. } y_i(\boldsymbol{w}^T\Phi(\boldsymbol{x}_i)+b) \geq \rho_1-\xi_i,\ \ i=1,\ldots,r && \text{(13b)}\\
&\qquad \boldsymbol{w}^T\Phi(\boldsymbol{x}_i)+b \geq \rho_2-\xi_i,\ \ i=r+1,\ldots,n && \text{(13c)}\\
&\qquad \xi_i \geq 0,\ \ i=1,\ldots,n && \text{(13d)}\\
&\qquad \rho_1 \geq 0 && \text{(13e)}\\
&\qquad \rho_2 \geq 0 && \text{(13f)}
\end{aligned}$$

where $0 < \nu_1 \leq 1$ and $0 < \nu_2 \leq 1$ are two parameters controlling the values of the slack variables, which can be tuned according to the application (the choice

[1] Note that ρ has been replaced by $-b$ from the original formulation to be in agreement with the C-SVM formulation.

of these parameters will be discussed later). As one can see, we propose to use two different margins ρ_1 and ρ_2 for labeled and unlabeled data, which will allow specific properties detailed in Sec. 3.2.3 to be satisfied (as shown at the end of Appendix D). By assigning the same value to ρ_1 and ρ_2, one would also perform AD but without satisfying these properties.

The idea of the proposed algorithm is to define a boundary between normal data and anomalies using the available labels and assuming that the majority of unlabeled data are normal. This problem consists of looking for a hyperplane in the feature space defined by $\boldsymbol{w}$ and b, with slack variables allowing some labeled data to be in the wrong side of the boundary, and some unlabeled data to be declared as anomalies. As shown in Appendix A, the dual problem of Problem (13) is

$$\begin{aligned}
&\arg\min_{\boldsymbol{\alpha}\in\mathbb{R}^n} \frac{1}{2}\boldsymbol{\alpha}^T \boldsymbol{Y}\boldsymbol{G}\boldsymbol{Y}\boldsymbol{\alpha} && \text{(14a)}\\
&\text{s.t.} \sum_{i=1}^{n} y_i\alpha_i = n-r && \text{(14b)}\\
&\sum_{i=r+1}^{n} \alpha_i \geq (n-r) && \text{(14c)}\\
&\sum_{i=1}^{r} \alpha_i \geq r && \text{(14d)}\\
&0 \leq \alpha_i \leq \frac{1}{\nu_1},\ \ i=1,\dots,r && \text{(14e)}\\
&0 \leq \alpha_i \leq \frac{1}{\nu_2},\ \ i=r+1,\dots,n && \text{(14f)}
\end{aligned}$$

where

- $\boldsymbol{\alpha} = (\alpha_1,\dots,\alpha_n) \in \mathbb{R}^n$ is the vector of Lagrange multipliers
- $\boldsymbol{Y} = \operatorname{diag}(y_i)_{1\leq i\leq n} \in \mathbb{R}^{n\times n}$ is the diagonal matrix of labels
- $\boldsymbol{G} = \boldsymbol{\Phi}\boldsymbol{\Phi}^T \in \mathbb{R}^{n\times n}$ is the Gram matrix of the problem
- $\boldsymbol{\Phi} = \Phi(\boldsymbol{X}) = \begin{bmatrix}\Phi(\boldsymbol{x}_1) & \dots & \Phi(\boldsymbol{x}_n)\end{bmatrix}^T \in \mathbb{R}^{n\times q}$ is the matrix gathering the training data mapped into the feature space.

This problem can be solved using quadratic programming. The values of b, ρ_1 and ρ_2 can be determined by considering the support vectors such that $0 < \alpha_i < \frac{1}{\nu_1}$ for $i = 1,\dots,r$, and $0 < \alpha_i < \frac{1}{\nu_2}$ for $i = r+1,\dots,n$. More precisely, b, ρ_1 and ρ_2 are obtained as the solution of the following linear system of equations

$$\begin{cases}
\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) = \rho_1 - b & \text{if } 1 \leq i \leq r \quad \text{and } y_i = +1\\
-\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) = \rho_1 + b & \text{if } 1 \leq i \leq r \quad \text{and } y_i = -1\\
\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) = \rho_2 - b & \text{if } r+1 \leq i \leq n
\end{cases} \tag{15}$$

which can be solved using the least squares method.

3.2. Properties of ν-SSVM

This section provides some properties of the ν-SSVM method, whose proofs follow the work of [31] and are reported in the appendices.

3.2.1. Cases in the limit

The ν-SSVM method reduces to ν-SVM when all the data are labeled and to OCSVM when all the data are unlabeled (see proofs in Appendix B).

3.2.2. Necessary conditions on ν_1 and ν_2

Problem (14) is feasible if and only if $0 < \nu_1 \leq \nu_{1,\max}$ and $0 < \nu_2 \leq 1$, where

$$\nu_{1,\max} = \frac{\min\left(\#\{i \leq r | y_i = +1\}, \#\{i \leq r | y_i = -1\}\right)}{r} + \frac{\#\{i \leq r | y_i = -1\}}{r} \tag{16}$$

where $\#\{i \leq r | y_i = +1\}$ (resp. $\#\{i \leq r | y_i = -1\}$) is the number of labeled samples satisfying $y_i = +1$ (resp. $y_i = -1$). The proof is given in Appendix C.

3.2.3. Interpretation of ν_1 and ν_2

The two parameters ν_1 and ν_2 are easy to interpret, which simplifies their choice:

- ν_1 is a lower bound for the fraction of support vectors among the labeled data. Moreover, if $\rho_1 > 0$, ν_1 is an upper bound for the fraction of labeled data that are on the wrong side of the boundary. Therefore, it can be interpreted as the trust behind the expert feedback or the tolerance with respect to the labels. This hyperparameter will be generally chosen to a small value (since we are confident with the expert feedback) and is bounded by $\nu_{1,\max}$ as explained before. In the experiments considered in this paper, this hyperparameter was fixed to $\nu_1 = 0.05$ (the proportion of errors in the labeled dataset is upper-bounded by 5%).

- ν_2 is a lower bound for the fraction of support vectors among the unlabeled data. Moreover, if $\rho_2 > 0$, ν_2 is an upper bound for the fraction of unlabeled data that are on the wrong side of the boundary. Therefore, this hyperparameter can be interpreted as the prior knowledge about the proportion of anomalies located in the training dataset. It was fixed to $\nu_2 = 0.1$ in all experiments (the maximum proportion of anomalies located in the unlabeled dataset is 10%).

Proofs of these properties are provided in Appendix D.

4. Experiments

This section evaluates the performance of the proposed method on 2D synthetic data and several benchmark datasets. Using synthetic data with controlled ground truth allows us to determine important performance measures such as the probability of anomaly detection and the probability of false alarm. It also allows the shape of the decision boundaries for the different detectors to be analyzed.

4.1. 2D synthetic data

To have visual information on the algorithms, ν-SSVM has been first implemented on the *toy2* dataset from [34] [2]. This dataset is composed of 485 vectors of $\mathbb{R}^2$ including 35 anomalies (corresponding to a probability of anomaly close to 7%). This dataset has been normalized in order to be zero-mean with a unit variance and is depicted in Fig. 1, where blue points correspond to normal data and red points to anomalies.

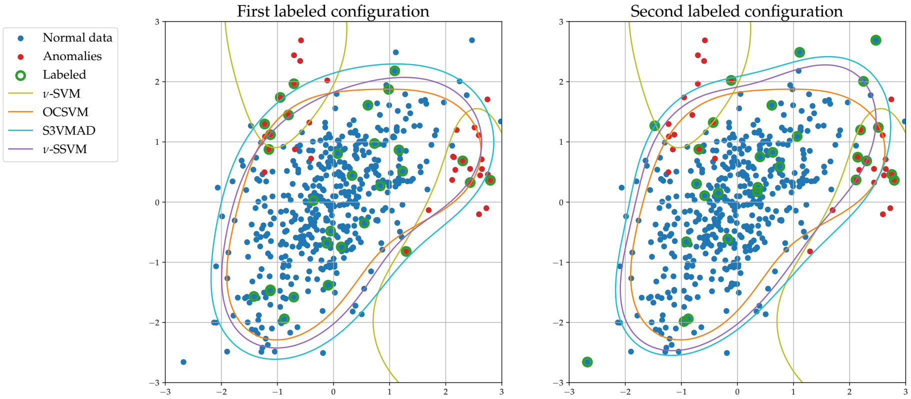

Figure 1: Illustration of the proposed ν-SSVM approach with two configurations of labeled vectors versus unsupervised OCSVM and supervised ν-SVM.

The proposed algorithm has been compared to several state-of-the-art methods that have been applied to the whole dataset. The different methods are summarized below:

- OCSVM with a Gaussian kernel whose parameter has been adjusted using the heuristic presented in [33]. The algorithm is unsupervised and was applied to the whole dataset (containing normal data and anomalies) with a maximal proportion of data lying outside the boundary fixed to $\nu_2 = 0.1$.

- ν-SVM with a Gaussian kernel. This algorithm is supervised and was applied to the labeled anomalies and normal instances. The kernel parameter was adjusted as in [33] using normal data only. The maximal proportion of data lying outside the separating boundary was fixed to $\nu_1 = 0.05$.

- S3VMAD is a reference for semi-supervised learning, which is known (see [25] for details) to outperform the semi-supervised AD (SSAD) method introduced in [17], the support vector data description with negative exemples (SVDD negative) [8] and the low density separation (LDS) method [35]. Looking carefully at (12a) and (13a), the hyperparameters of S3VMAD were chosen as follows: $C_0 = \frac{1}{n-r+1}, C_1 = \frac{\nu_2}{n-r+1}$ and $C_2 = \frac{\nu_1}{r+1}$.

[2] The dataset is available in the author webpage at `https://github.com/shubhomoydas/ad_examples/tree/master/ad_examples/datasets/anomaly`

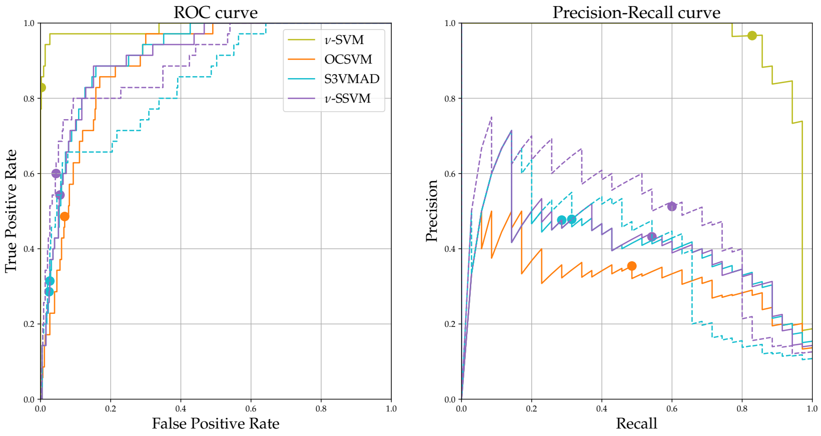

Figure 2: ROC (left) and Precision-Recall (right) curves of the various algorithms. Regarding S3VMAD and ν-SSVM, plain lines correspond to the first configuration of labeled data, and dashed lines to the second. For all curves, dots represent the results obtained with the thresholds given by the algorithms.

- The proposed ν-SSVM method was applied to the whole dataset with partially labeled data. More precisely, 20 normal data and 10 anomalies were randomly selected (circled in green in Fig.1) in order to build the labeled subset of data. The Gaussian kernel was used in the analysis, with the heuristic rule in [33] applied to all the data except labeled anomalies. The two hyperparameters of the algorithm were fixed to $\nu_1 = 0.05$ and $\nu_2 = 0.1$, as explained before.

- To emphasize the importance of labelling, S3VMAD and ν-SSVM were applied to a second subset of labeled vectors with the same values of ν_1 and ν_2.

The corresponding boundaries of the 3 algorithms are displayed in Fig. 1, showing that the use of labeled data changes the boundary of the OCSVM, in order to be in agreement with the given labels. Note that the influence of labeled data on the decision boundary can be observed in the two figures. For instance, labeled examples that are on the wrong side of the boundary for OCSVM are located on the good side (or on the boundary) for ν-SSVM. Moreover, in the second configuration, due to the labeling of the top right and bottom left normal data, the decision function for ν-SSVM is stretched around the axis defined by these two points. This clearly shows that if the number of labeled examples fed to the algorithm is limited, the vectors to be labeled should be optimized. These observations will be confirmed by the following statistical analysis. Note that ν-SVM seems to overfit and gather the anomalies in small clusters, i.e., it estimates the support of the anomalies rather than the support of normal data. In order to complement this analysis, the Receiver Operational Characteristic (ROC) and Precision-Recall (PR) curves are depicted in Fig.2

Table 1: 2D AD Algorithms.

Algorithm	Precision	Recall	F_1	ROC AUC	PR AUC
OCSVM	0.3541	0.4857	0.4096	0.8933	0.3353
ν-SVM	0.9666	0.8285	0.8923	0.9883	0.9542
S3VMAD - first configuration	**0.4761**	0.2857	0.3571	**0.9140**	0.4141
ν-SSVM - first configuration	0.4318	**0.5428**	**0.4810**	0.9101	**0.4167**
S3VMAD - second configuration	0.4782	0.3142	0.3793	0.8422	0.3977
ν-SSVM - second configuration	**0.5121**	**0.6**	**0.5526**	**0.8951**	**0.4993**

for all the algorithms. As one can see, for any configuration of labeled data, the ROC and PR curves for the proposed ν-SSVM are above those of S3VMAD.

To have more insight on the AD performance, classical metrics are derived for each method, namely the precision, the recall, the F_1 score, the area under the PR curve (PR-AUC) and the area under the ROC curve (ROC-AUC). Denoting as TP the number of true positives (an actual anomaly is detected, i.e., the true and estimated labels are -1), TN as the number of true negatives (a normal point is declared as normal, i.e., the true and estimated labels are equal to $+1$), FP as the number of false positives (a normal point is declared as anomaly, i.e., the true label is $+1$ and the estimated one is -1), and FN the number of false negatives (an actual anomaly is not detected, i.e., the true label is -1 and the estimated one is $+1$), the precision, recall and F_1 score can be defined as follows

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad F_1 = 2\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}.$$

The results shown in Table 1 allow us to make the following comments. The metrics obtained with ν-SSVM are far better than with OCSVM, even if the algorithm uses few labeled instances (30 labels out of 485 data, i.e., roughly 6% of the data). Note that this observation is valid for the two considered configurations. The ν-SVM algorithm, which can be considered as an ideal case where all the labels of the dataset are available, outperforms ν-SSVM. However, one has to keep in mind that ν-SVM requires a fully labeled dataset, which is rarely the case in AD applications. As claimed before, the way labeled examples are chosen has an impact on the decision function. At this point, it is interesting to mention that some methods have been designed to optimize the way data can be labeled [36, 37, 34, 26]. These methods are based on active learning, which requires queries to an oracle. In what follows, more complete datasets are considered, and the corresponding metrics are averaged over Monte-Carlo runs to have a better appreciation of the proposed method.

4.2. Benchmark datasets

This section evaluates the proposed ν-SSVM method using the unsupervised AD benchmark from Harvard dataverse [38], which contains several datasets with various numbers of features and various sample sizes. The data corresponds to two classes $+1$ and -1, with the class -1 highly unrepresented, and have been normalized in order to be zero-mean with a unit variance. All the

Table 2: Datasets used to evaluate the algorithms.

Name	Samples n	Features d	# +1	# -1
ANN Thyroid	6916	21	6666	250 (0.0361%)
Breast Cancer	367	30	357	10 (2.72%)
Letter	1600	32	1500	100 (6.25%)
Pen Global	809	16	719	90 (11.12%)
Satellite	5100	36	5025	75 (1.47%)
Speech	3686	400	3625	61 (1.65%)

datasets used in this paper along with their properties are gathered in Table 2. The proposed algorithm was tested with various proportions of labeled data, i.e., from 10% to 90% (the labeled data were selected uniformly in the full dataset). All the experiments were repeated 100 times to perform Monte Carlo simulations. The presented results are averaged and the corresponding standard deviations are computed using these 100 iterations. Note that for supervised (ν-SVM) and unsupervised (OCSVM) algorithms, there is no need to perform Monte Carlo simulations since only one configuration of labeled data is possible. The hyperparameters of the different algorithms were initially set to $\nu_1 = \min(0.05, 0.999\nu_0)$, where ν_0 is the maximum value for ν in ν-SVM, and $\nu_2 = 0.1$. Note that the value of $\nu_{1,\max}$ changes with respect to the number of labeled data and the considered configuration, cf. Eq. (16). When $\nu_1 \geq \nu_{1,\max}$, the hyperparameter ν_1 was set to $0.999\nu_{1,\max}$. For each SVM-based algorithm, a Gaussian kernel was used with a hyperparameter adjusted as for the 2D dataset.

4.2.1. Why should we introduce expert feedback into AD?

The first results illustrate the importance of introducing expert feedback in an AD algorithm. The average precision-recall curves with their confidence intervals (defined as plus and minus one standard deviation) for ν-SVM, OCSVM, S3VMAD, ν-SSVM show the impact of expert feedback into the unsupervised OCSVM algorithm. Note that ν-SVM requires that all the vectors from the dataset are labeled, providing an upper bound of performance. We recall here that S3VMAD is the state-of-the art in terms of semi-supervised AD and that ν-SSVM is the proposed approach. The corresponding curves have been computed for the datasets Breast Cancer, Letter and Pen Global and are displayed in Figures 3, 4 and 5[3]. As one can see, the curves obtained for S3VMAD and ν-SSVM are above those of OCSVM, showing the interest of introducing labeled data for AD. Moreover, the performance of the unsupervised methods tend to approach those of ν-SVM when the percentage of labeled data increases. Finally, it is interesting to note that the proposed ν-SSVM methods performs better than S3VMAD in all cases. The next section is dedicated to the qualitative evaluation of the proposed method.

[3] This paper only considers three datasets for space limitations. However, the conclusions would be the same for the other datasets from Harvard dataverse.

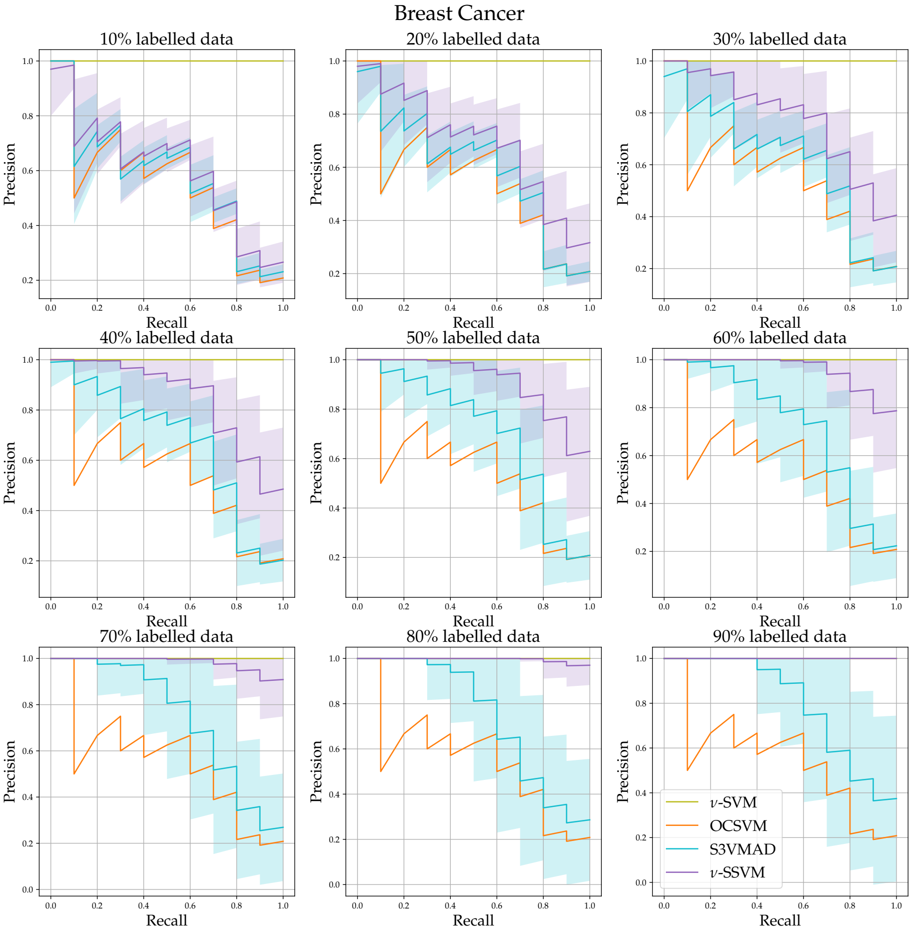

Figure 3: Mean PR curves for various percentages of labeled data - Breast Cancer dataset.

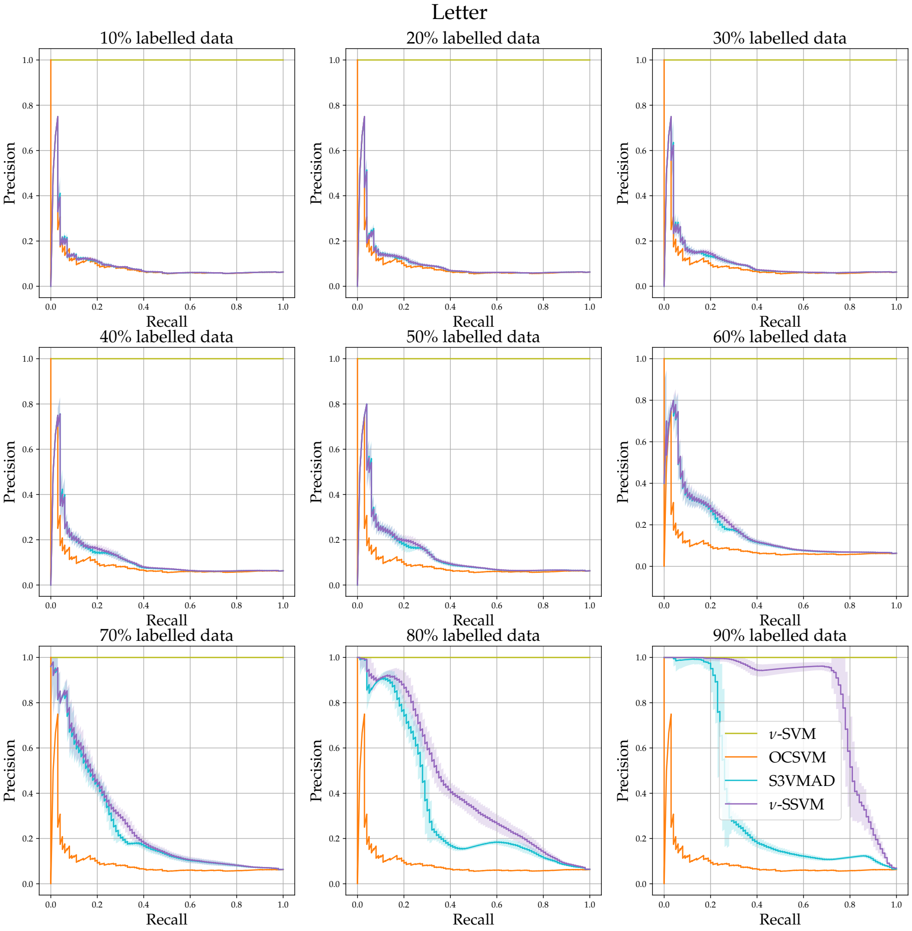

Figure 4: Mean PR curves for various percentages of labeled data - Letter dataset.

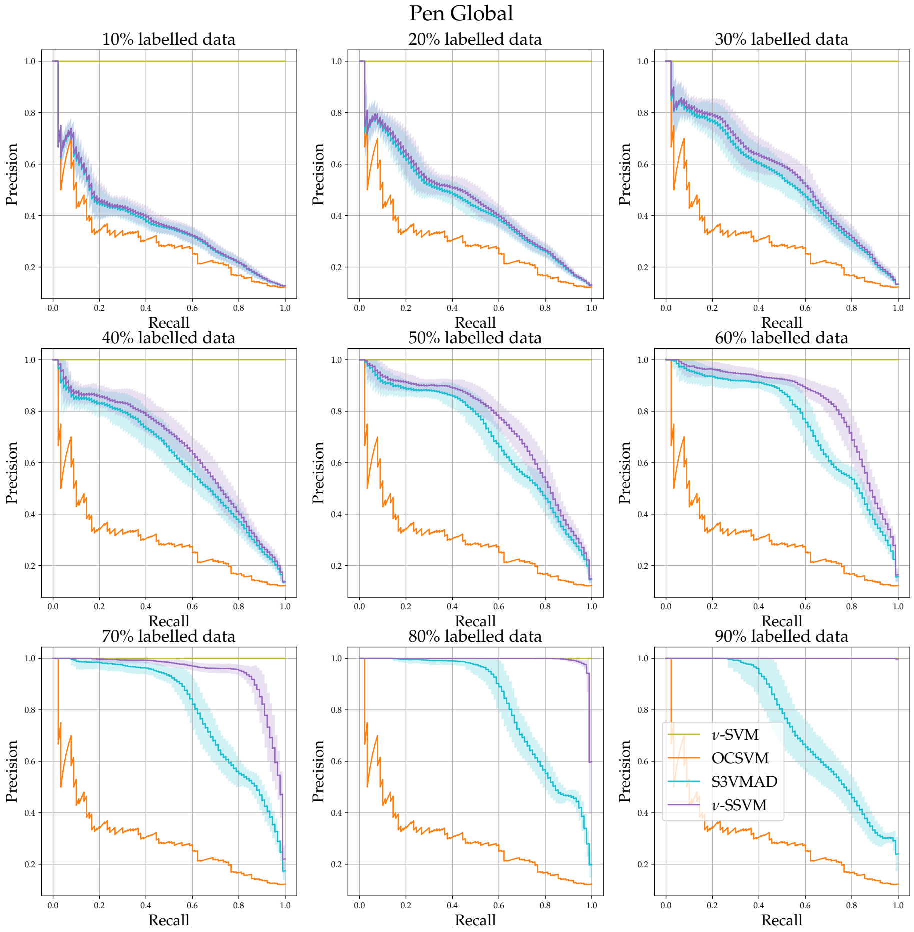

Figure 5: Mean PR curves for various percentages of labeled data - Pen Global dataset.

Table 3: Precision of S3VMAD, SSAD-IF and ν-SSVM for various datasets

	ANN Thyroid	Breast Cancer	Letter	Pen Global	Satellite	Speech
OCSVM	0.069	0.216	0.110	0.325	0.637	0.024
S3VMAD - 10%	0.160 ± 0.005	0.498 ± 0.054	0.136 ± 0.008	0.466 ± 0.044	0.708 ± 0.011	0.026 ± 0.004
SSAD-IF - 10%	**0.769** ± 0.068	0.529 ± 0.174	**0.339** ± 0.095	**0.963** ± 0.112	**0.977** ± 0.058	**0.085** ± 0.027
ν-SSVM - 10%	0.161 ± 0.057	**0.560** ± 0.194	0.125 ± 0.024	0.419 ± 0.037	0.628 ± 0.012	0.030 ± 0.017
S3VMAD - 20%	0.189 ± 0.006	0.571 ± 0.043	0.142 ± 0.010	0.559 ± 0.047	0.705 ± 0.010	0.032 ± 0.005
SSAD-IF - 20%	**0.813** ± 0.044	0.613 ± 0.090	**0.517** ± 0.095	**0.977** ± 0.058	**0.934** ± 0.106	**0.140** ± 0.033
ν-SSVM - 20%	0.170 ± 0.039	**0.734** ± 0.148	0.137 ± 0.010	0.500 ± 0.033	0.623 ± 0.009	0.034 ± 0.019
S3VMAD - 30%	0.213 ± 0.008	0.607 ± 0.059	0.154 ± 0.012	0.660 ± 0.054	0.722 ± 0.010	0.034 ± 0.005
SSAD-IF - 30%	**0.826** ± 0.036	0.643 ± 0.078	**0.646** ± 0.097	**0.995** ± 0.016	**0.836** ± 0.173	**0.184** ± 0.032
ν-SSVM - 30%	0.184 ± 0.016	**0.835** ± 0.126	0.151 ± 0.011	0.575 ± 0.039	0.636 ± 0.006	0.049 ± 0.028
S3VMAD - 40%	0.234 ± 0.013	0.668 ± 0.079	0.174 ± 0.013	0.760 ± 0.051	0.750 ± 0.008	0.031 ± 0.006
SSAD-IF - 40%	**0.829** ± 0.030	0.650 ± 0.074	**0.739** ± 0.070	**1.0** ± 0.000	**0.839** ± 0.178	**0.227** ± 0.031
ν-SSVM - 40%	0.212 ± 0.019	**0.923** ± 0.111	0.169 ± 0.013	0.637 ± 0.044	0.656 ± 0.006	0.057 ± 0.032
S3VMAD - 50%	0.243 ± 0.019	0.783 ± 0.101	0.198 ± 0.016	0.835 ± 0.040	0.790 ± 0.010	0.030 ± 0.003
SSAD-IF - 50%	**0.827** ± 0.041	0.676 ± 0.057	**0.796** ± 0.071	**0.999** ± 0.007	**0.799** ± 0.165	**0.274** ± 0.028
ν-SSVM - 50%	0.246 ± 0.027	**0.990** ± 0.031	0.200 ± 0.016	0.710 ± 0.048	0.677 ± 0.006	0.089 ± 0.045
S3VMAD - 60%	0.264 ± 0.021	0.865 ± 0.107	0.217 ± 0.028	0.880 ± 0.027	**0.832** ± 0.014	0.032 ± 0.004
SSAD-IF - 60%	**0.823** ± 0.034	0.691 ± 0.067	**0.838** ± 0.056	**1.0** ± 0.000	0.798 ± 0.186	**0.311** ± 0.031
ν-SSVM - 60%	0.347 ± 0.044	**0.997** ± 0.014	0.238 ± 0.019	0.800 ± 0.067	0.705 ± 0.008	0.148 ± 0.064
S3VMAD - 70%	0.329 ± 0.021	0.940 ± 0.075	0.228 ± 0.019	0.913 ± 0.024	**0.866** ± 0.010	0.037 ± 0.004
SSAD-IF - 70%	**0.815** ± 0.033	0.703 ± 0.065	**0.861** ± 0.060	**1.0** ± 0.000	0.773 ± 0.199	**0.343** ± 0.029
ν-SSVM - 70%	0.592 ± 0.022	**1.0** ± 0.000	0.296 ± 0.031	0.928 ± 0.026	0.735 ± 0.010	0.321 ± 0.132
S3VMAD - 80%	0.533 ± 0.034	**1.0** ± 0.000	0.253 ± 0.017	0.935 ± 0.065	**0.877** ± 0.016	0.057 ± 0.005
SSAD-IF - 80%	0.814 ± 0.032	0.715 ± 0.082	**0.881** ± 0.039	**1.0** ± 0.000	0.771 ± 0.188	0.379 ± 0.028
ν-SSVM - 80%	**0.837** ± 0.014	**1.0** ± 0.000	0.386 ± 0.054	0.986 ± 0.009	0.762 ± 0.014	**0.736** ± 0.129
S3VMAD - 90%	0.906 ± 0.048	**1.0** ± 0.000	0.276 ± 0.024	0.755 ± 0.091	**0.826** ± 0.032	0.043 ± 0.005
SSAD-IF - 90%	0.801 ± 0.041	0.729 ± 0.065	0.887 ± 0.041	**1.0** ± 0.000	0.769 ± 0.186	0.408 ± 0.025
ν-SSVM - 90%	**0.954** ± 0.009	**1.0** ± 0.000	**0.937** ± 0.021	0.999 ± 0.002	0.793 ± 0.022	**0.998** ± 0.007
ν-SVM	0.968	1.0	1.0	1.0	0.882	1.0

4.2.2. Quantitative evaluation

The proposed ν-SSVM algorithm was also compared quantitatively to a recent non SVM-based AD method based on isolation forests [34] (referred to as SSAD-IF). The various parameters for SSAD-IF have been tuned as advised in the original paper. The performance of the algorithms can be appreciated using the previous metrics (precision, recall, F_1, PR AUC and ROC AUC), which are reported in Tables 3, 4, 5, 6 and 7 respectively. Note that all the codes used to obtain the different results are available on the first author webpage[4]. Our conclusions are summarized below:

- For all metrics, the values of the performance measures obtained for semi-supervised algorithms vary from the values obtained for OCSVM (unlabeled data) to the values obtained with ν-SVM (fully labeled data), as expected.
- The performance of S3VMAD, SSAD-IF and ν-SSVM increases when using 10% of labeled data instead of 0%. This shows the benefit of using few labeled data and highlights the interest of incorporating expert feedback into an AD algorithm.

[4] `https://perso.tesa.prd.fr/jlesouple/codes.html`

Table 4: Recall of S3VMAD, SSAD-IF and ν-SSVM for various datasets

	ANN Thyroid	Breast Cancer	Letter	Pen Global	Satellite	Speech
OCSVM	0.192	0.8	0.18	0.3	0.201	0.098
S3VMAD - 10%	0.145 ± 0.002	0.654 ± 0.065	0.087 ± 0.012	0.188 ± 0.033	0.172 ± 0.008	0.045 ± 0.010
SSAD-IF - 10%	**0.643** ± 0.057	**0.688** ± 0.227	**0.166** ± 0.046	0.278 ± 0.032	0.093 ± 0.005	**0.156** ± 0.049
ν-SSVM - 10%	0.139 ± 0.025	0.637 ± 0.234	0.138 ± 0.059	**0.342** ± 0.097	**0.277** ± 0.008	0.031 ± 0.017
S3VMAD - 20%	0.149 ± 0.003	0.667 ± 0.067	0.098 ± 0.012	0.248 ± 0.048	0.220 ± 0.010	0.056 ± 0.011
SSAD-IF - 20%	**0.680** ± 0.037	**0.797** ± 0.117	**0.253** ± 0.046	0.282 ± 0.017	0.089 ± 0.010	**0.257** ± 0.062
ν-SSVM - 20%	0.157 ± 0.019	0.528 ± 0.233	0.136 ± 0.031	**0.435** ± 0.081	**0.340** ± 0.008	0.028 ± 0.013
S3VMAD - 30%	0.149 ± 0.004	0.660 ± 0.085	0.116 ± 0.014	0.305 ± 0.055	0.265 ± 0.012	0.063 ± 0.011
SSAD-IF - 30%	**0.690** ± 0.030	**0.836** ± 0.102	**0.316** ± 0.047	0.287 ± 0.004	0.080 ± 0.016	**0.338** ± 0.058
ν-SSVM - 30%	0.187 ± 0.017	0.519 ± 0.244	0.152 ± 0.033	**0.524** ± 0.079	**0.402** ± 0.009	0.031 ± 0.015
S3VMAD - 40%	0.147 ± 0.010	0.655 ± 0.121	0.141 ± 0.017	0.370 ± 0.067	0.314 ± 0.011	0.078 ± 0.015
SSAD-IF - 40%	**0.693** ± 0.025	**0.846** ± 0.097	**0.362** ± 0.034	0.288 ± 0.000	0.080 ± 0.017	**0.418** ± 0.057
ν-SSVM - 40%	0.212 ± 0.021	0.554 ± 0.203	0.168 ± 0.029	**0.606** ± 0.063	**0.463** ± 0.008	0.032 ± 0.013
S3VMAD - 50%	0.137 ± 0.012	0.650 ± 0.132	0.184 ± 0.023	0.443 ± 0.050	0.350 ± 0.011	0.102 ± 0.011
SSAD-IF - 50%	**0.691** ± 0.034	**0.879** ± 0.075	**0.390** ± 0.035	0.288 ± 0.002	0.076 ± 0.015	**0.503** ± 0.053
ν-SSVM - 50%	0.24 ± 0.021	0.563 ± 0.210	0.201 ± 0.032	**0.674** ± 0.059	**0.522** ± 0.009	0.044 ± 0.021
S3VMAD - 60%	0.124 ± 0.012	0.655 ± 0.154	0.229 ± 0.021	0.506 ± 0.048	0.391 ± 0.014	0.124 ± 0.018
SSAD-IF - 60%	**0.688** ± 0.028	**0.899** ± 0.087	**0.410** ± 0.027	0.288 ± 0.000	0.076 ± 0.017	**0.571** ± 0.057
ν-SSVM - 60%	0.311 ± 0.027	0.619 ± 0.188	0.250 ± 0.025	**0.746** ± 0.050	**0.586** ± 0.011	0.051 ± 0.024
S3VMAD - 70%	0.120 ± 0.013	0.653 ± 0.179	0.276 ± 0.018	0.553 ± 0.044	0.424 ± 0.014	0.158 ± 0.018
SSAD-IF - 70%	**0.682** ± 0.028	**0.915** ± 0.085	**0.421** ± 0.029	0.288 ± 0.000	0.074 ± 0.019	**0.630** ± 0.054
ν-SSVM - 70%	0.415 ± 0.038	0.674 ± 0.172	0.303 ± 0.032	**0.854** ± 0.032	**0.650** ± 0.011	0.062 ± 0.030
S3VMAD - 80%	0.119 ± 0.013	0.638 ± 0.178	0.316 ± 0.020	0.594 ± 0.049	0.444 ± 0.017	0.259 ± 0.027
SSAD-IF - 80%	**0.680** ± 0.027	**0.930** ± 0.107	0.431 ± 0.019	0.288 ± 0.000	0.073 ± 0.018	**0.696** ± 0.052
ν-SSVM - 80%	0.467 ± 0.031	0.723 ± 0.135	**0.448** ± 0.061	**0.947** ± 0.015	**0.711** ± 0.014	0.097 ± 0.035
S3VMAD - 90%	0.110 ± 0.019	0.632 ± 0.175	0.296 ± 0.028	0.509 ± 0.061	0.406 ± 0.024	0.155 ± 0.019
SSAD-IF - 90%	**0.669** ± 0.035	**0.948** ± 0.085	0.434 ± 0.020	0.288 ± 0.000	0.073 ± 0.017	**0.750** ± 0.046
ν-SSVM - 90%	0.484 ± 0.016	0.767 ± 0.120	**0.752** ± 0.026	**0.986** ± 0.014	**0.771** ± 0.021	0.246 ± 0.073
ν-SVM	0.5	0.8	0.98	1.0	0.920	0.573

Table 5: F_1 score of S3VMAD, SSAD-IF and ν-SSVM for various datasets

	ANN Thyroid	Breast Cancer	Letter	Pen Global	Satellite	Speech
OCSVM	0.102	0.340	0.136	0.312	0.306	0.039
S3VMAD - 10%	0.152 ± 0.002	0.563 ± 0.047	0.106 ± 0.010	0.266 ± 0.036	0.277 ± 0.010	0.033 ± 0.006
SSAD-IF - 10%	**0.700** ± 0.062	**0.598** ± 0.197	**0.223** ± 0.062	**0.431** ± 0.050	0.170 ± 0.010	**0.110** ± 0.035
ν-SSVM - 10%	0.141 ± 0.009	0.527 ± 0.147	0.122 ± 0.017	0.367 ± 0.066	**0.384** ± 0.008	0.028 ± 0.009
S3VMAD - 20%	0.167 ± 0.003	0.614 ± 0.042	0.115 ± 0.010	0.340 ± 0.046	0.336 ± 0.011	0.040 ± 0.007
SSAD-IF - 20%	**0.741** ± 0.040	**0.693** ± 0.102	**0.340** ± 0.062	0.438 ± 0.026	0.163 ± 0.018	**0.181** ± 0.043
ν-SSVM - 20%	0.159 ± 0.008	0.568 ± 0.181	0.134 ± 0.015	**0.460** ± 0.049	**0.440** ± 0.007	0.028 ± 0.009
S3VMAD - 30%	0.175 ± 0.005	0.630 ± 0.059	0.132 ± 0.011	0.415 ± 0.057	0.388 ± 0.013	0.044 ± 0.006
SSAD-IF - 30%	**0.752** ± 0.033	**0.726** ± 0.089	**0.424** ± 0.064	0.446 ± 0.007	0.146 ± 0.030	**0.238** ± 0.041
ν-SSVM - 30%	0.185 ± 0.012	0.596 ± 0.201	0.150 ± 0.018	**0.544** ± 0.046	**0.492** ± 0.007	0.034 ± 0.012
S3VMAD - 40%	0.181 ± 0.011	0.656 ± 0.088	0.155 ± 0.013	0.495 ± 0.067	0.442 ± 0.011	0.044 ± 0.007
SSAD-IF - 40%	**0.755** ± 0.027	**0.735** ± 0.084	**0.486** ± 0.046	0.448 ± 0.000	0.146 ± 0.031	**0.294** ± 0.040
ν-SSVM - 40%	0.211 ± 0.014	0.665 ± 0.169	0.167 ± 0.017	**0.618** ± 0.035	**0.543** ± 0.006	0.037 ± 0.012
S3VMAD - 50%	0.175 ± 0.015	0.700 ± 0.097	0.190 ± 0.018	0.577 ± 0.046	0.485 ± 0.010	0.046 ± 0.004
SSAD-IF - 50%	**0.753** ± 0.037	**0.764** ± 0.065	**0.524** ± 0.047	0.447 ± 0.001	0.139 ± 0.028	**0.355** ± 0.037
ν-SSVM - 50%	0.241 ± 0.016	0.692 ± 0.179	0.199 ± 0.019	**0.688** ± 0.029	**0.589** ± 0.005	0.051 ± 0.019
S3VMAD - 60%	0.169 ± 0.015	0.735 ± 0.125	0.222 ± 0.019	0.641 ± 0.042	0.532 ± 0.011	0.051 ± 0.006
SSAD-IF - 60%	**0.750** ± 0.031	**0.781** ± 0.076	**0.551** ± 0.036	0.448 ± 0.000	0.139 ± 0.032	**0.403** ± 0.040
ν-SSVM - 60%	0.326 ± 0.024	0.746 ± 0.159	0.243 ± 0.016	**0.769** ± 0.039	**0.640** ± 0.005	0.067 ± 0.026
S3VMAD - 70%	0.176 ± 0.016	0.755 ± 0.139	0.249 ± 0.014	0.688 ± 0.037	0.569 ± 0.011	0.060 ± 0.006
SSAD-IF - 70%	**0.743** ± 0.030	**0.795** ± 0.074	**0.566** ± 0.040	0.448 ± 0.000	0.135 ± 0.034	**0.444** ± 0.038
ν-SSVM - 70%	0.487 ± 0.026	0.791 ± 0.132	0.297 ± 0.015	**0.889** ± 0.021	**0.690** ± 0.003	0.091 ± 0.033
S3VMAD - 80%	0.195 ± 0.019	0.763 ± 0.142	0.280 ± 0.015	0.724 ± 0.043	0.589 ± 0.012	0.093 ± 0.009
SSAD-IF - 80%	**0.741** ± 0.029	0.808 ± 0.093	**0.579** ± 0.026	0.448 ± 0.000	0.134 ± 0.033	**0.491** ± 0.037
ν-SSVM - 80%	0.598 ± 0.026	**0.831** ± 0.096	0.408 ± 0.023	**0.966** ± 0.008	**0.735** ± 0.003	0.168 ± 0.050
S3VMAD - 90%	0.197 ± 0.031	0.759 ± 0.146	0.285 ± 0.023	0.604 ± 0.052	0.544 ± 0.021	0.068 ± 0.007
SSAD-IF - 90%	**0.729** ± 0.038	0.824 ± 0.074	0.583 ± 0.027	0.448 ± 0.000	0.134 ± 0.032	**0.529** ± 0.032
ν-SSVM - 90%	0.642 ± 0.015	**0.862** ± 0.079	**0.834** ± 0.016	**0.992** ± 0.007	**0.781** ± 0.002	0.390 ± 0.092
ν-SVM	0.659	0.888	0.989	1.0	0.901	0.729

Table 6: PR AUC of S3VMAD, SSAD-IF and ν-SSVM for various datasets

	ANN Thyroid	Breast Cancer	Letter	Pen Global	Satellite	Speech
OCSVM	0.079	0.577	0.095	0.308	0.450	0.018
S3VMAD - 10%	0.091 ± 0.002	0.600 ± 0.038	0.100 ± 0.001	0.374 ± 0.027	0.481 ± 0.005	0.018 ± 0.000
SSAD-IF - 10%	**0.759** ± 0.083	0.620 ± 0.250	**0.244** ± 0.058	**0.791** ± 0.085	**0.885** ± 0.048	**0.133** ± 0.038
ν-SSVM - 10%	0.091 ± 0.002	**0.625** ± 0.060	0.100 ± 0.001	0.381 ± 0.030	0.481 ± 0.005	0.018 ± 0.000
S3VMAD - 20%	0.103 ± 0.004	0.619 ± 0.054	0.106 ± 0.002	0.450 ± 0.032	0.518 ± 0.005	0.019 ± 0.000
SSAD-IF - 20%	**0.817** ± 0.048	**0.763** ± 0.163	**0.371** ± 0.063	**0.888** ± 0.070	**0.854** ± 0.098	**0.212** ± 0.051
ν-SSVM - 20%	0.104 ± 0.004	0.699 ± 0.079	0.106 ± 0.002	0.464 ± 0.032	0.519 ± 0.005	0.019 ± 0.000
S3VMAD - 30%	0.118 ± 0.006	0.644 ± 0.069	0.114 ± 0.003	0.534 ± 0.045	0.563 ± 0.007	0.020 ± 0.000
SSAD-IF - 30%	**0.838** ± 0.033	**0.837** ± 0.132	**0.482** ± 0.071	**0.948** ± 0.029	**0.770** ± 0.152	**0.283** ± 0.046
ν-SSVM - 30%	0.123 ± 0.006	0.787 ± 0.091	0.114 ± 0.003	0.559 ± 0.045	0.565 ± 0.007	0.020 ± 0.000
S3VMAD - 40%	0.139 ± 0.010	0.684 ± 0.082	0.125 ± 0.004	0.616 ± 0.048	0.619 ± 0.006	0.021 ± 0.000
SSAD-IF - 40%	**0.843** ± 0.027	0.850 ± 0.101	**0.575** ± 0.057	**0.971** ± 0.017	**0.770** ± 0.157	**0.353** ± 0.050
ν-SSVM - 40%	0.146 ± 0.011	**0.853** ± 0.102	0.127 ± 0.004	0.654 ± 0.045	0.622 ± 0.006	0.022 ± 0.002
S3VMAD - 50%	0.164 ± 0.008	0.718 ± 0.110	0.143 ± 0.005	0.694 ± 0.036	0.675 ± 0.004	0.022 ± 0.000
SSAD-IF - 50%	**0.843** ± 0.042	0.893 ± 0.067	**0.641** ± 0.063	**0.981** ± 0.011	**0.726** ± 0.150	**0.425** ± 0.043
ν-SSVM - 50%	0.183 ± 0.014	**0.916** ± 0.070	0.145 ± 0.005	0.744 ± 0.035	0.683 ± 0.004	0.025 ± 0.002
S3VMAD - 60%	0.188 ± 0.009	0.736 ± 0.138	0.176 ± 0.010	0.752 ± 0.031	0.722 ± 0.005	0.026 ± 0.001
SSAD-IF - 60%	**0.843** ± 0.026	0.892 ± 0.075	**0.703** ± 0.042	**0.986** ± 0.009	0.731 ± 0.160	**0.487** ± 0.045
ν-SSVM - 60%	0.255 ± 0.020	**0.960** ± 0.054	0.180 ± 0.010	0.828 ± 0.030	**0.738** ± 0.005	0.033 ± 0.005
S3VMAD - 70%	0.206 ± 0.012	0.751 ± 0.148	0.248 ± 0.014	0.798 ± 0.029	0.756 ± 0.004	0.035 ± 0.003
SSAD-IF - 70%	**0.839** ± 0.026	0.895 ± 0.075	**0.744** ± 0.061	**0.992** ± 0.004	0.719 ± 0.162	**0.538** ± 0.043
ν-SSVM - 70%	0.490 ± 0.027	**0.983** ± 0.032	0.262 ± 0.015	0.940 ± 0.017	**0.789** ± 0.003	0.058 ± 0.013
S3VMAD - 80%	0.212 ± 0.015	0.749 ± 0.151	0.345 ± 0.013	0.824 ± 0.027	0.768 ± 0.006	0.072 ± 0.009
SSAD-IF - 80%	**0.839** ± 0.023	0.899 ± 0.095	**0.787** ± 0.038	**0.993** ± 0.004	0.709 ± 0.163	**0.593** ± 0.041
ν-SSVM - 80%	0.764 ± 0.013	**0.995** ± 0.016	0.440 ± 0.023	0.993 ± 0.002	**0.838** ± 0.002	0.166 ± 0.020
S3VMAD - 90%	0.190 ± 0.021	0.804 ± 0.161	0.354 ± 0.018	0.738 ± 0.047	0.712 ± 0.012	0.180 ± 0.021
SSAD-IF - 90%	0.833 ± 0.033	0.900 ± 0.097	**0.822** ± 0.044	0.996 ± 0.002	0.714 ± 0.154	**0.633** ± 0.034
ν-SSVM - 90%	**0.931** ± 0.006	**1.0** ± 0.000	0.816 ± 0.019	**0.999** ± 0.000	**0.888** ± 0.002	0.560 ± 0.031
ν-SVM	0.944	1.0	1.0	0.999	0.975	0.984

Table 7: ROC AUC of S3VMAD, SSAD-IF and ν-SSVM for various datasets

	ANN Thyroid	Breast Cancer	Letter	Pen Global	Satellite	Speech
OCSVM	0.587	0.973	0.501	0.741	0.540	0.460
S3VMAD - 10%	0.619 ± 0.007	0.975 ± 0.003	0.511 ± 0.004	0.791 ± 0.019	0.589 ± 0.004	0.465 ± 0.002
SSAD-IF - 10%	**0.980** ± 0.023	0.967 ± 0.036	**0.736** ± 0.049	**0.954** ± 0.017	**0.937** ± 0.014	**0.616** ± 0.044
ν-SSVM - 10%	0.620 ± 0.007	**0.978** ± 0.006	0.512 ± 0.004	0.795 ± 0.020	0.590 ± 0.004	0.466 ± 0.001
S3VMAD - 20%	0.662 ± 0.011	0.973 ± 0.006	0.522 ± 0.005	0.832 ± 0.015	0.637 ± 0.005	0.471 ± 0.002
SSAD-IF - 20%	**0.990** ± 0.009	**0.984** ± 0.017	**0.814** ± 0.038	**0.977** ± 0.018	**0.932** ± 0.037	**0.689** ± 0.046
ν-SSVM - 20%	0.662 ± 0.012	0.981 ± 0.010	0.524 ± 0.005	0.838 ± 0.014	0.638 ± 0.005	0.472 ± 0.002
S3VMAD - 30%	0.717 ± 0.011	0.973 ± 0.009	0.535 ± 0.007	0.865 ± 0.018	0.683 ± 0.006	0.478 ± 0.004
SSAD-IF - 30%	**0.993** ± 0.002	**0.989** ± 0.012	**0.860** ± 0.033	**0.991** ± 0.007	**0.898** ± 0.077	**0.735** ± 0.036
ν-SSVM - 30%	0.721 ± 0.013	0.987 ± 0.012	0.538 ± 0.006	0.873 ± 0.017	0.685 ± 0.006	0.480 ± 0.002
S3VMAD - 40%	0.775 ± 0.014	0.970 ± 0.014	0.552 ± 0.008	0.896 ± 0.017	0.736 ± 0.007	0.488 ± 0.004
SSAD-IF - 40%	**0.994** ± 0.001	**0.991** ± 0.007	**0.896** ± 0.022	**0.995** ± 0.003	**0.897** ± 0.081	**0.780** ± 0.036
ν-SSVM - 40%	0.780 ± 0.017	0.988 ± 0.022	0.556 ± 0.008	0.908 ± 0.016	0.740 ± 0.007	0.490 ± 0.003
S3VMAD - 50%	0.817 ± 0.008	0.972 ± 0.014	0.578 ± 0.010	0.922 ± 0.012	0.787 ± 0.004	0.504 ± 0.006
SSAD-IF - 50%	**0.993** ± 0.007	0.994 ± 0.004	**0.923** ± 0.019	**0.997** ± 0.001	**0.878** ± 0.088	**0.831** ± 0.032
ν-SSVM - 50%	0.840 ± 0.013	**0.995** ± 0.006	0.582 ± 0.009	0.936 ± 0.011	0.793 ± 0.004	0.508 ± 0.004
S3VMAD - 60%	0.833 ± 0.009	0.971 ± 0.018	0.617 ± 0.013	0.939 ± 0.009	0.820 ± 0.004	0.526 ± 0.007
SSAD-IF - 60%	**0.994** ± 0.001	0.994 ± 0.005	**0.941** ± 0.011	**0.998** ± 0.001	**0.881** ± 0.088	**0.866** ± 0.028
ν-SSVM - 60%	0.890 ± 0.010	**0.997** ± 0.004	0.621 ± 0.014	0.958 ± 0.008	0.832 ± 0.003	0.531 ± 0.005
S3VMAD - 70%	0.835 ± 0.010	0.970 ± 0.020	0.686 ± 0.019	0.951 ± 0.008	0.843 ± 0.003	0.565 ± 0.011
SSAD-IF - 70%	**0.994** ± 0.001	0.995 ± 0.004	**0.954** ± 0.015	**0.999** ± 0.000	**0.877** ± 0.087	**0.902** ± 0.026
ν-SSVM - 70%	0.938 ± 0.007	**0.999** ± 0.003	0.694 ± 0.020	0.986 ± 0.004	0.868 ± 0.002	0.574 ± 0.006
S3VMAD - 80%	0.827 ± 0.013	0.966 ± 0.025	0.770 ± 0.011	0.957 ± 0.007	0.851 ± 0.003	0.640 ± 0.010
SSAD-IF - 80%	**0.994** ± 0.000	0.995 ± 0.006	**0.968** ± 0.010	**0.999** ± 0.000	0.869 ± 0.098	**0.935** ± 0.017
ν-SSVM - 80%	0.974 ± 0.004	**0.999** ± 0.001	0.820 ± 0.014	0.998 ± 0.001	**0.903** ± 0.001	0.658 ± 0.008
S3VMAD - 90%	0.743 ± 0.019	0.966 ± 0.033	0.759 ± 0.012	0.933 ± 0.014	0.795 ± 0.008	0.753 ± 0.012
SSAD-IF - 90%	**0.994** ± 0.003	0.996 ± 0.003	**0.980** ± 0.007	0.999 ± 0.000	0.876 ± 0.085	**0.963** ± 0.013
ν-SSVM - 90%	0.994 ± 0.001	**1.0** ± 0.000	0.933 ± 0.007	**0.999** ± 0.000	**0.936** ± 0.001	0.845 ± 0.012
ν-SVM	0.996	1.0	1.0	1.0	0.986	0.997

- The precision can be very low for small percentages of labeled data for SVM-based algorithms, see for instance the dataset speech with 50% of labeled data. This highlights the fact that the data to be labeled should be optimized.

- Globally, the proposed ν-SSVM seems to provide very competitive results when compared to S3VMAD, whereas the comparison between ν-SSVM and SSAD-IF depends on the dataset. Note that for small fractions of labeled data, the values obtained using ν-SSVM are not far from those of S3VMAD.

4.2.3. Discussion

The proposed ν-SSVM and SSAD-IF algorithms provide the best AD results for all datasets. However, ν-SSVM only requires the two hyperparameters ν_1 and ν_2 to be tuned. These two hyperparameters have a clear interpretation, which makes them easy to adjust. Conversely, SSAD-IF requires 6 hyperparameters to adjust: the forest size, the subsampling size, and the tree maximal height (to initialize the unsupervised IF), and three parameters to take into account the given labels (the fraction of data detected as anomalies to determine the threshold of the anomaly score, and two parameters in the cost function), which are less interpretable. The algorithm S3VMAD has also 3 hyperparameters that may be difficult to adjust in practical applications. Note that for SSAD-IF the threshold is determined using a fixed proportion of data located outside the boundary, whereas in the proposed approach this threshold is automatically provided by the algorithm to satisfy the constraints imposed by ν_1 and ν_2. In addition to that, SSAD-IF requires to project data into a high dimensional feature space (one component per node in the forest). Thus, we might have to deal with big vectors, whereas the proposed approach benefits from the kernel trick. To conclude this discussion, we think that ν-SSVM is an interesting algorithm for incorporating expert feedback into AD.

5. Conclusion

This paper studied a new SVM-based algorithm for anomaly detection using partially labeled datasets, with a given tolerance on the labels to avoid overfitting. The presented approach, called ν-SSVM, was shown to have interesting properties: 1) the proposed approach is equivalent to the OCSVM method when the whole dataset is unlabeled and to ν-SVM when the whole dataset is labeled, and 2) the user can control the tolerance on both the labeled and unlabeled data independently, thanks to two hyperparameters (denoted as ν_1 and ν_2 in the paper) that are easy to adjust and lead to smooth decision boundaries. Experiments conducted on both synthetic and real datasets allowed us to appreciate the performance of the proposed algorithm compared with OCSVM (unsupervised), ν-SVM (supervised), S3VMAD (semi-supervised) and SSAD-IF (semi-supervised). One interesting result is that few labels allow the results of OCSVM to be improved significantly. However, general conclusions on the optimal number of labeled data to consider could not be drawn because the results vary from one dataset to another.

Future works will consider applications of the proposed method to active learning. Active learning consists in finding efficient ways to partially label datasets, so as to minimize the number of requests to the so-called oracle. Some methods were already proposed using SVM [36, 26] or Isolation Forest [7, 34] and are compatible with the presented ν-SSVM. For instance, one could apply user feedback to Isolation Forest, thanks to the approach presented in [34] and control the tolerance on the labeled and unlabeled data using ν-SSVM.

Appendix A. Dual problem for ν-SSVM.

The Lagrangian of Problem (13) is

$$\begin{aligned}\mathcal{L}(\boldsymbol{w}, \boldsymbol{\xi}, b, \rho_1, \rho_2, \boldsymbol{\alpha}, \boldsymbol{\beta}, \delta) =& \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} - r\rho_1 - (n-r)(\rho_2 - b) + \frac{1}{\nu_1}\sum_{i=1}^{r}\xi_i + \frac{1}{\nu_2}\sum_{i=r+1}^{n}\xi_i \\ &- \sum_{i=1}^{r}\alpha_i\left(-\rho_1 + \xi_i + y_i\left(\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) + b\right)\right) \\ &- \sum_{i=r+1}^{n}\alpha_i\left(-\rho_2 + \xi_i + \boldsymbol{w}^T\Phi(\boldsymbol{x}_i) + b\right) \\ &- \sum_{i=1}^{n}\beta_i\xi_i - \delta\rho_1 - \gamma\rho_2 \qquad \text{(A.1)}\end{aligned}$$

where $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_n) \in \mathbb{R}^n$ contains the Lagrange multipliers associated with constraints (13b) (for $\alpha_1, \dots, \alpha_r$) and (13c) (for $\alpha_{r+1}, \dots, \alpha_n$), $\boldsymbol{\beta} = (\beta_1, \dots, \beta_n) \in \mathbb{R}^n$ is a vector whose components are the Lagrange multipliers for constraint (13d), $\delta \in \mathbb{R}$ is the Lagrange multiplier for (13e), and $\gamma \in \mathbb{R}$ is the Lagrange multiplier for constraint (13f). Setting to zero the derivatives of $\mathcal{L}$ with respect to the primal variables to zero, one obtains

$$\boldsymbol{w} = \sum_{i=1}^{n} y_i \alpha_i \Phi(\boldsymbol{x}_i) \tag{A.2}$$

$$\beta_i = \frac{1}{\nu_1} - \alpha_i, \quad , i = 1, \ldots, r \tag{A.3}$$

$$\beta_i = \frac{1}{\nu_2} - \alpha_i, \quad , i = r+1, \ldots, n \tag{A.4}$$

$$\sum_{i=1}^{n} y_i \alpha_i = n - r \tag{A.5}$$

$$\sum_{i=1}^{r} \alpha_i = r + \delta \tag{A.6}$$

$$\sum_{i=r+1}^{n} \alpha_i = n - r + \gamma. \tag{A.7}$$

Moreover, KKT conditions lead to

$$\alpha_i \geq 0 \tag{A.8}$$

$$\beta_i \geq 0 \tag{A.9}$$

$$\delta \geq 0 \tag{A.10}$$

$$\gamma \geq 0 \tag{A.11}$$

$$\alpha_i \left[-\rho_1 + \xi_i + y_i \left(\boldsymbol{w}^T \Phi(\boldsymbol{x}_i) + b \right) \right] = 0, \;\; i = 1, \ldots, r \tag{A.12}$$

$$\alpha_i \left(-\rho_2 + \xi_i + \boldsymbol{w}^T \Phi(\boldsymbol{x}_i) + b \right) = 0, \;\; i = r+1, \ldots, n \tag{A.13}$$

$$\beta_i \xi_i = 0 \tag{A.14}$$

$$\delta \rho_1 = 0 \tag{A.15}$$

$$\gamma \rho_2 = 0. \tag{A.16}$$

Using (A.3), (A.4), (A.8) and (A.9) leads to

$$0 \leq \alpha_i \leq \frac{1}{\nu_1}, \quad , i = 1, \ldots, r \tag{A.17}$$

$$0 \leq \alpha_i \leq \frac{1}{\nu_2}, \quad , i = r+1, \ldots, n \tag{A.18}$$

whereas (A.6) and (A.10) yield

$$\sum_{i=1}^{r} \alpha_i \geq r. \tag{A.19}$$

In addition, (A.7) and (A.11) yield

$$\sum_{i=r+1}^{n} \alpha_i \geq n - r. \tag{A.20}$$

Using the notation

$$\boldsymbol{Y} = \operatorname{diag}(y_i)_{1\leq i\leq n} \tag{A.21}$$

with (A.2) leads to the compact formula

$$\boldsymbol{w} = \boldsymbol{\Phi}^T \boldsymbol{Y}\boldsymbol{\alpha} \tag{A.22}$$

where $\boldsymbol{\Phi} = \Phi(\boldsymbol{X}) = \begin{bmatrix}\Phi(\boldsymbol{x}_1) & \dots & \Phi(\boldsymbol{x}_n)\end{bmatrix}^T \in \mathbb{R}^{n\times q}$. After replacing (A.22), (A.3), (A.4) (A.5) and (A.6) into the Lagrangian (A.1), the following result is obtained

$$\mathcal{L}(\boldsymbol{\alpha}) = -\frac{1}{2}\boldsymbol{\alpha}^T\boldsymbol{Y}\boldsymbol{G}\boldsymbol{Y}\boldsymbol{\alpha}. \tag{A.23}$$

As a consequence, the dual problem consists of maximizing the Lagrangian with respect to the Lagrange multipliers under constraints resulting from KKT conditions, i.e.,

$$\begin{aligned}
&\operatorname*{arg\,min}_{\boldsymbol{\alpha}\in\mathbb{R}^n} \frac{1}{2}\boldsymbol{\alpha}^T\boldsymbol{Y}\boldsymbol{G}\boldsymbol{Y}\boldsymbol{\alpha} && \text{(A.24a)}\\
&\text{s.t. } \sum_{i=1}^{n} y_i\alpha_i = n - r && \text{(A.24b)}\\
&\sum_{i=r+1}^{n} \alpha_i \geq n - r && \text{(A.24c)}\\
&\sum_{i=1}^{r} \alpha_i \geq r && \text{(A.24d)}\\
&0 \leq \alpha_i \leq \frac{1}{\nu_1},\ \ i = 1, \dots, r && \text{(A.24e)}\\
&0 \leq \alpha_i \leq \frac{1}{\nu_2},\ \ i = r+1, \dots, n. && \text{(A.24f)}
\end{aligned}$$

Appendix B. Cases in the limit for $\boldsymbol{\nu}$-SSVM.

If all the data are labeled, $r = n$, and (13) is equivalent to

$$\begin{aligned}
&\operatorname*{arg\,min}_{\boldsymbol{w}\in\mathbb{R}^q, \boldsymbol{\xi}\in\mathbb{R}^n, b, \rho_1\in\mathbb{R}} \frac{1}{2}\|\boldsymbol{w}\|_2^2 - n\rho_1 + \frac{1}{\nu_1}\sum_{i=1}^{n}\xi_i && \text{(B.1a)}\\
&\text{s.t. } y_i(\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) + b) \geq \rho_1 - \xi_i,\ \ i = 1, \dots, n && \text{(B.1b)}\\
&\xi_i \geq 0 && \text{(B.1c)}\\
&\rho_1 \geq 0. && \text{(B.1d)}
\end{aligned}$$

The problem does not change if the objective function is multiplied by $\frac{\nu_1^2}{n^2}$ and the constraints by $\frac{\nu_1}{n} > 0$. Thus, Problem (B.1) is equivalent to

$$\begin{aligned}
&\underset{\boldsymbol{w}\in\mathbb{R}^q,\boldsymbol{\xi}\in\mathbb{R}^n,b,\rho_1\in\mathbb{R}}{\arg\min} \frac{\nu_1^2}{2n^2}\|\boldsymbol{w}\|_2^2 - \frac{\nu_1^2}{n}\rho_1 + \frac{\nu_1}{n^2}\sum_{i=1}^n \xi_i && \text{(B.2a)}\\
&\text{s.t. } y_i\left(\frac{\nu_1}{n}\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) + \frac{\nu_1}{n}b\right) \geq \frac{\nu_1}{n}\rho_1 - \frac{\nu_1}{n}\xi_i && \text{(B.2b)}\\
&\frac{\nu_1}{n}\xi_i \geq 0 && \text{(B.2c)}\\
&\frac{\nu_1}{n}\rho_1 \geq 0. && \text{(B.2d)}
\end{aligned}$$

Introducing the notations $\boldsymbol{w'} = \frac{\nu_1}{n}\boldsymbol{w}, b' = \frac{\nu_1}{n}b, \rho' = \frac{\nu_1}{n}\rho_1$ and $\xi_i' = \frac{\nu_1}{n}\xi_i$, Problem (B.2) can be rewritten

$$\begin{aligned}
&\underset{\boldsymbol{w}'\in\mathbb{R}^q,\boldsymbol{\xi}'\in\mathbb{R}^n,b',\rho'\in\mathbb{R}}{\arg\min} \frac{1}{2}\|\boldsymbol{w}'\|_2^2 + \frac{1}{n}\sum_{i=1}^n \xi_i' - \nu_1\rho' && \text{(B.3a)}\\
&\text{s.t. } y_i\left(\boldsymbol{w}'^T\Phi(\boldsymbol{x}_i) + b'\right) \geq \rho' - {\xi_i}' && \text{(B.3b)}\\
&\xi_i' \geq 0 && \text{(B.3c)}\\
&\rho' \geq 0 && \text{(B.3d)}
\end{aligned}$$

which is exactly ν-SVM (10). Because ν_1 is strictly positive, the solution of (B.1) and (B.3) will lead to the same decision function, i.e., $\text{sign}(\boldsymbol{w}'^T\Phi(\boldsymbol{x})+b') = \text{sign}(\frac{\nu_1}{n}\boldsymbol{w}^T\Phi(\boldsymbol{x}) + \frac{\nu_1}{n}b) = \text{sign}(\boldsymbol{w}^T\Phi(\boldsymbol{x}) + b)$.

If all the data are unlabeled, $r = 0$, and (13) is equivalent to

$$\begin{aligned}
&\underset{\boldsymbol{w}\in\mathbb{R}^q,\boldsymbol{\xi}\in\mathbb{R}^n,b,\rho_2\in\mathbb{R}}{\arg\min} \frac{1}{2}\|\boldsymbol{w}\|_2^2 - n(\rho_2 - b) + \frac{1}{\nu_2}\sum_{i=1}^n \xi_i && \text{(B.4a)}\\
&\text{s.t. } \boldsymbol{w}^T\Phi(\boldsymbol{x}_i) \geq \rho_2 - b - \xi_i,\ \ i = 1,\dots,n && \text{(B.4b)}\\
&\xi_i \geq 0. && \text{(B.4c)}
\end{aligned}$$

Multiplying the cost function by $\frac{1}{n^2}$ and the constraints by $\frac{1}{n}$, one has

$$\begin{aligned}
&\underset{\boldsymbol{w}\in\mathbb{R}^q,\boldsymbol{\xi}\in\mathbb{R}^n,b,\rho_2\in\mathbb{R}}{\arg\min} \frac{1}{2n^2}\|\boldsymbol{w}\|_2^2 - \frac{1}{n}(\rho_2 - b) + \frac{1}{n^2\nu_2}\sum_{i=1}^n \xi_i && \text{(B.5a)}\\
&\text{s.t. } \frac{1}{n}\boldsymbol{w}^T\Phi(\boldsymbol{x}_i) \geq \frac{\rho_2 - b}{n} - \frac{1}{n}\xi_i,\ \ i = 1,\dots,n && \text{(B.5b)}\\
&\frac{1}{n}\xi_i \geq 0 && \text{(B.5c)}
\end{aligned}$$

Introducing the notations $\boldsymbol{w'} = \frac{1}{n}\boldsymbol{w}, b' = \frac{1}{n}b, \rho' = \frac{1}{n}(\rho_2 - b)$ and $\xi_i' = \frac{1}{n}\xi_i$, Problem (B.5) can be rewritten

$$\begin{aligned}
\underset{\boldsymbol{w}'\in\mathbb{R}^q,\boldsymbol{\xi}'\in\mathbb{R}^n,\rho'\in\mathbb{R}}{\arg\min}\; & \frac{1}{2}\|\boldsymbol{w}'\|_2^2 - \rho' + \frac{1}{n\nu_2}\sum_{i=1}^{n}\xi_i' && \text{(B.6a)}\\
\text{s.t. } & \boldsymbol{w}'^T\Phi(\boldsymbol{x}_i) \geq \rho' - \xi_i',\;\; i = 1,\dots,n && \text{(B.6b)}\\
& \xi_i' \geq 0 && \text{(B.6c)}
\end{aligned}$$

which is (11).

Appendix C. Feasibility of ν-SSVM

If Problem (14) has a solution, (14b) implies that it exists $\boldsymbol{\alpha} \in \mathbb{R}^n$ such that $\sum_{i=1}^n y_i\alpha_i = n - r \geq 0$. Moreover, the constraint (14c) implies $\sum_{i=r+1}^n y_i\alpha_i = \sum_{i=r+1}^n \alpha_i \geq n - r$ as $y_i = 1$ for unlabeled data. These two constraints imply

$$\sum_{i=1}^{r} y_i\alpha_i \leq 0, \text{ hence } \sum_{i\leq r, y_i=1}\alpha_i \leq \sum_{i\leq r, y_i=-1}\alpha_i. \tag{C.1}$$

Moreover, because of (14e), one has $\sum_{i\leq r,y_i=1}\alpha_i \leq \frac{\#\{i\leq r|y_i=+1\}}{\nu_1}$ and $\sum_{i\leq r,y_i=-1}\alpha_i \leq \frac{\#\{i|y_i=-1\}}{\nu_1}$. Using (C.1), these two quantities are both upper bounds of $\sum_{i\leq r,y_i=1}\alpha_i$, and therefore the tightest one is the minimum, i.e.,

$$\sum_{i\leq r,y_i=1}\alpha_i \leq \frac{\min\left(\#\{i\leq r|y_i=+1\}, \#\{i\leq r|y_i=-1\}\right)}{\nu_1}. \tag{C.2}$$

As a consequence

$$\begin{aligned}
\sum_{i=1}^{r}\alpha_i &= \sum_{i\leq r,y_i=1}\alpha_i + \sum_{i\leq r,y_i=-1}\alpha_i\\
&\leq \frac{\min\left(\#\{i\leq r|y_i=+1\}, \#\{i\leq r|y_i=-1\}\right)}{\nu_1} + \frac{\#\{i\leq r|y_i=-1\}}{\nu_1}.
\end{aligned} \tag{C.3}$$

Using (14d) and (C.3) leads to

$$\nu_1 \leq \frac{\min\left(\#\{i\leq r|y_i=+1\}, \#\{i\leq r|y_i=-1\}\right)}{r} + \frac{\#\{i\leq r|y_i=-1\}}{r} = \nu_{1,\max}. \tag{C.4}$$

The condition for ν_2 is obtained similarly. If Problem (14) admits a solution, because of (14c), it exists $\boldsymbol{\alpha} \in \mathbb{R}^n$ such that $\sum_{i=r+1}^n \alpha_i \geq n-r$, with $0 \leq \alpha_i \leq \frac{1}{\nu_2}$ due to (14f). Therefore

$$n - r \leq \sum_{i=r+1}^{n}\alpha_i \leq \frac{n-r}{\nu_2}, \text{hence} \quad 0 < \nu_2 \leq 1.$$

On the other hand, we assume $0 < \nu_1 \leq \nu_{1,\max}$ and $0 < \nu_2 \leq 1$. Denoting

$$k_y = \min\left(\#\{i \leq r | y_i = +1\}, \#\{i \leq r | y_i = -1\}\right) \tag{C.5}$$

and defining $\boldsymbol{\alpha}$ as follows

$$\alpha_j = \begin{cases} \dfrac{r}{k_y + \#\{i, y_i = -1\}} & \text{for the first } k_y \text{ components such as } y_j = 1 \\ 0 & \text{for the other components such that } y_j = 1 \\ \dfrac{r}{k_y + \#\{i, y_i = -1\}} & \text{if } y_j = -1 \end{cases} \quad , \quad j = 1, \ldots, r$$

$$\alpha_j = 1, \quad j = r+1, \ldots, n.$$

This value is well defined: if $k_y = \#\{i \leq r | y_i = +1\}$, there is no component equal to 0 in $\boldsymbol{\alpha}$, and if $k_y = \#\{i \leq r | y_i = -1\}$, the first k_y occurences of $\boldsymbol{\alpha}$ corresponding to $y_i = 1$ are non zero. The following result is obtained

$$\sum_{i=1}^{r} \alpha_i = \sum_{i, y_i = 1}^{r} \alpha_i + \sum_{i, y_i = -1}^{r} \alpha_i = r \frac{k_y}{k_y + \#\{i, y_i = -1\}} + r \frac{\#\{i, y_i = -1\}}{k_y + \#\{i, y_i = -1\}} = r$$

$$\begin{aligned} \sum_{i=1}^{r} y_i \alpha_i &= r \frac{k_y}{k_y + \#\{i, y_i = -1\}} - r \frac{\#\{i, y_i = -1\}}{k_y + \#\{i, y_i = -1\}} \\ &= \frac{r}{k_y + \#\{i, y_i = -1\}} \left(k_y - \#\{i, y_i = -1\}\right) \leq 0 \quad \text{(by definition of } k_y) \end{aligned}$$

$$\sum_{i=r+1}^{n} \alpha_i = n - r$$

Moreover, since $\nu_1 \leq \nu_{1,\max}$, we obtain

$$\frac{r}{k_y + \#\{i, y_i = -1\}} \leq \frac{1}{\nu_1}$$

hence

$$\alpha_i \leq \frac{1}{\nu_1}$$

and for $i = r+1, \ldots, n$

$$\nu_2 \leq 1 \Rightarrow \alpha_i \leq \frac{1}{\nu_2}. \tag{C.6}$$

Consequently, the vector $\boldsymbol{\alpha}$ satisfies all the constraints of Problem (14), which then admits at least one solution.

Appendix D. Properties of the $\boldsymbol{\nu}$-SSVM hyperparameters

This appendix derives some properties of ν_1 and ν_2 appearing in Eq. (13a). To prove the first property, we can consider (14d). The extreme case with the fewest support vectors corresponds to the case where all support vectors have their value equal to the maximum allowed, i.e., $\frac{1}{\nu_1}$, with a sum equal to r. If k is the minimum number of support vectors, in the extreme case one has $\frac{k}{\nu_1} = r$ thus $\nu_1 = \frac{k}{r}$, i.e., ν_1 is a lower bound on the fraction of support vectors for the labeled data.

Using the condition $\rho_1 > 0$ with the KKT condition (A.15) yields $\delta = 0$. Thus (A.6) leads to $\sum_{i=1}^{r} \alpha_i = r$. The labeled data that are in the wrong side of the boundary are those for which $\xi_i > 0$, i.e., using KKT condition (A.14) those for which $\beta_i = 0$. Equation (A.3) shows that these vectors satisfy the condition $\alpha_i = \frac{1}{\nu_1}$. In the extreme case where all the support vectors for labeled data are outside the boundary, these vectors are such that $\alpha_i = \frac{1}{\nu_1}$. Since their sum must be r, due to condition (A.6) with $\delta = 0$, there are $\nu_1 r$ of them among r labeled data. Therefore, ν_1 is an upper bound of the fraction of training data outside the boundary. The same reasoning can be applied to ν_2.

Note that if we would use a single parameter ρ instead of ρ_1 and ρ_2, conditions (A.6) and (A.7) would merge into a single condition

$$\sum_{i=1}^{n} \alpha_i = n + \delta. \tag{D.1}$$

If the previous reasoning is applied for $\delta = 0$, denoting as m_1 the number of misclassified labeled vectors and m_2 as the number of unlabeled data with predicted label -1, one has

$$\frac{m_1}{\nu_1} + \frac{m_2}{\nu_2} = n, \tag{D.2}$$

and we cannot conclude. The two parameters ρ_1 and ρ_2 are the theoretical distances from the support vectors (associated with $\xi_i = 0$) to the classification hyperplane, for labeled and unlabeled data respectively. They can be determined using Eq. (15) from the hyperparameters ν_1 and ν_2 (controlling the maximum proportions of data from the normal and abnormal classes located in the wrong side of the margin).

References

[1] C. M. Bishop, Pattern Recognition and Machine Learning, Springer-Verlag New York, 2006.

[2] B. Schölkopf, A. J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond, The MIT Press, Cambridge, MA, 2002.

[3] V. Chandola, A. Banerjee, V. Kumar, Anomaly Detection: A Survey, ACM Computing Surveys 41 (3) (2009) 15:1–15:58.

[4] M. A. Pimentel, D. A. Clifton, L. Clifton, L. Tarassenko, A Review of Novelty Detection, Signal Processing 99 (2014) 215–249.

[5] M. M. Breunig, H.-P. Kriegel, R. T. Ng, J. Sander, LOF: Identifying Density-Based Local Outliers, in: Proc. Int. Conf. on Management of Data (SIGMOD), Dallas, Tx, 2000, pp. 93–104.

[6] H.-P. Kriegel, P. Kröger, E. Schubert, A. Zimek, LoOP: Local Outlier Probabilities, in: Proc. Int. Conf. on Information and Knowledge Management (CIKM), Hong-Kong, China, 2009, pp. 1649–1652.

[7] F. T. Liu, K. M. Ting, Z.-H. Zhou, Isolation Forest, in: Proc. Int. Conf. on Data Mining (ICDM), Pisa, Italy, 2008, pp. 413–422.

[8] D. M. Tax, R. P. Duin, Support Vector Data Description, Machine Learning 54 (1) (2004) 45–66.

[9] B. Schölkopf, J. C. Platt, J. C. Shawe-Taylor, A. J. Smola, R. C. Williamson, Estimating the Support of a High-Dimensional Distribution, Neural Computation 13 (7) (2001) 1443–1471.

[10] G. Pang, C. Shen, L. Cao, A. van den Hengel, Deep Learning for Anomaly Detection: A Review, ACM Computing Survey 54 (2) (2021) 38:1–38:38.

[11] S. M. Erfani, S. Rajasegarar, S. Karunasekera, C. Leckie, High-Dimensional and Large-Scale Anomaly Detection Using a Linear One-Class SVM with Deep Learning, Pattern Recognition 58 (2016) 121–134.

[12] C. Zhou, R. C. Paffenroth, Anomaly Detection with Robust Deep Autoencoders, in: Proc. Int. Conf. on Knowledge Discovery and Data mining (KDD 17), Halifax, Canada, 2017, pp. 665–674.

[13] M. Sabokrou, M. Khalooei, M. Fathy, E. Adeli, Adversarially learned one-class classifier for novelty detection, in: Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 3379–3388.

[14] X. Zhu, Semi-Supervised Learning Literature Survey, Tech. Rep. Computer Sciences TR 1530, University of Wisconsin-Madison (July 2008).

[15] O. Chapelle, B. Schölkopf, A. Zien, Semi-Supervised Learning, The MIT Press, Cambridge, MA, 2006.

[16] J. E. van Engelen, H. H. Hoos, A Survey on Semi-Supervised Learning, Machine Learning 109 (2) (2020) 373–440.

[17] N. Görnitz, M. Kloft, K. Rieck, U. Brefeld, Toward Supervised Anomaly Detection, Journal of Artificial Intelligence Research 46 (2013) 235–262.

[18] H. Song, Z. Jiang, A. Men, B. Yang, A Hybrid Semi-Supervised Anomaly Detection Model for High-Dimensional Data, Computational Intelligence and Neuroscience 2017.

[19] V. Vercruyssen, W. Meert, G. Verbruggen, K. Maes, R. Bäumer, J. Davis, Semi-Supervised Anomaly Detection with an Application to Water Analytics, in: Proc. IEEE Int. Conf. on Data Mining, Singapore, 2018, pp. 527–536.

[20] L. Bergman, Y. Hoshen, Classification-based anomaly detection for general data, in: International Conference on Learning Representations (ICLR), 2020.

[21] S. Ding, Z. Zhibin, X. Zhang, An Overview on Semi-Supervised Support Vector Machine, Neural Computing and Applications 28 (5) (2017) 969–978.

[22] K. P. Bennett, A. Demiriz, Semi-Supervised Support Vector Machines, in: Proc. Advances in Neural Information Processing Systems 11 (NIPS 1998), Denver, CO, 1998, pp. 368–374.

[23] B. Liu, Y. Dai, X. Li, W. sun Lee, P. S. Yu, Building Text Classifiers Using Positive and Unlabeled Examples, in: Proc. IEEE Int. Conf. Data Mining, Melbourne, FL, 2003, pp. 19–22.

[24] F. Bovolo, L. Bruzzone, M. Marconcini, A Novel Approach to Unsupervised Change Detection Based on a Semisupervised SVM and a Similarity Measure, IEEE Trans. Geosci. and Remote Sensing 46 (7) (2008) 2070–2082.

[25] J. Kim, P. Montague, An Efficient Semi-Supervised SVM for Anomaly Detection, in: Proc. Int. Joint Conf. on Neural Networks, Anchorage, AK, 2017, pp. 2843–2850.

[26] J. Lesouple, J.-Y. Tourneret, Incorporating User Feedback Into One-Class Support Vector Machines for Anomaly Detection, in: Proc. Eur. Conf. on Signal Processing (EUSIPCO), Amsterdam, Netherlands, 2020, pp. 1608–1612.

[27] W. Wang, Z. Xu, W. Lu, X. Zhang, Determination of the Spread Parameter in the Gaussian Kernel for Classification and Regression, Neurocomputing 55 (3–4) (2003) 643–663.

[28] C. Cortes, V. Vapnik, Support-Vector Networks, Machine Learning 20 (1995) 273–297.

[29] B. Schölkopf, A. J. Smola, R. C. Williamson, P. L. Bartlett, New Support Vector Algorithms, Neural Computation 12 (5) (2000) 1207–1245.

[30] P.-H. Chen, C.-J. Lin, B. Schölkopf, A Tutorial on ν-Support Vector Machines, Applied Stochastic Models in Business and Industry 21 (2) (2005) 111–136.

[31] C.-C. Chang, C.-J. Lin, Training ν-Support Vector Classifiers: Theory and Algorithms, Neural Computation 13 (9) (2001) 2219–2147.

[32] C.-C. Chang, C.-J. Lin, LIBSVM: A library for support vector machines, ACM Transactions on Intelligent Systems and Technology 2 (2011) 27:1–27:27, software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[33] T. Jaakkola, M. Diekhans, D. Haussler, Using the Fisher Kernel Method to Detect Remote Protein Homologies, in: Proc. Int. Conf. on Intelligent Systems for Molecular Biology, Heidelberg, Germany, 1999, pp. 149–158.

[34] S. Das, W.-K. Wong, A. Fern, T. G. Dietterich, M. A. Siddiqui, Incorporating Feedback into Tree-based Anomaly Detection, in: Proc. Workshop on Interactive Data Exploration and Analytics (IDEA), Halifax, Canada, 2017, pp. 25–33.

[35] O. Chapelle, A. Zien, Semi-Supervised Classification by Low Density Separation, in: Proc. Workshop on Artificial Intelligence and Statistics, Barbados, 2005, pp. 57–64.

[36] D. Cohn, L. Atlas, R. Ladner, Improving Generalization with Active Learning, Machine Learning 15 (2) (1994) 201–221.

[37] S. Das, W.-K. Wong, T. G. Dietterich, A. Fern, A. Emmott, Incorporating Expert Feedback into Active Anomaly Discovery, in: Proc. Int. Conf. on Data Mining (ICDM), Barcelona, Spain, 2016, pp. 853–858.

[38] M. Goldstein, Unsupervised Anomaly Detection Benchmark (2015). `doi:10.7910/DVN/OPQMVF`.
URL `https://doi.org/10.7910/DVN/OPQMVF`