

Detecting Abnormal Ship Trajectories using Functional Isolation Forests and Dynamic Time Warping

Valérien MANGÉ
Hensoldt Nexeya France¹
TéSA Laboratory²
Toulouse, France

¹valerian.mange@hensoldt.fr

²valerian.mange@tesa.prd.fr

Yoann ANEZIN
TéSA Laboratory
Toulouse, France
yoann.anezin@tesa.prd.fr

Jean-Yves TOURNERET
University of Toulouse, ENSEEIHT-IRIT³
TéSA Laboratory²
Toulouse, France

³jean-yves.tourneret@toulouse-inp.fr

François VINCENT
ISAE-Supaéro
Toulouse, France
francois.vincent@isae-supaero.fr

Laurent MIRAMBELL
Hensoldt Nexeya France
Toulouse, France
laurent.mirambell@hensoldt.fr

Fábio MANZONI VIEIRA
Hensoldt Nexeya France
Toulouse, France
fabio.manzoni@hensoldt.fr

Abstract—This paper studies an algorithm allowing the isolation forest method to be adapted to time series associated with ship trajectories. This algorithm builds decision trees using different similarity measures between the ship trajectories of interest and the atoms of a dictionary constructed by the user. The similarity measure used to compare trajectories with potentially different lengths is based on dynamic time warping. Results obtained on synthetic data with an available ground truth yield promising results, when compared to the state-of-the-art.

Index Terms—Maritime surveillance, Anomaly detection, Dynamic Time Warping, Isolation Forest

I. INTRODUCTION

Context. Oceans contain a wide variety of resources while also serving as an inexpensive means to transport goods across the globe. High priority sectors (such as protected or high traffic zones) must be monitored and the behaviors of ships analyzed efficiently so as to take action as quickly as possible in case of illicit actions. Some recent methods allowing abnormal trajectories to be detected are based on deep learning [1], [2], clustering algorithms able to detect outliers such as DBSCAN [3] and conformal anomaly detection [4]. Different sensors exist to track the movement of ships, such as AIS (Automatic Identification System), radar and SAR (Synthetic Aperture Radar) images, which can be combined. The detection of anomalies from these data, even after association, remains an important research topic, especially because ship trajectories are time series with potentially different lengths.

State-of-the-art. Several anomaly detection (AD) algorithms have been proposed in the literature [5] among which Isolation Forest (IF) [6] remains a standard and very efficient strategy. However, IF requires training and test vectors to have the same

dimension. This can be a problem when the measurements are time series with different lengths such as ship trajectories. A state-of-the-art algorithm for detecting anomalies in time series of different lengths has been proposed by Laxhammar [4]. This algorithm uses conformal AD with the Hausdorff distance and will be one of two methods used for comparison in this paper as it contains a reference dataset. The other method is DBSCAN combined with the longest common subsequence (LCSS) similarity measure from [3]. An adaptation of the One-Class SVM method for the detection of abnormal ship trajectories has been recently proposed in [7]. IF has also been considered for AD but less intensively. The existing IF-based methods detecting abnormal ship trajectories generally consider a set of features extracted from these trajectories and possibly combine existing AD algorithms with clustering methods [8]. To the best of the authors' knowledge, the only recent paper dealing with IF for time series is [9].

Objectives, contributions and organization. This paper studies a new IF method to detect anomalies in time series built from ship trajectories. The paper is organized as follows: Section II recalls the principles of the IF algorithm. Section III explains how to adapt IF to time series having potentially different lengths, using functional IF and dynamic time warping (DTW). Section IV is devoted to experiments comparing different AD methods for ship trajectories. Conclusions are reported in Section V.

II. ANOMALY DETECTION USING ISOLATION FOREST

AD consists of identifying data that does not conform to a normal behavior [5]. In reality, there is no precise definition of an anomaly and the assumption often made in this field is that anomalies are rare. This makes model training complicated because the two classes are unbalanced (there are fewer

anomalies than normal data) and this training must be done in an unsupervised way because it is difficult, in principle, to identify all potential anomalies. This paper focuses on IF that has received a considerable attention in the literature [5], [9]. This section recalls the principles of IF [6] and one of its extensions based on functional isolation forest (FIF) [9].

A. Isolation Forest

Consider a set of vectors $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ with $N \geq 1$ and $\mathbf{x}_i \in \mathbb{R}^d, i = 1, \dots, N$ with d the number of features. The goal of IF is to build trees isolating each vector \mathbf{x}_i from the database. The more ‘‘anomalous’’ the vector \mathbf{x}_i , the easier or faster it can be isolated from the other vectors. Building multiple trees leads to a forest of trees. To build a binary tree for a subsample $\mathcal{X}_\psi \subset \mathcal{X}$ (with sub-sample size $\psi > 2$), a feature is selected randomly. The maximum and minimum of this feature are computed before randomly selecting a split for this attribute (located between the minimum and the maximum of the feature). In this step, all vectors with a feature higher than the split go into the right branch of the node whereas the others go to the left branch. This operation is repeated until all vectors have either been isolated or have reached a maximum depth fixed by the user. After constructing a forest of trees, an anomaly score $s(\mathbf{x}_i, \psi)$ is defined for each vector \mathbf{x}_i [6]:

$$s(\mathbf{x}_i, \psi) = 2^{-\frac{E[h(\mathbf{x}_i)]}{c(\psi)}}, \quad (1)$$

where $c(\psi) = 2 \left[\ln(\psi - 1) + \gamma - \frac{\psi - 1}{N} \right]$, γ is the Euler constant and $E[h(\mathbf{x}_i)]$ is the average tree path length of \mathbf{x}_i computed using all the trees of the forest. As such, the number of trees and the maximum depth of a tree are both parameters that have to be adjusted by the user. It is worth noting that an interesting property of IF is that it can be used for both continuous and discrete features.

B. Functional Isolation Forest

An extension of IF referred to as ‘‘Functional Isolation Forest’’ (FIF) was proposed for time series [9]¹. The motivation of FIF was to detect different abnormal behaviors in time series by computing similarity scores between the vectors \mathbf{x}_i and a fixed set of atoms belonging to a dictionary $\mathcal{D} = \{\mathbf{d}_1, \dots, \mathbf{d}_{N_D}\}$, with $\mathbf{d}_k \in \mathbb{R}^d$. More precisely, for a given node of the tree, instead of choosing a random feature in a time series \mathbf{x}_i , a random element \mathbf{d}_k is selected from \mathcal{D} and all the vectors \mathbf{x}_i such that $\langle \mathbf{x}_i, \mathbf{d}_k \rangle$ is larger (resp. smaller) than a threshold (where $\langle \cdot \rangle$ is a scalar product chosen by the user) are going to the left (resp. right) branch of the node.

Note that different choices of dictionaries were proposed in [9] including the self dictionary containing the data to be analyzed, the cosine dictionary composed of cosine functions with different frequencies and the mexican hat wavelet dictionary made of negative second derivatives of the normal density. In this work, we consider a self dictionary $\mathcal{D}_{\text{self}}$ since it seems complicated to choose bases adapted to all normal

ship trajectories. All the scalar products between the atoms \mathbf{d}_k of the dictionary that correspond to the trajectories of the dataset are computed to form a similarity matrix \mathbf{S} of size $N \times N$. IF is finally applied to the columns of \mathbf{S} .

C. Principal Component Analysis

As explained in [6], IF can suffer from the curse of dimensionality. This paper considers the possibility of reducing the dimension of the similarity matrix \mathbf{S} while keeping as much information as possible. Principal Component Analysis (PCA) is a common method that can be used for dimensionality reduction. PCA can be used here to project the columns of the similarity matrix \mathbf{S} into a lower dimensional subspace spanned by the eigenvectors of the covariance matrix of \mathbf{S} associated with its largest eigenvalues $\lambda_1, \dots, \lambda_q$. The number of largest eigenvalues of the matrix \mathbf{S} has been selected in order to obtain an appropriate value of the variance ratio $r = \sum_{k=1}^q \lambda_k / \sum_{k=1}^d \lambda_k$.

III. DYNAMIC TIME WARPING

A. Principle

Even if IF and FIF are powerful AD algorithms, they should be applied to time series having the same length. To compare time series with different lengths, an appropriate similarity metric needs to be defined. This paper proposes to use dynamic time warping (DTW), which can be used to build a correspondence or ‘‘warping’’ between two time series to evaluate their proximity (see Figs. 1 and 2 for illustration). DTW has been used in several applications such as trajectory outlier detection [10]. Its principle is recalled in this section.

Consider two multivariate time series concatenated in matrices \mathbf{T} and \mathbf{U} (potentially of different dimensions) describing the time evolution of features associated with the trajectories such as latitude, longitude, velocity, heading, etc... Denote as $\mathbf{T} = (\mathbf{t}_1, \dots, \mathbf{t}_n), n \geq 1$ and $\mathbf{U} = (\mathbf{u}_1, \dots, \mathbf{u}_m), m \geq 1$, where \mathbf{t}_i and \mathbf{u}_j contain the features of the i th and j th columns of the $d \times n$ and $d \times m$ matrices \mathbf{T} and \mathbf{U} . The DTW cumulated similarity score between the first i columns of \mathbf{T} and the first j columns of \mathbf{U} , denoted as $s(i, j)$, is defined by:

$$s(i, j) = \min\{s(i-1, j-1), s(i, j-1), s(i-1, j)\} + \|\mathbf{t}_i - \mathbf{u}_j\|^2, \quad (2)$$

with $1 \leq i \leq n, 1 \leq j \leq m$ and with the constraint that the first and last elements of both time series are paired together. The value of $s(n, m)$ is denoted as $\text{DTW}(\mathbf{T}, \mathbf{U})$. Figs. 1 and 2 illustrate the difference of point associations using the Euclidean norm and DTW.

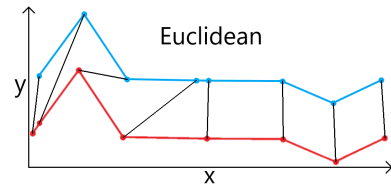


Fig. 1. Euclidean norm between two trajectories sampled differently.

¹The authors of this paper would like to thank Guillaume Staerman for interesting discussions regarding functional isolation forest.

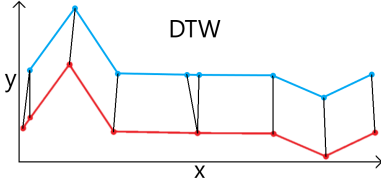


Fig. 2. DTW between two trajectories sampled differently.

B. Similarity Measures based on DTW

This paper studies a modification of FIF that can handle trajectories of different lengths using similarity scores based on DTW. More precisely, similarity measures can be defined for trajectories in the original spaces \mathbb{R}^n and \mathbb{R}^m and in higher dimensionality spaces defined by reproducing kernels.

1) *Similarity measures in the original data space:* Three similarity measures based on DTW denoted as DTW, sim_{DTW} and cos_{DTW} are considered. DTW was introduced in the previous section whereas sim_{DTW} and cos_{DTW} are defined as

$$\begin{aligned} \text{sim}_{\text{DTW}}(\mathbf{T}, \mathbf{U}) &= \frac{\text{DTW}^2(\mathbf{T}, -\mathbf{U}) - \text{DTW}^2(\mathbf{T}, \mathbf{U})}{4}, \\ \text{cos}_{\text{DTW}}(\mathbf{T}, \mathbf{U}) &= \frac{\text{sim}_{\text{DTW}}(\mathbf{T}, \mathbf{U})}{\|\mathbf{T}\| \times \|\mathbf{U}\|}. \end{aligned} \quad (3)$$

The smaller these similarity measures, the closer the corresponding trajectories \mathbf{T} and \mathbf{U} .

2) *Similarity measures in an implicit feature space:* Initially used for support vector machines (SVMs), reproducing kernels are endowed with a scalar product in a vector space of higher dimension (see [11] for a background on reproducing kernel Hilbert spaces). A classic kernel is the Gaussian kernel (GK) that is used to define the following similarity measure:

$$\kappa(\mathbf{T}, \mathbf{U}) = \exp \left[-\frac{\text{DTW}^2(\mathbf{T}, \mathbf{U})}{2\sigma^2} \right], \quad (4)$$

where $\sigma > 0$ is a bandwidth parameter chosen by the user. It can be shown that the GK corresponds to a scalar product in a space of infinite dimension. The scores defined by DTW and (3) can be used to evaluate the similarity between a ship trajectory and an atom of the dictionary. It should be noted that these similarity scores can be used with other distance measures. As such, the previous notations are extended to sim_{dist} and cos_{dist} with “dist” a distance measure.

C. Bandwidth selection for the Gaussian kernel

The hyperparameter σ should be chosen in order to optimize AD. A strategy studied in [12] consists of maximizing the following cost function J :

$$J(\sigma) = \frac{2}{N} \sum_{i=1}^N \left[\exp \left(-\frac{C(\mathbf{X}_i)}{2\sigma^2} \right) - \exp \left(-\frac{F(\mathbf{X}_i)}{2\sigma^2} \right) \right], \quad (5)$$

where F and C are defined as:

$$\begin{aligned} F(\mathbf{X}_i) &= \max_{1 \leq j \leq N} \text{DTW}^2(\mathbf{X}_i, \mathbf{X}_j), \\ C(\mathbf{X}_i) &= \min_{1 \leq j \leq N, i \neq j} \text{DTW}^2(\mathbf{X}_i, \mathbf{X}_j), \end{aligned}$$

with \mathbf{X}_i the i th multivariate time series with $i = 1, \dots, N$. The maximum of J can be found by using gradient descent methods. In this paper, the initial point of this gradient descent is calculated using Aggarwal’s trick [13, p. 83-88]. It consists in calculating as a first estimate of σ the median of all pairwise distances between the trajectories. Trinh and al. proposed to estimate σ as the value maximizing J , when there is no anomaly in the dataset. Since we use this method with a dataset potentially containing anomalies, instead of building F as the maximum of DTW over all vectors of the database, we propose to define F as the $1 - \nu$ percentile of the DTW values to filter the anomalies (with $\nu = 0.04$ in our experiments).

D. Anomaly detection

Once a similarity measure (such as DTW, $\text{sim}_{\text{DTW}}(\mathbf{T}, \mathbf{U})$, $\text{cos}_{\text{DTW}}(\mathbf{T}, \mathbf{U})$ or $\kappa(\mathbf{T}, \mathbf{U})$) has been defined, an isolation forest can be constructed to isolate each vector $\mathbf{x}_i \in \mathcal{X}$. An anomaly score is then be computed for any vector $\mathbf{x}_i \in \mathcal{X}$ using (1). Anomaly detection is finally performed by comparing this anomaly score to an appropriate threshold. One possibility is to set this threshold to 0.5, i.e., to detect as abnormal the vectors that are more likely to be anomalies. Indeed, uncertain data will have an average path length close to $c(\psi)$ which yields the default threshold of 0.5 [6]. Another possibility adopted in this work is to set the threshold so that it matches the proportion of expected anomalies contained in the dataset, denoted as $\nu \ll 1$ (and fixed to $\nu = 0.04$ in the section devoted to experimental results).

IV. EXPERIMENTS

A. Performance criteria

Suitable metrics have to be defined to evaluate the performance of the algorithm for an appropriate choice of its hyperparameters. A popular performance measure used for binary hypothesis testing is the receiver operational characteristic (ROC) which shows the probability of detection as a function of the probability of false alarm. However, in AD, to bypass the imbalanced proportion of normal data and anomalies, it is common to consider Precision-Recall curves [14]. In addition to ROCs, classical performance measures include the precision, recall, F_1 -score and accuracy (Acc) defined as follows:

$$\begin{aligned} \text{Precision} &= \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \\ F_1 &= 2 \left(\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \right), \quad \text{Acc} = \frac{\text{TP} + \text{TN}}{N}, \end{aligned} \quad (6)$$

where TN, TP, FN and FP denote the numbers of true negatives, true positives, false negatives and false positives.

B. Trajectories with the same length

The first simulations were completed using synthetic trajectories (considered in [15]) containing $N_T = 1000$ sets of trajectories denoted as $\mathcal{E}_n, n = 1, \dots, N_T$. Each trajectory is defined by 16 positions $(x_i(1), x_i(2)) \in \mathbb{R}^2$. Each set \mathcal{E}_n is made up of 260 trajectories (hence $d = 260$) divided into 5

railways of 50 normal trajectories and 10 anomalies. Examples of a set before and after AD are displayed in Figs. 4 and 5. The dataset and results from [4] serve as a reference for performance comparison. The similarity measures based on DTW introduced before, as well as another popular distance measure called Time Warp Edit Distance (TWED), were computed between all pairs of elements to calculate the similarity matrix S . The isolation forest algorithm was finally applied to the columns of S using 200 trees and a max depth of 260.

The algorithm was also tested with and without applying PCA to the similarity matrix S . The variance ratio chosen for PCA was adjusted by cross validation leading to $r = 0.9999$ and the maximum proportion of anomalies was set to $\nu = 0.04$ for each set of trajectories (see Section III-D for details). In order to appreciate the performance of the method, Fig. 3 displays examples of anomaly scores provided by the proposed algorithm for one dataset \mathcal{E}_n . Anomaly scores tend to be small for all the normal samples and are significantly higher for the anomalies. Quantitative results reported in Table I and Fig. 6 allow the performance of the proposed method to be appreciated. A comparison with the reference method [4] and a previously proposed method is also possible using the results of Table II. Note that the proposed method is not compared to deep learning AD strategies such as [1], [2] since the size of the database is too small for an appropriate learning. Our conclusions are summarized below:

- 1) PCA allows the detection performance to be improved in all cases, which shows that dimensionality reduction is interesting for detecting anomalies in trajectories,
- 2) DTW seems to provide better results than TWED in most cases, with the advantage of being less computationally intensive,
- 3) the performance of the proposed DTW-based FIF is globally very satisfactory with an F_1 -score close to 90% in the best case (highlighted in bold in Table I) and is comparable to the results obtained from [3].

C. Incomplete trajectories with different lengths

This section considers incomplete trajectories obtained after removing some points of each trajectory randomly. More precisely, for each trajectory, a number of points (to be removed) was chosen uniformly at random between 0 and 5. The removed points were chosen randomly as well, yielding

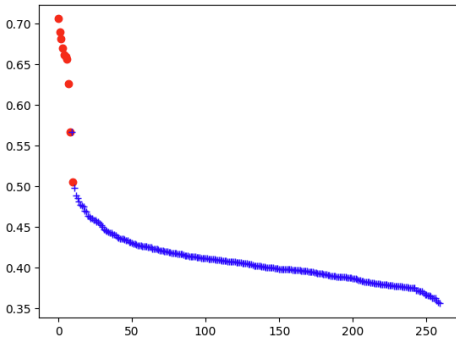


Fig. 3. Example of FIF scores (anomalies in red).

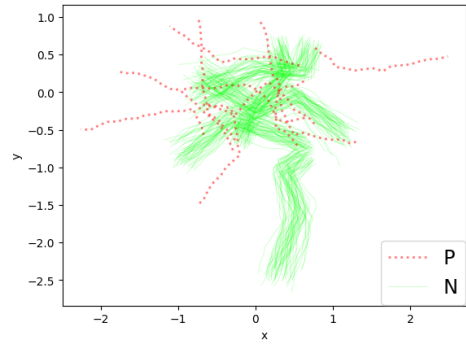


Fig. 4. Trajectories before AD.

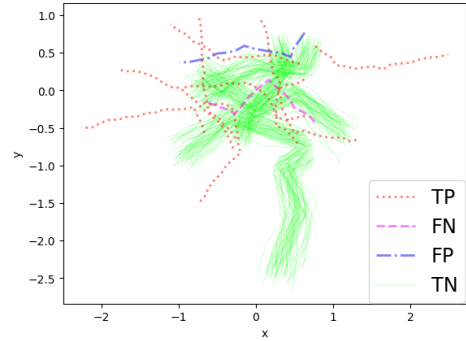


Fig. 5. Trajectories after AD (with one false positive and one false negative).

trajectories with different lengths. The detection performance obtained with these trajectories is shown in Fig. 7, as well as in Table IV, which can be compared to Table III. Due to missing data, the detection performance is slightly reduced when compared to the previous section but remains satisfactory and at least as efficient as the state-of-the-art. TWED-based measures seem to be more sensitive to the removal of points than DTW-based measures.

V. CONCLUSION

This work showed the interest of combining dynamic time warping with functional isolation forest for the detection of anomalies in trajectories for maritime surveillance. The different similarities proposed allow for the computation of

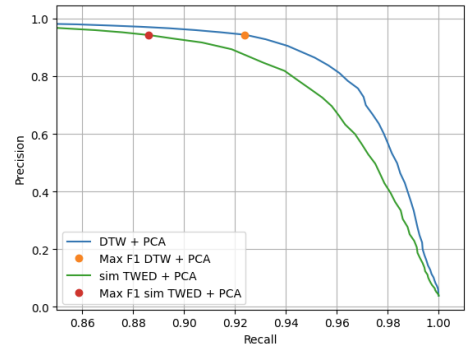


Fig. 6. Precision-Recall curves for complete data.

TABLE I
PERFORMANCES WITH FIF FOR COMPLETE DATA.

	Precision	Recall	F ₁	Acc
DTW	70.28%	77.31%	73.62%	97.87%
simDTW	72.89%	80.98%	76.36%	98.09%
cosDTW	78.84%	86.72%	82.59%	98.59%
GKDTW	62.32%	68.55%	65.29%	97.20%
TWED	71.00%	78.10%	74.38%	97.93%
simTWED	63.64%	70.00%	66.67%	97.31%
cosTWED	75.46%	83.01%	79.06%	98.31%
GKTWED	44.95%	49.44%	47.09%	95.73%
DTW PCA	86.41%	95.04%	90.52%	99.23%
simDTW PCA	81.44%	89.58%	85.31%	98.81%
cosDTW PCA	81.54%	89.69%	85.42%	98.82%
GKDTW PCA	85.67%	94.24%	89.75%	99.17%
TWED PCA	80.68%	88.75%	84.52%	98.75%
simTWED PCA	84.66%	93.13%	88.70%	99.09%
cosTWED PCA	80.55%	88.60%	84.38%	98.74%
GKTWED PCA	81.73%	89.90%	85.62%	98.84%

TABLE II

METRICS FROM THE STATE-OF-THE-ART FOR COMPLETE TRAJECTORIES.

	Precision	Recall	F ₁	Acc
SHNN-CAD [4]	—	—	—	97.09%
One-Class SVM [4]	58.78%	24.63%	31.82%	96.17%
One-Class SVM [7]	93.83%	66.32%	76.31%	98.51%
DBSCAN + LCSS [3]	96.20%	97.04%	96.45%	99.72%

TABLE III

METRICS FROM THE STATE-OF-THE-ART FOR INCOMPLETE TRAJECTORIES.

	Precision	Recall	F ₁	Acc
One-Class SVM [7]	93.97%	64.39%	75.21%	98.47%
DBSCAN + LCSS [3]	96.83%	94.59%	95.49%	99.66%

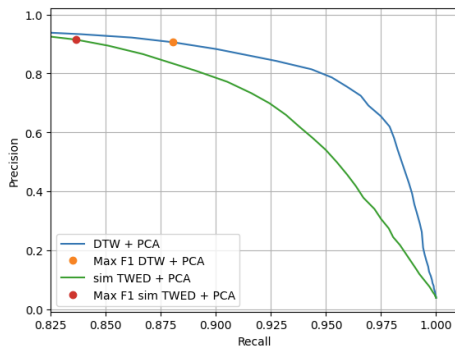


Fig. 7. Precision-Recall curves for incomplete data.

different anomaly scores for the purpose of AD. The best performance measures obtained with this method are globally very satisfactory with F₁ scores close to 90% in the best cases. These performances are close to the state-of-the-art, with only a slight decrease in the F₁. Some applications require to detect in real time abnormal ship behaviour (eg. piracy, illegal fishing, etc.), which will be considered in future work.

TABLE IV
PERFORMANCES WITH FIF FOR INCOMPLETE DATA.

	Precision	Recall	F ₁	Acc
DTW	60.77%	66.85%	63.67%	97.07%
simDTW	64.18%	70.60%	67.24%	97.35%
cosDTW	76.04%	83.64%	79.66%	98.36%
GKDTW	54.83%	60.31%	57.44%	96.56%
TWED	43.06%	47.37%	45.11%	95.57%
simTWED	66.75%	73.43%	69.93%	97.57%
cosTWED	68.03%	74.83%	71.27%	97.68%
GKTWED	31.36%	34.50%	32.86%	94.58%
DTW PCA	84.38%	92.82%	88.40%	99.06%
simDTW PCA	67.87%	74.66%	71.10%	97.67%
cosDTW PCA	74.57%	82.03%	78.12%	98.23%
GKDTW PCA	85.65%	94.21%	89.73%	99.17%
TWED PCA	51.91%	57.10%	54.38%	96.32%
simTWED PCA	80.97%	89.07%	84.83%	98.77%
cosTWED PCA	68.78%	75.66%	72.06%	97.74%
GKTWED PCA	61.65%	67.81%	64.58%	97.14%

REFERENCES

- [1] J. Venskus, P. Treigys, J. Bernataviciene, G. Tamulevicius, and V. Medvedev, "Real-time maritime traffic anomaly detection based on sensors and history data embedding," *Sensors*, pp. 1–10, Aug. 2019.
- [2] M. L. D. Dias, C. L. C. Mattos, T. L. C. Da Silva, J. A. F. De Macêdo, and W. C. P. Silva, "Anomaly detection in trajectory data with normalizing flows," in *Proc. International Joint Conference on Neural Networks (IJCNN)*, Glasgow, UK, July 2020, pp. 1–8.
- [3] Y. Zhang and W. Li, "Dynamic maritime traffic pattern recognition with online cleaning, compression, partition, and clustering of AIS data," pp. 1–18, Aug. 2022.
- [4] R. Laxhammar and G. Falkman, "Online learning and sequential anomaly detection in trajectories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 6, pp. 1158–1173, June 2014.
- [5] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, vol. 41, pp. 1–58, July 2009.
- [6] F. Liu, K. Ting, and Z. Zhou, "Isolation-based anomaly detection," *ACM Trans. Knowl. Discov. Data*, vol. 6, pp. 1–39, March 2012.
- [7] V. Mangé, J.Y. Tourneret, F. Vincent, L. Mirambell, F. Manzoni Vieira, and B. Pilastre, "Détection de comportements anormaux dans des trajectoires de navires avec one-class SVM et dynamic time warping," Grenoble, France, Aug. 2023, pp. 1–4, Proc. Conf. GRETSI.
- [8] L. Zhang, Y. Zhu, J. Ren, W. Lu, and Y. Yao, "A method for detecting abnormal behavior of ships based on multi-dimensional density distance and an abnormal isolation mechanism," *Mathematical Biosciences and Engineering*, vol. 20, pp. 13921–13946, March 2023.
- [9] G. Staerman, P. Mozharovskiy, S. Cléménçon, and F. d'Alché-Buc, "Functional isolation forest," in *Proc. of The Eleventh Asian Conference on Machine Learning*. Nov. 2019, vol. 101, pp. 332–347, Proc. of Machine Learning Research.
- [10] F. Meng, G. Yuan, S. Lv, Z. Wang, and S. Xia, "An overview on trajectory outlier detection," *Artificial Intelligence Review*, vol. 52, pp. 2437–2456, Feb. 2018.
- [11] M. Badiane and P. Cunningham, "An empirical evaluation of kernels for time series," *Artificial Intelligence Review*, vol. 55, pp. 1803–1820, July 2021.
- [12] V. V. Trinh, K. P. Tran, and T. T. Huong, "Data driven hyperparameter optimization of one-class support vector machines for anomaly detection in wireless sensor networks," in *Proc. Int. Conf. Advanced Tech. for Commun. (ATC)*, Quy Nhon, Vietnam, Oct. 2017, pp. 6–10.
- [13] C. C. Aggarwal, *Outlier Analysis*, Springer Verlag, New York, Dec. 2016.
- [14] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets," *PLOS One*, vol. 10, no. 3, pp. 1–21, March 2015.
- [15] C. Piciarelli, C. Micheloni, and G. L. Foresti, "Trajectory-based anomalous event detection," *IEEE Trans. Circuits Sys. Video Technol.*, vol. 18, no. 11, pp. 1544–1554, Sept. 2008.