Mini-Project (ML for Time Series) - MVA 2023/2024

Antoine Ratouchniak antoine.ratouchniak@ens-paris-saclay.fr
Hugo Queniat hugo.queniat@telecom-paris.fr

January 15, 2024

1 Introduction and contributions

Anomaly detection in time series is a crucial topic in many fields ranging from finance and healthcare to monitoring and meteorology. Anomalies, often manifested with outliers, irregularities, or distinctive patterns, can serve as indicators of important events such as malfunctions, seizures etc. Consequently, developing robust anomaly detection methodologies is essential (to ensure comprehension of the data and the effective detection of those vents).

To conduct this work, we took a close look to **Anomaly Detection in Time Series: A Comprehensive Evaluation** by Schmidl, Wenig et al. [23]. Throughout the article, the writers carry an extensive review of the existing anomaly detection methods: they tested no less than 61 different algorithms. Indeed, the broadness of applications for time series data triggers interest in many different areas of Mathematics, from Statistics to Signal Processing and Machine Learning. The authors took advantage of this by dealing with very different methods coming from distinct areas of study.

To exploit the richness of the article and provide a critical view of some of the methods presented as well as the results gathered by the authors, we have picked a handful of methods to check the reliability of the results. In addition, we sought to play with the parameters of the models to check the model's limits and evaluate the influence of its parameters on the performance of the algorithm. We took advantage of the wide range of methods on offer to try out some very different paradigms and thinking as you will see in the next section.

We each selected a few methods to study: Antoine worked on the Median method, FFTBSOM, PS-SVM, and Sub-LOF while Hugo focused on SR, STAMPi, and DWT-MLEAD. Depending on the method, we both implemented and used the public code available. We used available code for the Median method while implementing the multiscale setting, we used available code for FFTBSOM and we coded SR entirely. We also used the stumpy Python library [12] for STAMPi and we used the available code for PS-SVM and DWT-MLEAD. We used scikit-learn [17] for Sub-LOF. The paper conducted a very thorough review of the methods, so we chose distinct datasets to verify their results and, then, evaluate our modifications or certain parameters whose influence had not been tested. Finally, we modified some of the algorithms to try and improve their performance as well as finding parameters to ensure optimal performance and robustness.

2 Method

Throughout this work, we denote $x = \{x[1], \dots, x[N]\} \in \mathcal{X} \subseteq \mathbb{R}^N$ an univariate time series with length N. The indices $k \in \{1, \dots, N\}$ may be referred to as timestamps, points or instants.

An anomaly in a time series refers to an instant or a sequence of instants that deviate from the regular patterns of the series. The different algorithms we present will return a *time series scoring* $S = \{s[1], \dots, s[N]\}$ where each value indicates how anomalous a data point is.

To assess the algorithm's performance, we employ the same metrics as the survey that we are reminding.

$$\text{Precision} = \frac{TP}{TP + FP} \quad \text{Recall} = \frac{TP}{TP + FN} \quad \text{Expectancy} = \frac{FP}{TN + FP}$$

where TP stands for True Positive, FP for False Positive and and FN for False Negative.
The **AUC-ROC** is the area under the ROC curve. The ROC curve is a plot where the x-axis is the Precision as known as True Positive Rate (TPR) and the y-axis is the Expectancy as known as False Positive Rate (FPR), both computed at different thresholds. This value can be interpreted as the model's performance and can be particularly interesting in the case of imbalanced datasets as it focuses on the trade-off between TPR and FPR.
The **AUC-PR**, on the other hand, is the area under the Precision-Recall curve. As suggested by its name, the x-axis represents the Precision and the y-axis the Recall. This value encapsulates the model's ability to reasonably identify anomalies while avoiding undue false alarms.

2.1 Median method

The Median method [2] is an intuitive algorithm that classifies points as anomalies if they are too far from the median in a given window. Formally, we let $\kappa \in \mathbb{N}^*$ the window size, we define a set of points $\gamma[t] = \{x[t-\kappa], \ldots, x[t-1], x[t+1], \ldots, x[t+\kappa]\}\ \forall t \in \{\kappa+1, N-\kappa\}$ of size 2κ. A point $x[t]$ is classified as an anomaly if $|x[t] - \text{med}(\gamma[t])| > \tau\sigma(\gamma[t])$ where $\text{med}(\cdot)$ indicates the median, $\sigma(\cdot)$ the standard deviation and τ a threshold. One can make an analogy with [20, 13]. Finally, we slide the window along the time series to determine whether each point is considered an anomaly. Instinctively, the algorithm will detect well aberrant values but may exhibit reduced performance on pattern anomalies. An example is shown in Figure 1.

2.2 FFTBSOM

The Fourier transform based spatial outlier mining (FFTBSOM), introduced in [18] is, as the name states, based on the Fourier transform of a signal. First, we calculate the difference between the original signal and the signal with a low-pass filter applied to it. By doing this, we expect outliers, which can be sometimes associated with abrupt changes, to have larger values. We call the large value of this difference local outliers. Next, a set of points, or region, is considered as anomaly if two consecutive local outliers have opposite signs. Outliers are also constrained to a specific length. We show a toy example in Figure 2.

2.3 Spectral Residual

The **Spectral Residual**, SR, was first introduced in the context of saliency detection in a Computer Vision setting [9]. The method relies on analysis in the Fourier space. Indeed, as its name indicates, the goal is to recover the spectral residuals meaning, for a time series x :

$$R(\nu) = \log(|\hat{x}(\nu)|) - \overline{\log(|\hat{x}(\nu)|)}$$

the residuals are computed as the difference between the log-spectrum and its average. Then, the Saliency Map $S(x)$ is computed through the Inverse Fourier Transform. The computation anomaly detection scores thus become only the computation of the local relative amplitude of $S(x)$ as demonstrated by Ren et al. [19] and shown in 4.

2.4 Phase Space SVM

Phase Space SVM, denoted as PS-SVM, introduced in [14], is a kernel method to detect anomalies. We first recall essential definitions. Let $\phi : \mathcal{X} \to F$ be a feature map where $\mathcal{X}$ is the input space and $F \subseteq \mathbb{R}^n$ is the feature space. The goal of this embedding function is to transform (possibly) non-linear relations into linear ones, thus making the data separable by a hyperplane. However, as the number of features increases, the number of combinations will grow as well, leading to intractable computations. To solve this problem, we use the kernel trick, which represents the data as a set of pairwise similarity comparisons between the embedded data. We define the kernel $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ as

$$K(x, y) = \langle \phi(x), \phi(y) \rangle$$

To detect anomalies, the authors propose to unfold the time series using a time-delay embedding process [16]. In simple words, it shifts the signal by a certain delay as $x_d[t] = \{x[d], \dots x[t]\}$. Following that, we can consider the set of shifted time series $S_d = \{x_d[t] \mid t \in \{d, \dots, N\}\}$. The final step is to use a one-class SVM on these vectors in the phase space. The optimization problem is formulated as

$$\begin{aligned} \underset{w,\rho,\xi}{\arg\min} \quad & \|w\|_2^2 + \frac{1}{\nu(N-d)} \sum_{i=1}^{N-d} \xi_i - \rho \\ \text{s.t.} \quad & \langle w, \phi(S_d[i]) \rangle \geq \rho - \xi_i \\ & \xi_i \geq 0,\ i = 1, \dots, N-d \end{aligned} \tag{1}$$

where ν is an upper bound on the fraction of anomalies and a lower bound of the fraction of support vectors; ξ_i are slack variables; ρ is the threshold and w is the hyperplane to be found. Solving the dual problem of (1) [24] yields

$$\begin{aligned} \underset{\alpha}{\arg\min} \quad & \frac{1}{2} \alpha^T K \alpha \\ \text{s.t.} \quad & \|\alpha\|_1 = 1 \\ & 0 \leq \alpha_i \leq \frac{1}{\nu(N-d)},\ i = 1, \dots, N-d \end{aligned} \tag{2}$$

2.5 STAMPi

The **STAMPi** method exploits the knowledge acquired by computing the Matrix Profile, first introduced by Yeh et al. in [26]. Its computation solves the all-pair-similarity-search problem, meaning that the initial objective of the algorithm is to compute the *1NN* join function $\theta_{1\text{nn}}$: for two sub-sequences sets $\boldsymbol{A}$ and $\boldsymbol{B}$, $\theta_{1\text{nn}}(\boldsymbol{A}[i], \boldsymbol{B}[j]) = 1$ if and only if $\boldsymbol{B}[j]$ is the closest neighbor of $\boldsymbol{A}[i]$ in $\boldsymbol{B}$. From this function, we deduce the similarity join set $\boldsymbol{A} \bowtie_{\theta 1\text{nn}} \boldsymbol{B}$, which contains the tuple paired by $\theta_{1\text{nn}}$.

With this knowledge, the Matrix Profile $\mathbf{P_{AB}}$ is defined as the vector of Euclidean distances between each pair of sub-sequences in $\boldsymbol{A} \bowtie_{\theta 1\text{nn}} \boldsymbol{B}$. To harness this quantity for our problem, the idea forwarded in STAMPi is to choose a single sub-sequence-length m and compute:

$$\boldsymbol{A} \bowtie_{\theta 1\text{nn}} \boldsymbol{A} \text{ for } \boldsymbol{A} = \{x[i : i + m] \mid 0 \leq i \leq |x| - m\}$$

As a result, we obtain the anomaly scores with the values stored in $\mathbf{P_{AA}}$: the farther the nearest neighbor to a certain pattern, the more anomalous this pattern must be 5.

2.6 Sub-LOF

Subsequence Local Outlier Factor (Sub-LOF) is a well-known and powerful technique introduced in [4] that measures the local density deviation of each data point to decide whether these are outliers or not.

First, we let $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$ a distance and define the $k - distance(z)$ of a data point $z \in \mathcal{X}$ as the distance to its k^{th} nearest neighbor i.e. the k^{th} closest point. We denote the k neighborhood of z as $N_k(z) = \{p \in \mathcal{X} \mid d(z,p) \leq k - distance(z)\}$. We also define the reachability distance (RD) as the distance required to travel from a given point to its neighboring point.

$$\text{RD}_k(z,p) = \max(k - distance(p), distance(z,p)) \quad \text{where } p \in \mathcal{X} \text{ is an arbitrary point}$$

Subsequently, to identify reachable points in a neighborhood, we define the local reachability density (LRD) as the inverse of the average RD of its neighbors.

$$LRD_k(z) = 1/\left(\frac{\sum_{p \in N_k(z)} RD_k(z,p)}{|N_k(z)|}\right)$$

Finally, outliers are identified by assigning a score with the LOF by comparing the LRD of a point with its k-neighbors.

$$LOF_k(z) = \frac{\sum_{p \in N_k(z)} \frac{LRD(p)}{LRD(z)}}{|N_k(z)|}$$

To apply this algorithm to time series, the signal is split into subsequences.

2.7 Discrete Wavelet Transform MLE Anomaly Detection

The **DWT-MLEAD** method was introduced by Thill et al. in [25] and combines the Discrete Wavelet Transform with a Statistics approach to tackle the anomaly detection problem.

The method uses Haar's Wavelet [8] to compute the DWT, on which a sliding window is passed through to retrieve temporal relationships. This outcomes matrices of detailed and approximation coefficients for each level of the DWT, which rows we fit a multivariate Gaussian distribution. Finally, we use the likelihood as an efficient anomaly detector: the lower the likelihood, the more likely the window corresponds to an anomalous pattern.

3 Data

We use the 5 datasets available on univariate time series used by the authors to conduct the experiments. Among them, 3 are captured from real data while 2 of them are a mix of real and synthetic data. The data can be of varying lengths with a few or many anomalies within segments. The datasets we use are Dodgers [11], which contains real data, NAB [1] which contains both real and synthetic data, NASA-MSL and NASA-SMAP [10] that both contain real data and NormA [3] that contains both real and synthetic data.

4 Results

4.1 Median method

We propose to use different window sizes and average their results, in a multi-scale way, to capture further details throughout the time. Indeed, having multiple views on a subsequence can be important in deciding on an anomaly. We note a slight improvement using the multiscale setting in section B.

4.2 FFTBSOM

As shown in Figure 2, this method is sensitive to noise. Indeed, as we have seen in the class, the power spectrum of noise (supposed AWGN), is spread uniformly in the power spectrum. To remove as much noise as possible while preserving the important features, we propose to smooth the data. We explore three denoising techniques: moving average, dictionary learning [6, 21] and Savitzky-Golay filter [22]. Our findings in section B reveal a clear improvement when using the moving average and Savitzky-Golay filters. On the other hand, the lower performances of dictionary learning may be explained due to the varying sizes (potentially long) of each time series.

4.3 Spectral Residual

During their experiments, the authors proposed a fixed set of parameters that they had determined empirically for the sizes of the windows in both the approximation of the Saliency Map and the computation of the scores. Both w and m showed to have very little to no influence on the performance, however as shown in section B, the performance increases with the final parameter z, the window to compute the score.

4.4 Phase Space SVM

The core of this method is time-delay embedding process. To this end, we try feeding the algorithms with multiple shifts: 3, 3 and 5, 3, 5 and 7, and 3, 5, 7 and 9. The results in section B seem to show that there is not much difference with shifts. A higher shift could probably help. Other techniques could have involved adding a regularization term, adding weight to the sample (for example giving more weights to samples that are suspected to be outliers), or changing the metric.

4.5 STAMPi

In the original paper, they only tested the **STAMPi** method with a single fixed sub-sequence length of 256. Even though it could be the perfect value to balance accordingly performance and robustness to different timeseries behaviors, it seems interesting to investigate the influence of the unique parameter of the method. We clearly see in B that when the length approaches 0, the method loses performance drastically in terms of **AUC-PR**. This is because the subsequences become too thin to apprehend any usual pattern of the series, hence there are lots of false anomalies detected.

4.6 Sub-LOF

The metric can affect the performance of a model. Consequently, we try the three following metrics: DTW [15], the Soft-DTW [7], and the Wasserstein-Fourier [5]. All of these metrics make sense, the DTW would measure the temporal distortions of the signal, the soft-DTW is a more robust version of it while the Wasserstein-Fourier can capture dissimilarities using the spectrum. Unfortunately, computing these metrics, especially the DTW, is not scalable on large datasets. Instead, we display results on toy datasets in Figure 3. The results seem to be the most accurate using the Wasserstein-Fourier distance. This may be explained by the fact that in these examples, the changes are sudden, allowing this distance to capture it. These results have to be used cautiously as they are only tested on toy datasets.

4.7 Discrete Wavelet Transform MLE Anomaly Detection

While building their model, the authors have only thought about the multivariate Gaussian to fit the distribution of the slid DWT coefficients. Even though it is by far the most commonly used distribution and its log density is easy to compute, we thought that we may try to extend the algorithm to other distributions. The idea is still to compute the mean and the covariance matrix through the usual MLE

estimator, so we wanted distributions whose parameters were indeed (μ, Σ). Hence, we chose both the Laplace distribution and the Student. This allowed us to evaluate the impact of the tail of the distribution. Indeed, the Laplace distribution is known for its strong tail, and as expected the detection performance is lower since even samples very far from the mean could still have a non-negligible likelihood. On the other hand, we observed that when we increase the degree of freedom in the Student distribution, meaning when we reduce the tail, the performance increases although we were not able to reach that of the Gaussian distribution.

References

[1] Subutai Ahmad, Alexander Lavin, Scott Purdy, and Zuha Agha. Unsupervised real-time anomaly detection for streaming data. *Neurocomputing*, 262:134–147, 2017.

[2] Sabyasachi Basu and Martin Meckesheimer. Automatic outlier detection for time series: an application to sensor data. *Knowledge and Information Systems*, 11:137–154, 2007.

[3] P. Boniol, M. Linardi, F. Roncallo, and T. Palpanas. Automated anomaly detection in large sequences. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, pages 1834–1837, 2020.

[4] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 93–104, 2000.

[5] Elsa Cazelles, Arnaud Robert, and Felipe Tobar. The wasserstein-fourier distance for stationary time series. *IEEE Transactions on Signal Processing*, 69:709–721, 2020.

[6] Scott Shaobing Chen, David L Donoho, and Michael A Saunders. Atomic decomposition by basis pursuit. *SIAM review*, 43(1):129–159, 2001.

[7] Marco Cuturi and Mathieu Blondel. Soft-dtw: a differentiable loss function for time-series. In *International conference on machine learning*, pages 894–903. PMLR, 2017.

[8] Alfred Haar. Zur theorie der orthogonalen funktionensysteme - erste mitteilung. *Mathematische Annalen*, 69, 1910.

[9] Xiaodi Hou and Liqing Zhang. Saliency detection: A spectral residual approach. 2007.

[10] Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom. Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 387–395, 2018.

[11] Jon Hutchins. Dodgers Loop Sensor. UCI Machine Learning Repository, 2006. DOI: https://doi.org/10.24432/C51P50.

[12] Sean M. Law. STUMPY: A Powerful and Scalable Python Library for Time Series Data Mining. *The Journal of Open Source Software*, 4(39):1504, 2019.

[13] Christophe Leys, Christophe Ley, Olivier Klein, Philippe Bernard, and Laurent Licata. Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. *Journal of experimental social psychology*, 49(4):764–766, 2013.

[14] Junshui Ma and Simon Perkins. Time-series novelty detection using one-class support vector machines. In *Proceedings of the International Joint Conference on Neural Networks, 2003.*, volume 3, pages 1741–1745. IEEE, 2003.

[15] Meinard Müller. Dynamic time warping. *Information retrieval for music and motion*, pages 69–84, 2007.

[16] Norman H Packard, James P Crutchfield, J Doyne Farmer, and Robert S Shaw. Geometry from a time series. *Physical review letters*, 45(9):712, 1980.

[17] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[18] Faraz Rasheed, Peter Peng, Reda Alhajj, and Jon Rokne. Fourier transform based spatial outlier mining. In *Intelligent Data Engineering and Automated Learning-IDEAL 2009: 10th International Conference, Burgos, Spain, September 23-26, 2009. Proceedings 10*, pages 317–324. Springer, 2009.

[19] Hansheng Ren, Bixiong Xu, Yujing Wang, Chao Yi, Congrui Huang, Xiaoyu Kou, Tony Xing, Mao Yang, Jie Tong, and Qi Zhang. Time-series anomaly detection service at microsoft. 2019.

[20] SW Roberts. Control chart tests based on geometric moving averages. *Technometrics*, 42(1):97–101, 2000.

[21] Ron Rubinstein, Alfred M Bruckstein, and Michael Elad. Dictionaries for sparse representation modeling. *Proceedings of the IEEE*, 98(6):1045–1057, 2010.

[22] Abraham Savitzky and Marcel JE Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry*, 36(8):1627–1639, 1964.

[23] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. Anomaly detection in time series: A comprehensive evaluation. volume 15, 2022.

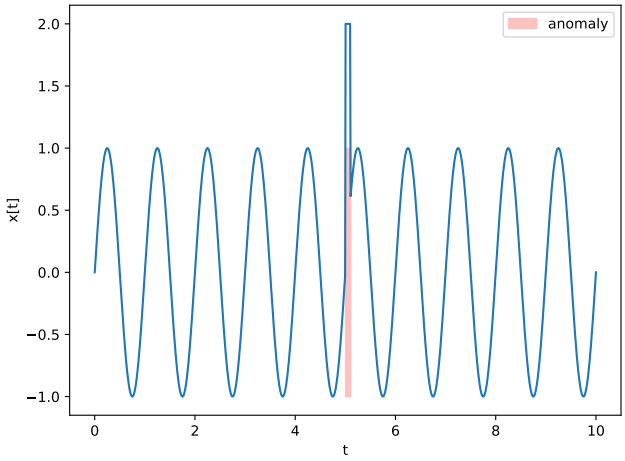
[24] Bernhard Schölkopf, Robert C Williamson, Alex Smola, John Shawe-Taylor, and John Platt. Support vector method for novelty detection. *Advances in neural information processing systems*, 12, 1999.

[25] Markus Thill, Thomas Bäck, and Wolfgang Konen. Time series anomaly detection with discrete wavelet transforms and maximum likelihood estimation. *Proceedings ITISE*, 2017.

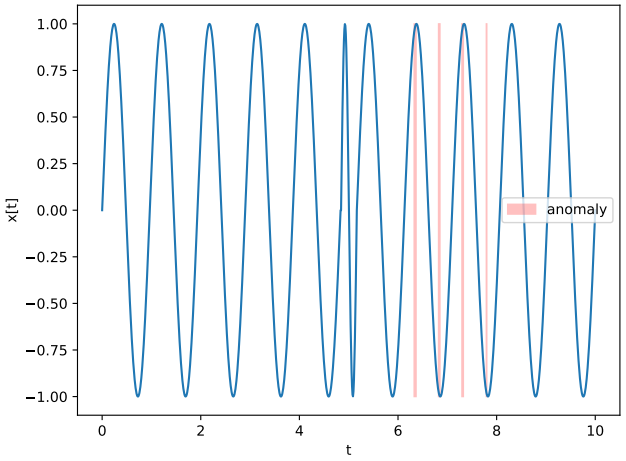
[26] Chin Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. Matrix profile i: All pairs similarity joins for time series: A unifying view that includes motifs, discords and shapelets. volume 0, 2016.

A Toy examples

A.1 Median method



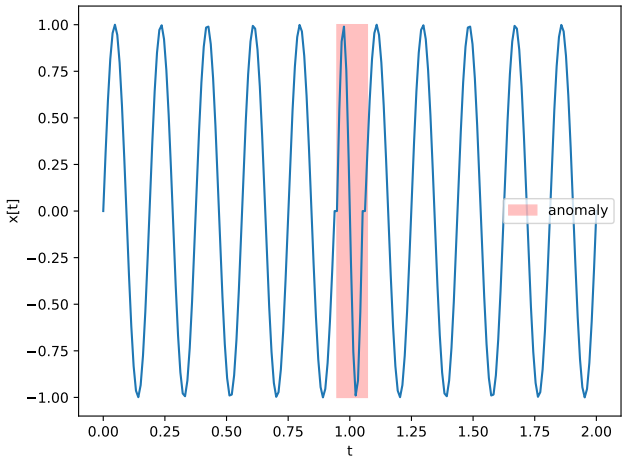
Anomaly well detected



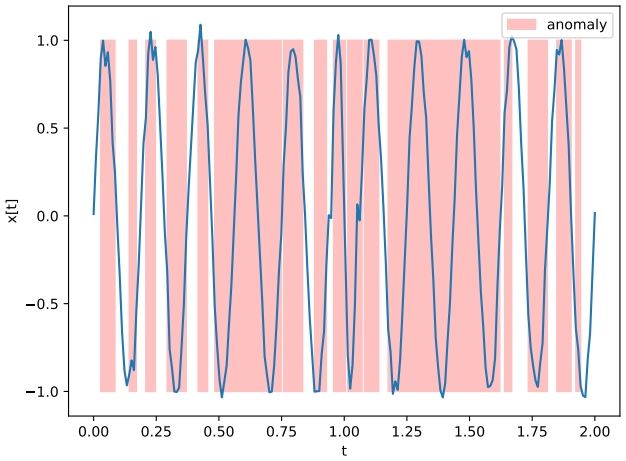
Sudden change of patterns are not easily detected using the median method

Figure 1: Anomaly detection using the median method ($\tau = 1$)

A.2 FFTBSOM



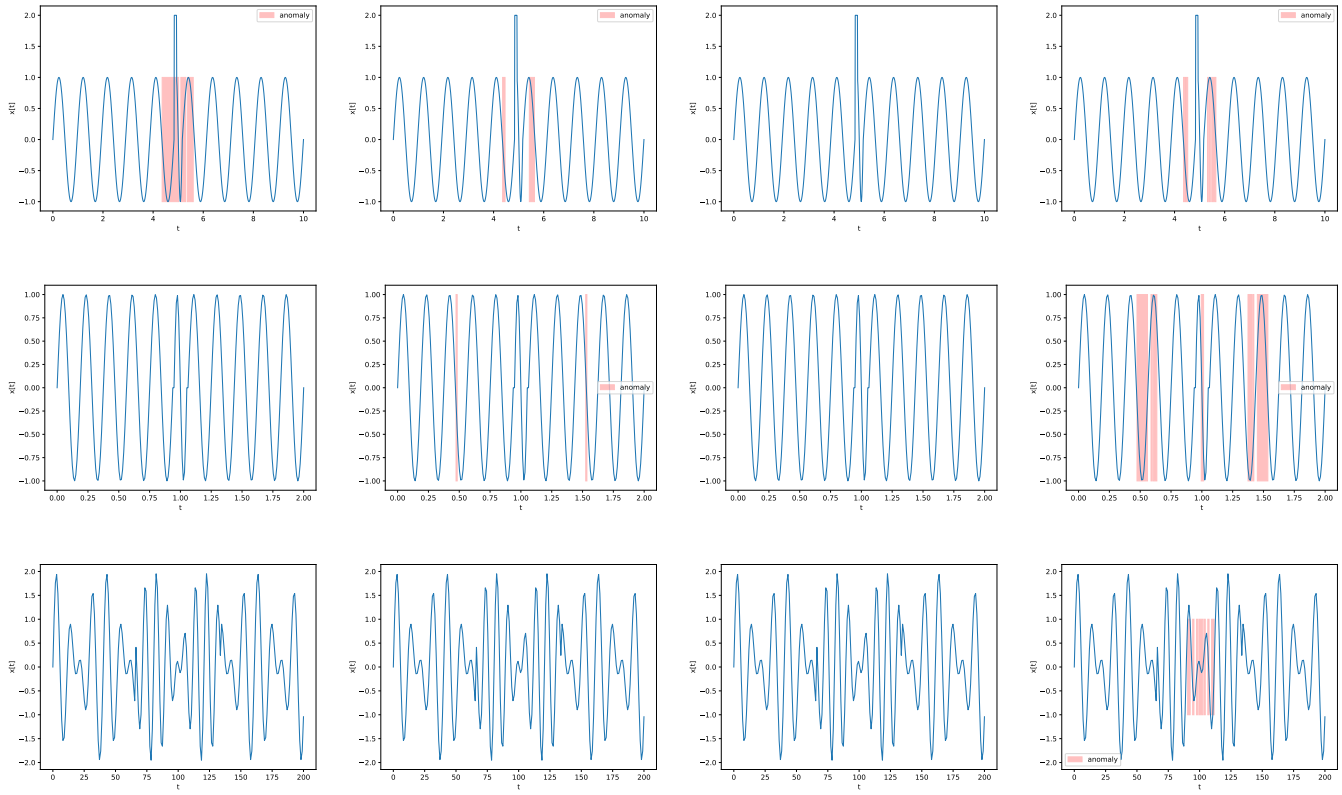
Anomaly well detected



Anomaly not detected because of Gaussian noise

Figure 2: Anomaly detection using the FFTBSOM method. We use a sinewave where the ordinary frequency changes suddenly in the middle.

A.3 Sub-LOF



Minkowski (default metric) | DTW | Soft-DTW with $\gamma = 1$ | Wasserstein-Fourier

Figure 3: Anomaly detection using Sub-LOF with different metrics

A.4 Spectral Residual

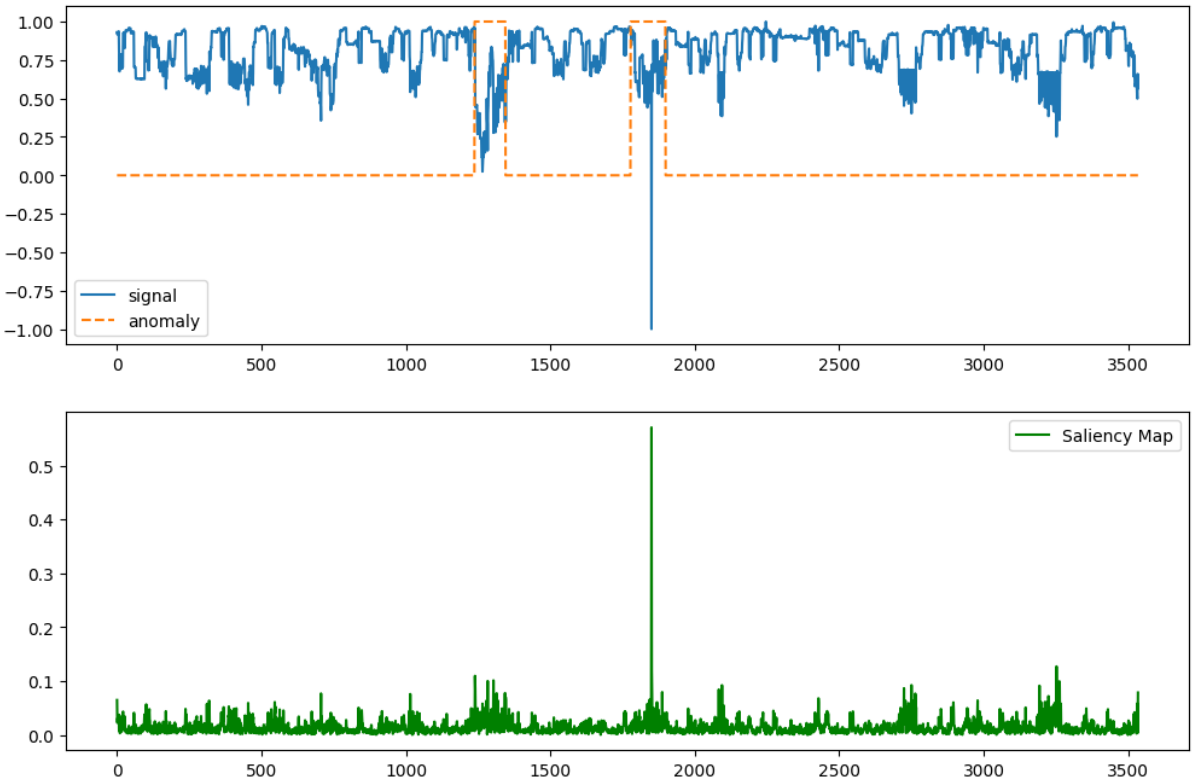


Figure 4: Evolution of the absolute value of the saliency map

A.5 STAMPi

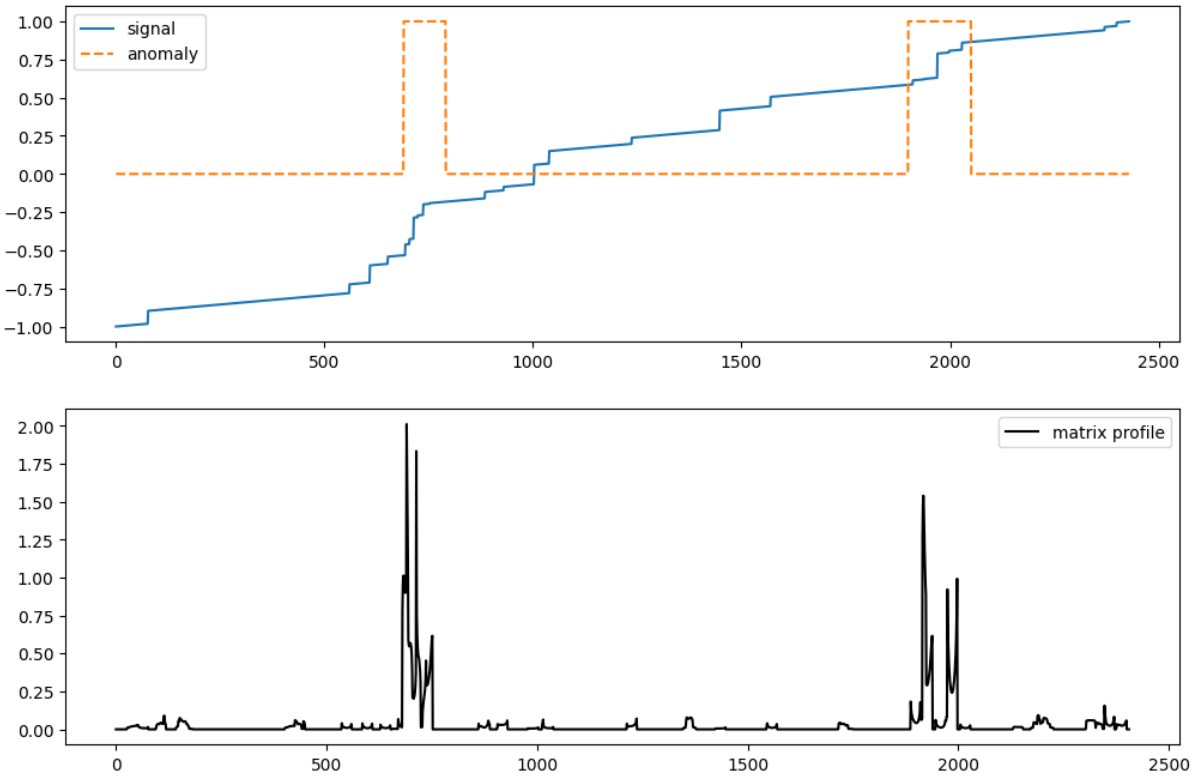


Figure 5: Instance of anomaly detection, with the scores, the matrix profile displayed

B Performances

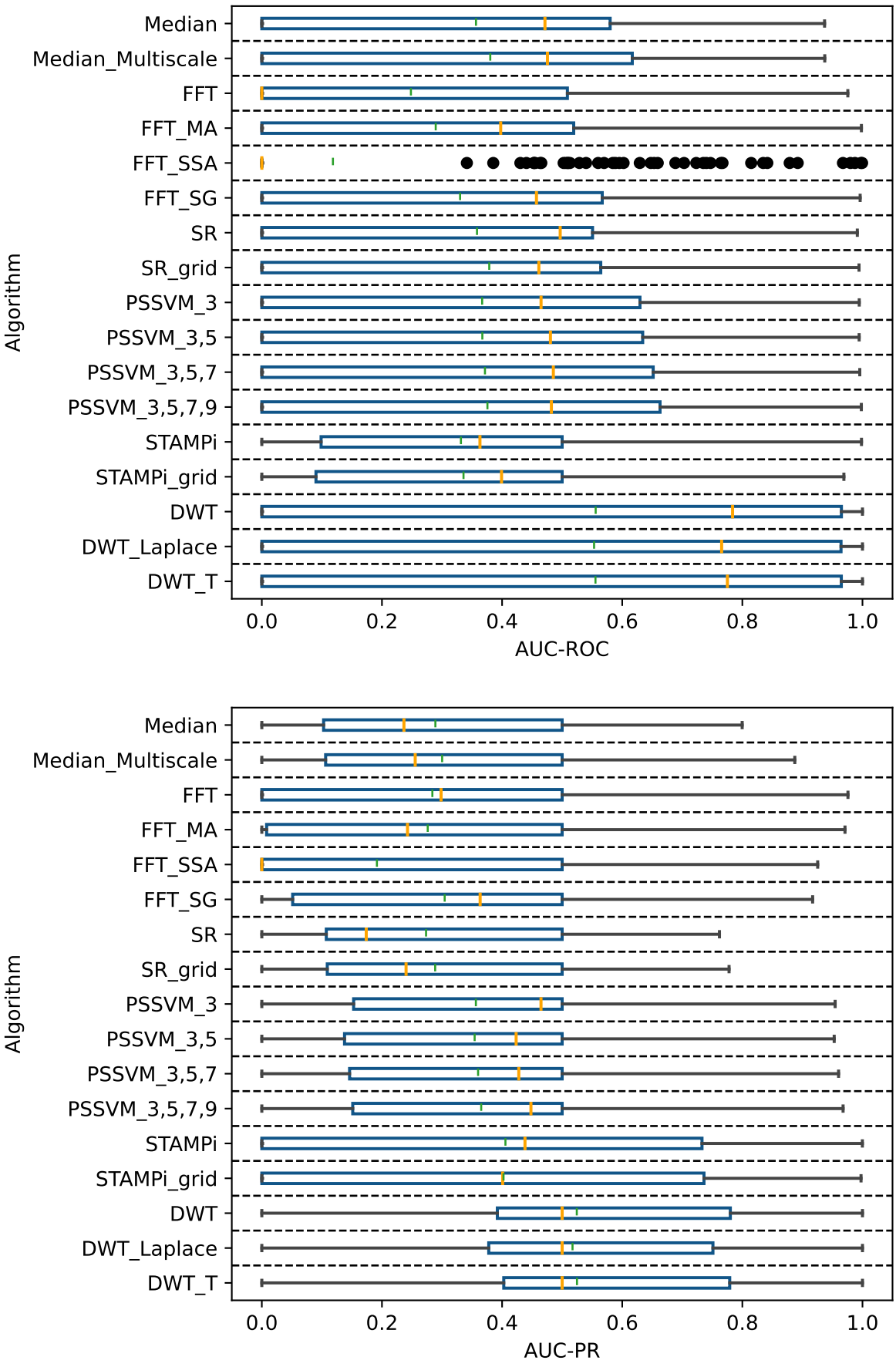


Figure 6: Boxplots performances of the algorithms and modified algorithms we used for the AUC-ROC and AUC-PR metrics. The mean value is in green and the median in orange.

Method / Score	ROC AUC	PR AUC
Median method	0.47	0.24
Median method improved	**0.48**	**0.26**
FFTBSOM	0.0	0.30
FFTBSOM moving average	0.4	0.24
FFTBSOM dictionary learning	0.0	0.0
FFTBSOM Savitzky-Golay	**0.46**	**0.36**
SR		
SR_grid		
PS-SVM 3	**0.46**	0.46
PS-SVM 3,5	0.48	0.42
PS-SVM 3,5,7	**0.49**	0.43
PS-SVM 3,5,7,9	0.48	0.45
STAMPi		
STAMPi-grid		
DWT	**0.78**	**0.50**
DWT-Laplace	0.76	**0.50**
DWT-T	0.77	**0.50**

Table 1: Medians obtained for the algorithms used

Method / Score	ROC AUC	PR AUC
Median method	0.36	0.29
Median method improved	**0.38**	**0.30**
FFTBSOM	0.25	0.28
FFTBSOM moving average	0.29	0.28
FFTBSOM dictionary learning	0.12	0.19
FFTBSOM Savitzky-Golay	**0.33**	**0.30**
SR		
SR_grid		
PS-SVM 3	0.36	0.36
PS-SVM 3,5	0.35	0.42
PS-SVM 3,5,7	0.36	0.43
PS-SVM 3,5,7,9	**0.38**	**0.37**
STAMPi		
STAMPi-grid		
DWT	**0.55**	0.52
DWT-Laplace	**0.55**	0.51
DWT-T	**0.55**	0.52

Table 2: Means obtained for the algorithms used

Value of z	ROC AUC	PR AUC
3	0.47	0.13
23	0.50	0.17
43	0.50	0.19
63	**0.51**	**0.20**

Table 3: Means obtained for different values of the score window size, z, in the Spectral Residual algorithm.

Value of m	ROC AUC	PR AUC
4	**0.48**	0.19
8	0.47	0.21
16	0.45	0.31
32	0.43	0.37
64	0.44	0.38
128	0.41	**0.40**
256	0.43	0.36

Table 4: Means obtained for different values of the subsequence length, m, in the STAMPi algorithm.