

EXPERIMENTAL SETTINGS

The machine used in the experiments has two AMD7742 64-core CPUs with 1024GB memory.

Subsequences in each time series are preprocessed with z-score normalization. All final scores output by detectors are normalized in $[0, 1]$. For detectors that rely on randomization, we report the average result of 10 trials on each time series.

[Datasets] Synthetic datasets noisy_sine and ARMA are from previous work[10]. Real datasets includes MIT-BIH Supraventricular Arrhythmia Database (MBA)[8, 14] and other datasets which are from various domains and have been studied in earlier works [10, 11, 24].

Some datasets have two versions, e.g., ann_gun and stdb_308; and each version uses one of the two variables. Because either version produces similar AUC for most detectors, we have chosen to use one only. Some datasets are trivial, e.g., chfdb_chf0175 and qtdbsel102; and all detectors have perfect results(AUC=1) so we do not show them in table 7.

We label anomalous periods for each time series referring to the previous work[10, 11, 24]. Details can be seen in the document "parameter.csv".

Note that all time series are periodic and the period length of each can be easily known from the prior knowledge about data, e.g., the frequency of sampling in data collection. The period of some datasets vary slightly but it has no effect on the result of algorithms. Our algorithm works well when choosing subsequence length to be roughly the length of the period.

The parameters of some datasets should be specially explained as follows.

dutch_pwrdemand: This time series has power consumption for a Dutch research facility for the year 1997 (one power measurement every 15 minutes for 365 days). It shows a characteristic weekly pattern that consists of 5 power usage peaks corresponding to the 5 weekdays followed by 2 days of low power usage on the weekends. Anomalous weeks occur when one or more of the normal usage peaks during a week do not occur due to holidays[10]. We can easily figure out that there are 672 points in each week($672=7 \times 24 \times 60/15$) so the period length is 672. It is Wednesday on January 1st so we let each week period start from Wednesday. There are a total of 6 anomalous weeks. Some papers[2, 10, 11] use this dataset with less anomalies because they treat continuous anomalous weeks as 1 anomaly. The detailed information about this dataset can be seen in [32].

ann_gun: The anomalous period 2 is annotated by Keogh's work [1] as Figure 4 shows. Other anomalous period subsequences which are proposed in Boniol's [2] can be seen in Figure 5.

Patient_respiration: We use the subset of dataset nprs44, beginning at 15500 and ending at 22000. There are one apparent anomaly and one subtle anomaly in this dataset as shown in Figure 6 [1].

TEK: Inspired by [10], we also concatenate dataset TEK14, TEK16 and TEK17 as TEK of length 15000. In Keogh's work [1], a total of 4 anomalies are marked. But in TEK14, 2 anomalous snippets belongs to the same period as shown in Figure 7. Since we detect anomalous subsequences by the unit of period, we treat it as

one anomalous period subsequence of length 1000. So there are 3 anomalous subsequences in our annotations of this dataset.

MBA803, MBA805, MBA806, MBA820, MBA14046: We use the subset of first 100K points from the full datasets of each like Boniol does in [2].

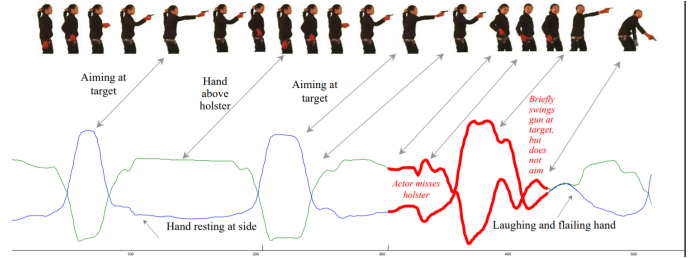


Figure 4: One anomaly period in ann_gun dataset cited from [1]

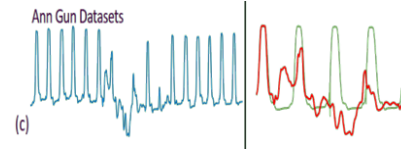


Figure 5: Anomalous periods in ann_gun dataset cited from [2]

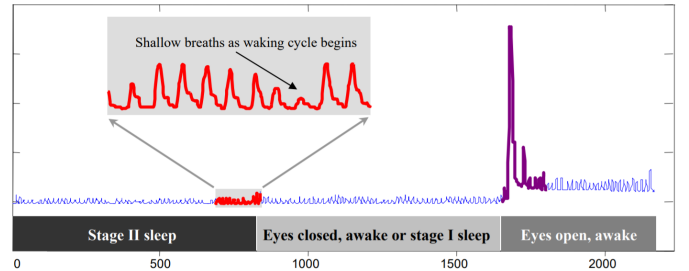


Figure 6: Anomalies in Patient_respiration dataset cited from [1]

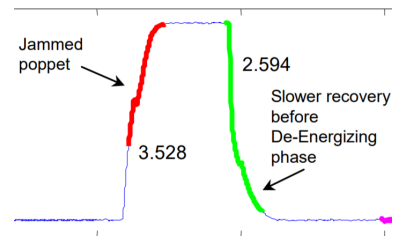


Figure 7: Anomalies in a period of TEK14 dataset cited from [1]

[Algorithms] The STOMP [34] implementation of MP is used; NormA is from <http://helios.mi.parisdescartes.fr/~themisp/norma/>; IDK-based detectors are our implementations based on [29]; and WFD is from github.com/GAMES-UChile/Wasserstein-Fourier. Others are from scikit-learn.org. All are in Python.

As for 1Line method, we use 5 type of basic vectorized primitive operations in Matlab:

- (i) $\pm \text{diff}(\mathbf{TS})$
- (ii) $\pm \text{movmax}(\mathbf{TS}, \omega)$
- (iii) $\pm \text{movmin}(\mathbf{TS}, \omega)$
- (iv) $\pm \text{movmean}(\mathbf{TS}, \omega)$
- (v) $\pm \text{movstd}(\mathbf{TS}, \omega)$

where \mathbf{TS} is the time series, ω is the window size to compute ω -points maximum, minimum, mean values or standard deviations. We run these 5 one-liner on each dataset and report the median value of AUC.

[Measures] The detection accuracy of an anomaly detector is measured in terms of AUC (Area under ROC curve). As all the anomaly detectors are unsupervised learners, all models are trained with unlabelled training sets. Only after the models have made

predictions, ground truth labels are used to compute the AUC for each dataset.

Given a periodic time series $T = [t_1, t_2, \dots, t_n]$ of length n and period length m , a subsequence $T_{i,m}$ of a data series T is a subset of contiguous values from T of length m starting at position i ; formally, $T_{i,m} = [t_i, t_{i+1}, \dots, t_{i+m-1}]$. Distribution-based (non-sliding-window) algorithms output similarity scores of each periodic subsequence $T_{im+1,m}$, where $i = 0, \dots, s-1$, where $s = \lfloor n/m \rfloor$. Then AUC can be calculated. While the sliding window method using window size ω produces a total of $n - \omega + 1$ subsequences from T . When calculating AUC of sliding-window-based algorithms, scores of the sliding subsequences are transformed into periodic subsequence scores as follows: Let S_i be the anomaly score of subsequence $T_{i,\omega}$, where $1 \leq i \leq n - \omega + 1$. The score corresponds to a periodic subsequence $T_{im+1,m}$ is $\max(S_{\max(1, im+1-zone)}, \dots, S_{\min(i*m+m+zone-\omega, n-\omega+1)})$, where $zone = \lfloor \omega/2 \rfloor$, $i = 0, \dots, s-1$, where $s = \lfloor n/m \rfloor$. It is reasonable to pick the maximum value from the anomaly scores of relevant overlapping windows and let it be the anomaly score for each period, since we always focus on the peak values of these anomaly scores and suppose there are anomalies at those locations.