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[9]. Real datasets includes MIT-BIH Supraventricular Arrhythmia Database (MBA)[7, 13] and oher datasets which are from various domains and have been studied in earlier works [9, 10, 23].

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 trival, 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[9, 10, 23]. Details can be seen in the document "parameter.csv".

[**Algorithms**] The STOMP [32] implementation of MP is used; NormA is from http://helios.mi.parisdescartes.fr/ themisp/norma/; IDK-based detectors are our implementations based on [28]; 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) ±diff(**TS**)
(ii) ±movmax(**TS**, $\boldsymbol{\omega}$)
(iii) ±movmin(**TS**, $\boldsymbol{\omega}$)
(iv) ±movmean(**TS**, $\boldsymbol{\omega}$)
(v) ±movstd(**TS**, $\boldsymbol{\omega}$)

where **TS** is the time series, $\boldsymbol{\omega}$ is the window size to compute $\boldsymbol{\omega}$-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 $T = [t_1, t_2, .., t_n]$ 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, $T_{i,m} = [t_i, t_{i+1}, .., t_{i+m-1}]$. Distribution-based (non-sliding-window) algorithms output similarity scores of each periodic subsequence $T_{1+(i-1)*m,m}$, where $i = 1, .., \lfloor n/m \rfloor$. 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 $T_{1+(i-1)*m,m}$ is $\max(S_{\max(1,1+(i-1)*m-zone)}, \ldots, S_{\min(i*m+zone-\omega,n-\omega+1)})$, where $zone = \lfloor \omega/2 \rfloor$, where $i = 1, .., \lfloor n/m \rfloor$. 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.