Robust Autocorrelation for Period Detection in Time Series

Zhi Yang, Likun Hou and Xing Zhao
SAP HANA, SAP Labs China, 1001 Chenghui Road, Shanghai, China
$\{$*zhi.yang, likun.hou, brook.zhao*$\}$*@sap.com*

Keywords: Robust Autocorrelation, MAD Filter, Period Detection, Time Series, Outlier.

Abstract: Autocorrelation is a key tool in time series period detection, but its sensitivity to outliers is a significant limitation. This paper introduces a robust autocorrelation method for period detection that minimizes the influence of outliers. By incorporating a moving average and applying a Median Absolute Deviation (MAD) filter to each cycle-subseries, we significantly enhance the robustness of the autocorrelation to outliers. The MAD filter identifies and corrects outliers in the cycle-subseries, based on the assumption that the cycle-subseries consists of a constant plus Gaussian noise. This innovative robust autocorrelation can effectively replace traditional autocorrelation in existing period detection algorithms. Additionally, we propose a new algorithm that leverages our robust autocorrelation. Both theoretical analysis and empirical tests on real-world and synthetic datasets indicate that period detection algorithms using our proposed robust autocorrelation outperform those using traditional autocorrelation. Furthermore, our proposed algorithm surpasses all other existing algorithms in comparison.

1 INTRODUCTION

A time series is considered periodic if a certain pattern repeats at regular intervals of time. Periodicity can occur due to natural phenomena, such as changes in temperature or seasonal fluctuations in product sales, or due to human activities, such as electricity usage or traffic flows. The causes of periodicity are often clear, and the periods are typically daily, weekly or yearly. However, there are also many time series that are related to other phenomena, and detecting periodicity can be challenging, particularly when there is limited information available. Nevertheless, the ability to identify the period of a time series is essential for data analysis and forecasting.

In real-world datasets, outliers are a common occurrence. In time series data, an outlier is a data point that deviates significantly from the general behavior of the remaining data points. Outliers can have various causes, including data entry errors, experimental errors, sampling errors, and natural outliers. They can significantly impact the results of data analysis and forecasting, such as period detection. Outliers can affect traditional autocorrelation and, therefore, influence the detection of periods. Inaccurate period detection of time series can result from outliers.

The existing period detection algorithms of time series can be classified into three groups:1) frequency domain methods that rely on the periodogram after Fourier transform, such as Hyndman's *findfrequency* (Hyndman, 2023), Fisher's test (Fisher, 1929; Wichert et al., 2004) and Lomb-Scargle (Hu et al., 2014; Glynn et al., 2006; Lomb, 1976); 2) time domain methods that rely on autocorrelation function (ACF), such as seasonality test in Predictive Analysis Library (PAL) of SAP HANA (SAP, 2024), median distance of autocorrelation function peaks (ACF-Med) as well as methods proposed in (Wang et al., 2005; Breitenbach et al., 2023); 3) methods that combine periodogram and autocorrelation, such as AUTOPERIOD (Vlachos et al., 2005; Puech et al., 2020), SAZED (Toller et al., 2019) and methods proposed in (Parthasarathy et al., 2006; Wen et al., 2023; Wen et al., 2021). However, most of these algorithms are not robust to outliers. The algorithms presented in (Wen et al., 2023; Wen et al., 2021) claim to be robust to outliers, and we will include their results in the comparison.

In this work, we propose a novel autocorrelation method that demonstrates high robustness to outliers, thereby enhancing the reliability of period detection algorithms. For every lag h, we detrend the time series using a moving avarage with window size h, followed by the application of a Median Absolute Deviation (MAD) filter on each cycle-subseries to identify and correct outliers. The MAD filter, a non-local filter,

Yang, Z., Hou, L. and Zhao, X.
Robust Autocorrelation for Period Detection in Time Series.
DOI: 10.5220/0013092500003890

In *Proceedings of the 17th International Conference on Agents and Artificial Intelligence (ICAART 2025) - Volume 2*, pages 36-44
ISBN: 978-989-758-737-5; ISSN: 2184-433X

detects and adjusts datapoints in the cycle-subseries that fall outside the range of 3 MAD. Our proposed autocorrelation method can effectively replace traditional autocorrelation techniques used in period detection algorithms, significantly enhancing their resilience to outliers. This improvement benefits algorithms such as ACF-Med, AUTOPERIOD, SAZED, and PAL's seasonality test. Furthermore, we present a novel algorithm that utilizes the proposed autocorrelation, showcasing superior performance compared to the aforementioned algorithms. Our approach involves performing peak analysis on the proposed autocorrelation. Although the source code for the algorithms introduced in (Wen et al., 2023; Wen et al., 2021) is not publicly available, our new algorithm achieves better results than those reported in (Wen et al., 2023; Wen et al., 2021) when evaluated on the CRAN dataset (Hyndman and Killick, 2023), despite the previous algorithms also claiming robustness to outliers.

Robust period detection plays a vital role in many real-world applications. One such application is the decomposition of time series with unknown periods using robust methods like STL (Cleveland et al., 1990), where robust period detection is essential. Additionally, in the context of time series outlier detection discussed in (Gao et al., 2020), robust period detection serves as an indispensable step. Numerous other applications that involve robust period detection can be found in (Tolas et al., 2021; Zhang et al., 2022; Wang et al., 2006; Vlachos et al., 2004).

The remainder of this paper is organized as follows: Section 2 delves into a detailed discussion of the proposed autocorrelation and provides an illustrative example. In Section 3, we present the experimental results obtained from public algorithms, where traditional autocorrelation has been substituted with proposed autocorrelation. Section 4 introduces our novel period detection algorithm and discusses the corresponding experiments and ablation studies. Finally, our conclusions are presented in Section 5.

To facilitate readers of interest, we summarize a list of abbreviations which are frequently used in this paper in Table 1.

2 PROPOSED AUTOCORRELATION

2.1 Proposed ACF Formula

Autocorrelation is a fundamental tool used to detect the period of a time series. Given a time series $\boldsymbol{x} = \{x_i | i = 0, 1, \cdots, N-1\}$ with N data points, the autocorrelation function (ACF) of lag h is defined as:

Table 1: List of Abbreviations.

Abbreviation	Description
M-MA ACF	ACF of time series detrended by moving average and filtered by MAD filter on every cycle-subseries.
MA ACF	ACF of time series detrended by moving average
M-RAW ACF	ACF of original time series filtered by MAD filter on every cycle-subseries.
RAW ACF	ACF of original time series
LPA	left peak analysis
ACF-LPA	left peak analysis on ACF
X ACF-LPA	left peak analysis on X ACF. X can be M-MA, MA, M-RAW and RAW.

$$\gamma_h = c_h / c_0, \tag{1}$$

where c_h is the autocovariance function at lag h. The definition of c_h is

$$c_h = \frac{1}{N} \sum_{t=0}^{N-h-1} (x_t - \bar{x})(x_{t+h} - \bar{x}), \tag{2}$$

where $\bar{x}$ is the mean of the time series. The value of γ_h is in the range of -1 to 1, with a larger value indicating more relevance. For a time series with N elements, γ_h is defined with $h = 0, 1, \cdots, N-1$.

To address the limitation of the above traditional ACF's sensitivity to outliers, we propose a robust ACF, as illustrated in Figure 1.

Figure 1: The process for computing proposed ACF.

The first step involves obtaining the detrended time series through a moving average. Similar to the weighted moving average discussed in (Hyndman, 2018) and the moving average used in PAL's seasonality test (SAP, 2024), the trend time series $\boldsymbol{m^h}$ is determined by taking the moving average of window size h for a given time series $\boldsymbol{x}$ and lag h, as shown in (3) and (4),

$$m_i^h = \begin{cases} \dfrac{1}{2q+1} \displaystyle\sum_{j=-q}^{q} x_{i+j}, & h = 2q+1 \\ \dfrac{1}{2q} X_i, & h = 2q \end{cases} \tag{3}$$

$$X_i = 0.5 x_{i-q} + \sum_{j=-q+1}^{q-1} x_{i+j} + 0.5 x_{i+q} \tag{4}$$

where $i = 0, 1, \cdots, N-1$. In (3) and (4), if $i - q < 0$ or $i + j < 0$, x_{i-q} or x_{i+j} is replaced by x_0. Conversely, if $i + q > N-1$ or $i + j > N-1$, x_{i+q} or x_{i+j} is replaced by x_{N-1}. This ensures the trend time series $\boldsymbol{m^h}$ retains the length N. The detrended time series $\boldsymbol{y^h}$ is then obtained by $\boldsymbol{y^h} = \boldsymbol{x} - \boldsymbol{m^h}$.

It is important to note that the window size must be the same as the lag at this step. When the actual period is the same as the lag h in this case, the periodicity is reinforced after removing the trend using the moving average. However, if h is smaller than the actual period, the periodicity of the detrended time series is reduced, resulting in a low ACF for lag h. When h is a little larger than the actual period, the seasonal part is also emphasized in the detrended time series. Nonetheless, the ACF for lag h is lower than the ACF of the actual period due to the ACF formula (1) and (2). As the lag h increases, more of the trend component is included in the detrended time series.

The second step involves the cycle-subseries MAD filter in the detrended time series $\boldsymbol{y^h}$. The MAD filter is a non-local filter based on MAD method. It detects and adjusts datapoints in the cycle-subseries that fall outside the range of 3 MAD. It is inspired by the cycle-subseries smoothing and filtering procedures in STL decomposition (Cleveland et al., 1990) and the non-local means algorithm used in image denoising (Buades et al., 2005). For each lag h, we generate h cycle-subseries. Given the i-th cycle-subseries $\boldsymbol{z}^{i,h} = \{y_{i+kh} | k \geq 0, i + kh < N\}$, we denote $z_k^{i,h} = y_{i+kh}$. Let

$$\hat{z}^{i,h} = \text{median}(\boldsymbol{z}^{i,h}) \tag{5}$$

$$d_k^{i,h} = |z_k^{i,h} - \hat{z}^{i,h}|, \tag{6}$$

$$mad^{i,h} = \text{median}(\boldsymbol{d}^{i,h}). \tag{7}$$

Then, we calculate the MAD score of each point in $\boldsymbol{z}^{i,h}$. The MAD score is defined as follows:

$$s_k^{i,h} = 0.67449 \cdot \frac{z_k^{i,h} - \hat{z}^{i,h}}{mad^{i,h}}. \tag{8}$$

The coefficient 0.67449 is used to make the MAD score equivalent to the Z score when the data follows a Gaussian distribution. After applying the MAD filter, the filtered cycle-subseries is as follows:

$$\tilde{z}_k^{i,h} = \begin{cases} \hat{z}^{i,h} - 3/0.67449 \cdot mad^{i,h}, & s_k^{i,h} < -3 \\ z_k^{i,h}, & -3 \leq s_k^{i,h} \leq 3 \\ \hat{z}^{i,h} + 3/0.67449 \cdot mad^{i,h}, & s_k^{i,h} > 3 \end{cases} \tag{9}$$

Let us put the cycle-subseries back to the whole time series. After applying the cycle-subseries MAD filter to all cycle-subseries, we get the filtered whole time series $\tilde{\boldsymbol{y}}^{\boldsymbol{h}}$ as $\tilde{y}_{i+kh}^h = \tilde{z}_k^{i,h}$.

This approach involves outlier detection using the MAD score, detecting and correcting outliers according to the 3 MAD rule. We assume that the seasonal component satisfies $S_i = S_{i+T}$. When the lag h is the actual period or the integer multiples of the actual period, the cycle-subseries satisfy $z_k^{i,h} = a^{i,h} + \varepsilon_k$, where ε_k follows a Gaussian distribution with zero mean, and $a^{i,h}$ is a constant. In this scenario, the MAD filter can accurately detect outliers within the cycle-subseries. The 3 MAD rule effectively detects and corrects these outliers, thereby making the ACF robust to outliers.

When the lag h is neither the actual period nor an integer multiple of the actual period, the cycle-subseries do not satisfy $z_k^{i,h} = a^{i,h} + \varepsilon_k$. Consequently, the effectiveness of outlier detection and correction is diminished compared to the previous case. This results in a smaller increase in the ACF than when the lag h aligns with the actual period or its integer multiples.

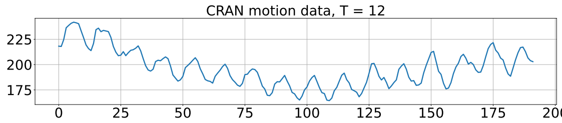
The final step involves calculating the proposed ACF using the time series after detrending and applying the cycle-subseries MAD filter, denoted as $\tilde{\boldsymbol{y}}^{\boldsymbol{h}}$. Similar to the traditional ACF, the proposed ACF at lag h is defined as $\tilde{\gamma}_h = \tilde{c}_h / \tilde{c}_0$, where $\tilde{c}_h$ is

$$\tilde{c}_h = \frac{1}{N} \sum_{i=0}^{N-h-1} (\tilde{y}_i^h - \overline{\tilde{y}^h})(\tilde{y}_{i+h}^h - \overline{\tilde{y}^h}), \tag{10}$$

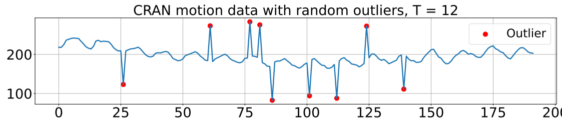
where $\overline{\tilde{y}^h}$ is the mean of the detrended time series $\tilde{\boldsymbol{y}}^{\boldsymbol{h}}$. The complexity of calculating the proposed ACF for each lag is $O(N)$, leading to a total complexity of $O(N^2)$ if the number of lags for autocorrelation scales with N. It's worth noting that the proposed ACFs across different lags can be computed in parallel.

2.2 An Illustrative Example

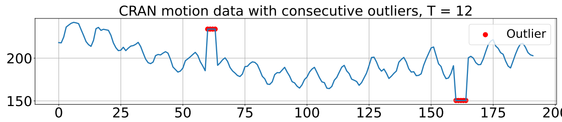
In this subsection, we will evaluate the proposed robust ACF with cycle-subseries MAD filter using a simple case. To assess its robustness to outliers and perform an ablation study, we construct four types of ACFs based on the procedures outlined in Figure 1. The proposed robust ACF depicted there is referred to as M-MA ACF. If we omit the detrending process and apply the cycle-subseries MAD filter on the original time series, this version of ACF is termed M-RAW ACF. When MAD filter is removed and the ACF is calculated from the detrended time series, it is called MA ACF. Finally, if both detrending and MAD filter are removed, and the ACF is computed using the original time series, we refer to it as RAW ACF. For our analysis, we employ motion data from CRAN dataset (Hyndman and Killick, 2023), which is monthly and has a period of 12, with a total length of 192. This data is illustrated in Figure 2a. The four ACFs of the motion data, with lags ranging from 2 to 96, are presented in Figure 3a. It is evident that both M-MA ACF and MA ACF exhibit a strong periodicity of 12. M-RAW ACF and RAW ACF, however, show a declining trend and reach maximum ACF at lag = 2, rather than 12. This occurs due to the presence of a trend in the motion data and the absence of the detrending process in M-RAW ACF and RAW ACF.



(a) CRAN motion data, T = 12

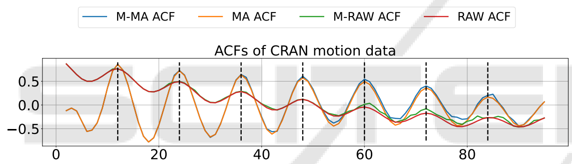


(b) CRAN motion data with 9 random outliers, T = 12

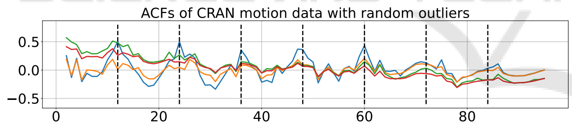


(c) CRAN motion data with 9 consecutive outliers, T = 12

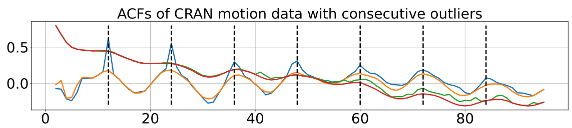
Figure 2: CRAN motion data without outlier, with random outliers and with consecutive outliers.



(a) ACFs of CRAN motion data without outlier



(b) ACFs of CRAN motion data with 9 random outliers



(c) ACFs of CRAN motion data with 9 consecutive outliers

Figure 3: ACFs of CRAN motion data without outlier, with random outliers and with consecutive outliers.

Now let us introduce random outliers into the motion data. We insert 9 outliers with an amplitude five times the standard deviation of the original time series. The corresponding outlier ratio is 0.05. The outlier-infused motion data is displayed in Figure 2b. The four ACFs of outlier-infused motion data are depicted in Figure 3b. It's evident that the M-MA ACF retains a strong periodicity of 12. For other ACFs, the periodicity of 12 is not obvious.

One might question the apparent outliers in Figure 2b, suggesting that local smoothing or range detection algorithms could easily identify them. Let's now consider consecutive outliers in the motion data, which prove more challenging to detect. Figure 2c shows motion data with these consecutive outliers, nine in total, divided into two sections. The four ACFs of motion data with consecutive outliers are depicted in Figure 3c. The periodicity of 12 remains evident in the M-MA ACF and MA ACF. However, the values of the MA ACF are significantly reduced, peaking at less than 0.2, with the maximum value taken at a lag of 24 rather than 12. This periodicity of 12 becomes less clear in the M-RAW ACF and RAW ACF.

Let's now compare the M-MA ACFs of motion data with and without outliers, as depicted in Figure 4. We observe that all instances exhibit a strong periodicity of 12, with comparable peak values. This analysis demonstrates the robustness of M-MA ACFs to outliers, including random, consecutive, and hard-to-detect outliers.

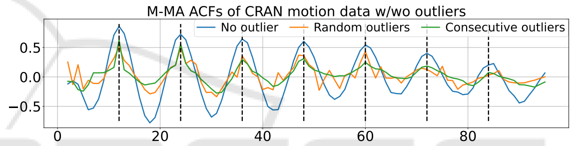


Figure 4: M-MA ACF of CRAN motion data without outlier, with random outliers and with consecutive outliers.

3 EXPERIMENTS ON PUBLIC ALGORITHMS WITH PROPOSED ACF

3.1 Algorithms and Datasets

We explore period detection algorithms utilizing ACF, substituting it with the four ACFs detailed in Section 2.2. These algorithms include PAL's seasonality test, ACF-Med, AUTOPERIOD, and SAZED. The original ACF in SAZED is a 10-fold autocorrelation, as outlined in (Toller et al., 2020).

PAL's seasonality test computes the ACF from lag = 2 up to half the series length, selecting the maximum ACF value. If this maximum ACF exceeds the threshold (default is 0.2, which we use in subsequent sections), the corresponding lag is considered as the period. If not, the series is deemed non-periodic. We utilize the AUTOPERIOD implementation from (Schmidl, 2023) and the SAZED implementation from (Toller et al., 2020) in R studio.

For our datasets, we use the publicly available, single-period time series data from CRAN, also referenced in (Toller et al., 2019). This data, sourced from the "Time Series Data" section of "CRAN Task View: Time Series Analysis" (Hyndman and Killick, 2023), comprises 82 time series, each labeled with its corresponding period. Series lengths vary from 16 to 3024, with periods ranging from 2 to 52. Figure 2a shows a time series from the CRAN dataset.

In addition, we generate 3000 synthetic time series, each of length 500, without outliers. Each series includes a piecewise linear trend, additive white Gaussian noise (variance = 0.1), and a periodic component with amplitude 1 and period between 10 to 50. We explore three periodic patterns: sinusoidal, square, and triangular waves, with each pattern represented in 1000 series. Figure 5 shows examples of sine wave data, square wave data, and triangle wave data, all without outliers.

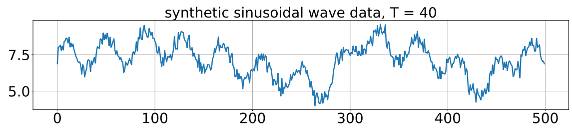
We also introduce outliers to both the CRAN and synthetic datasets, with an amplitude five times the standard deviation of the original time series. In (Wen et al., 2023), the same outlier amplitude was also employed. For most real-world examples, an outlier that exceeds five times the standard deviation is considered sufficiently large. Large outliers have a significant impact on periodicity detection. We aim to evaluate the performance of the proposed ACF in handling data with large outliers, as well as normal data. We set the outlier ratio (OR) at 0.01, 0.03, or 0.05. Like (Wen et al., 2023), since the outlier ratio is usually small in real world, we limit the outlier ratio to a maximum of 0.05. For instance, the motion data depicted in Figure 2b with an OR of 0.05 represents a CRAN time series with outliers. Similarly, Figure 6 illustrates examples of sine wave data, square wave data, and triangle wave data, each containing 5% outliers.

3.2 Experimental Results

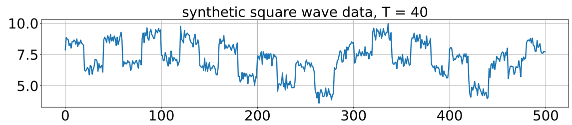
Similar to (Wen et al., 2023; Wen et al., 2021), the precision of every algorithm is calculated by the ratio of the number of time series with correctly estimated period length to the total number of time series. Table 2 presents results from various algorithms based on ACFs, applied to the CRAN dataset. For the unmodified CRAN dataset, M-MA ACF and MA ACF demonstrate comparable performance. However, in the presence of outliers, M-MA ACF surpasses other ACFs in algorithms PAL's seasonality test, ACF-Med, and SAZED. For AUTOPERIOD, the performances of the four ACFs are similar, due to the reliance on period hints provided by the periodogram method. This method, however, underperforms due to the complex trend in the CRAN dataset.

For the synthetic datasets, the findings are outlined in Table 3, mirroring those from the CRAN dataset. Without outliers, M-MA ACF and MA ACF show similar performance. However, in the presence of outliers, M-MA ACF surpasses other ACFs across all four algorithms, including AUTOPERIOD. This superior performance is attributed to the piecewise linear trend in synthetic data, which is simpler than CRAN, enabling the periodogram method in AUTOPERIOD to perform better.

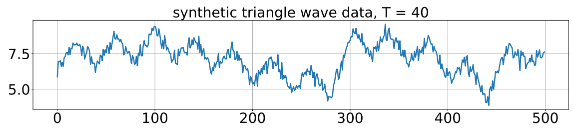
From our experiments on the CRAN and synthetic datasets, we observed superior or equivalent performance of M-MA ACF compared to other ACFs across all tested algorithms. M-MA ACF significantly out-



(a) An example of synthetic sinusoidal wave data, T = 40

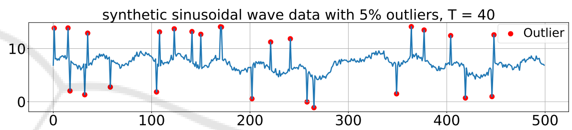


(b) An example of synthetic square wave data, T = 40

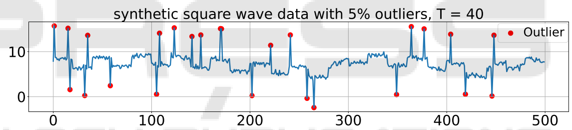


(c) An example of synthetic triangle wave data, T = 40

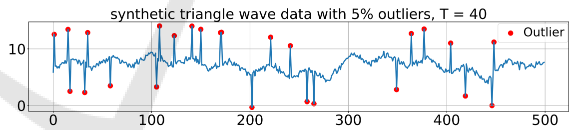
Figure 5: Examples of synthetic wave data, from top to bottom are sinusoidal wave, square wave, and triangle wave.



(a) An example of synthetic sinusoidal wave data with 5% outliers, T = 40



(b) An example of synthetic square wave data with 5% outliers, T = 40



(c) An example of synthetic triangle wave data with 5% outliers, T = 40

Figure 6: Examples of synthetic wave data with with 5% outliers, from top to bottom are sinusoidal wave, square wave, and triangle wave.

Table 2: Precision of public algorithms on CRAN dataset. The best results among ACFs are highlighted in bold.

Algorithms	ACFs	OR = 0	OR = 0.01	OR = 0.03	OR = 0.05
PAL	RAW	0.415	0.390	0.402	0.232
	M-RAW	0.402	0.402	0.378	0.341
	MA	0.768	0.537	0.329	0.195
	M-MA	**0.805**	**0.793**	**0.634**	**0.500**
ACF-Med	RAW	0.451	0.402	0.183	0.012
	M-RAW	0.451	0.366	0.317	0.220
	MA	**0.671**	0.354	0.085	0.012
	M-MA	**0.671**	**0.646**	**0.524**	**0.402**
AUTOPERIOD	RAW	**0.451**	0.366	0.293	0.207
	M-RAW	0.415	0.354	0.305	0.220
	MA	0.439	0.354	0.305	0.183
	M-MA	**0.451**	**0.390**	**0.317**	**0.256**
SAZED	10-fold	0.549	0.500	0.439	0.415
	RAW	0.524	0.512	0.463	0.390
	M-RAW	0.524	0.488	**0.500**	0.427
	MA	**0.598**	**0.537**	0.476	0.439
	M-MA	0.585	**0.537**	0.488	**0.488**

performs other ACFs in the presence of outliers, with its advantage becoming increasingly apparent as the number of outliers increases. M-MA ACF can enhance the performance of existing algorithms.

Based on our experiments, we observed that the PAL with M-MA ACF performs exceptionally well when the outlier ratio is either 0 or 0.01. However, its performance degrades at outlier ratios of 0.03 or 0.05. We also noticed that the PAL with M-MA ACF detects periods in the time series as integer multiples of the actual period in certain instances. For instance, a time series with a period of 12 might be identified as having a period of 24. In the case of the CRAN dataset, if we accept these integer multiples as correct detections, the precision is 0.841, 0.854, 0.744, and 0.695 for outlier ratios of 0, 0.01, 0.03, and 0.05, respectively. We will leverage this property to propose a new algorithm in the following section.

4 PROPOSED PERIOD DETECTION ALGORITHM

4.1 Algorithm Description

In PAL's seasonality test, the period is identified as the lag with the highest ACF. However, the lag with the maximum ACF doesn't always denote the period. As discussed in Section 3.2, it could also represent integer multiples of the period, particularly when outliers are present, and M-MA ACF is applied. In our new algorithm, we locate the maximum M-MA ACF of the time series at lag T_{raw}, and conduct peak analysis on ACFs with lags smaller than T_{raw}. Graphically, these ACFs are to the left of the maximum ACF. We term this process as left peak analysis and our new method as M-MA ACF-LPA. In the left peak analysis, we consider all integer factors of T_{raw}, excluding 1, and analyze them in ascending order. If an integer factor and its multiples correspond to ACF peaks and are greater than a certain threshold, that integer factor is identified as the period.

The M-MA ACF-LPA procedure is detailed in Algorithm 1. We compute M-MA ACFs for lags starting from 2 and identify the maximum ACF and its corresponding lag. Then, we sort all possible periods in ascending order and validate them sequentially. Once a possible period is validated, we deem it as the period of the input time series and conclude the algorithm. In Algorithm 1, we use 0.2 as the ACF threshold to determine the presence of a periodic pattern. This threshold value is consistent with the threshold used in our previously discussed PAL's seasonality test. The higher this value, the stricter the criteria for determining the presence of a periodic pattern. In the left peak analysis, we use 0.7 times the maximum ACF value as a threshold. This is because the ACF of the true period and its integer multiples are both peaks and are relatively large. It is reasonable to select a higher value as the threshold. If the threshold is too low, there is a higher risk of misidentifying periodicity, while a threshold that is too high may cause us to overlook the true period. In the following experiments, we utilize the threshold values specified in Algorithm 1.

4.2 Experiments

The new algorithm, M-MA ACF-LPA, allows for interchangeable ACF components, including MA ACF, M-RAW ACF, and RAW ACF. These modifications yield respective variants: MA ACF-LPA, M-RAW ACF-LPA, and RAW ACF-LPA, which are collectively referred to as ACF-LPA algorithms. This subsection presents an experimental comparison of these four ACF-LPA variants, effectively serving as an ablation study for the ACF component. By omitting the left peak analysis, we get PAL's seasonality test, which serves as an ablation study for the LPA element. The performance of our algorithm will be benchmarked against other widely used algorithms such as Fisher's Test, Lomb–Scargle Periodogram, and *findfrequency*. For *findfrequency*, we use the implementation (Hyndman, 2023) in R studio. Our datasets include the real-world CRAN dataset and the synthetic datasets detailed in Section 3.

The CRAN dataset results are shown in Table 4. When juxtaposed with the results in Table 2, it's evident that M-MA ACF-LPA matches the performance of PAL with M-MA ACF, and surpasses all other algorithms when the outlier ratio is 0 or 0.01. For out-

Table 3: Precision of public algorithms on synthetic datasets. The best results among ACFs are highlighted in bold.

Algorithms	ACFs	Synthetic Sinusoidal Time Series				Synthetic Square Time Series				Synthetic Triangle Time Series			
		OR = 0	OR = 0.01	OR = 0.03	OR = 0.05	OR = 0	OR = 0.01	OR = 0.03	OR = 0.05	OR = 0	OR = 0.01	OR = 0.03	OR = 0.05
PAL	RAW	0.091	0.088	0.066	0.058	0.231	0.224	0.160	0.135	0.070	0.075	0.057	0.045
	M-RAW	0.096	0.120	0.164	0.185	0.245	0.281	0.314	0.338	0.074	0.096	0.113	0.127
	MA	0.778	0.263	0.091	0.048	**0.978**	0.715	0.268	0.126	0.714	0.222	0.064	0.029
	M-MA	**0.880**	**0.891**	**0.722**	**0.576**	0.954	**0.875**	**0.763**	**0.657**	**0.836**	**0.838**	**0.650**	**0.462**
ACF-Med	RAW	0.378	0.159	0.031	0	0.780	0.410	0.068	0	0.318	0.104	0.022	0
	M-RAW	0.338	0.164	0.151	0.168	0.719	0.428	0.317	0.315	0.269	0.126	0.111	0.130
	MA	0.722	0.065	0.003	0	0.970	0.250	0.009	0	0.632	0.047	0.001	0
	M-MA	**0.758**	**0.744**	**0.706**	**0.457**	**0.975**	**0.879**	**0.877**	**0.809**	**0.706**	**0.702**	**0.496**	**0.233**
AUTOPERIOD	RAW	0.273	0.239	0.148	0.094	**0.792**	0.668	0.373	0.245	0.236	0.188	0.104	0.065
	M-RAW	0.302	0.332	0.325	0.297	0.748	0.634	0.539	0.479	0.255	0.272	0.237	0.213
	MA	0.548	0.263	0.141	0.098	0.782	0.659	0.351	0.237	0.426	0.222	0.100	0.066
	M-MA	**0.607**	**0.641**	**0.568**	**0.496**	0.787	**0.774**	**0.725**	**0.686**	**0.476**	**0.493**	**0.410**	**0.338**
SAZED	10-fold	0.273	0.239	0.148	0.094	**0.792**	**0.668**	0.373	0.245	0.236	0.188	0.104	0.065
	RAW	0.353	0.319	0.249	0.253	0.440	0.393	0.331	0.306	0.303	0.261	0.217	0.212
	M-RAW	0.356	0.328	0.261	0.266	0.444	0.409	0.326	0.320	0.295	0.260	0.214	0.210
	MA	**0.703**	0.560	0.436	0.406	0.727	0.647	0.514	0.488	0.612	0.478	0.355	0.336
	M-MA	0.683	**0.607**	**0.498**	**0.503**	0.686	0.651	**0.533**	**0.541**	**0.619**	**0.526**	**0.464**	**0.435**

Data: time series $\boldsymbol{x} = \{x_i | i = 0, 1, \cdots, N-1\}$
Result: period
Calculate M-MA ACFs of lags from 2 to $N/2$;
Take maximal ACF as acf_max and its corresponding lag as period_raw;
if *acf_max < 0.2* **then**
    return no period;
end
possible_periods = all factors of period_raw except 1, in ascending sort. For example, if period_raw = 12, possible_periods = [2,3,4,6,12];
for *pperiod in possible_periods* **do**
    pperiods = all integer multiple of pperiod and no larger than period_raw. For example, if pperiod = 3, pperiods = [3,6,9,12];
    if *pperiods are all ACF peaks and their coressponding ACFs are all larger than 0.7 · acf_max* **then**
        return pperiod;
    end
end
return period_raw;

Algorithm 1: Process of M-MA ACF-LPA.

lier ratios of 0.03 or 0.05, M-MA ACF-LPA continues to outperform all competitors.

We also benchmarked our results against the methods in (Wen et al., 2023; Wen et al., 2021). However, as their source code was unavailable, we could only compare our algorithms with the results published in their papers. Their precision in the CRAN dataset ranges from 0.60 to 0.63, and our precision is 0.805. Our method significantly improves on these figures. Regarding the CRAN dataset with outliers, we don't have their specific data but replicated outliers in the same manner. Their precision peaks at 0.62 and 0.60 for outlier ratios of 0.01 and 0.05. Our M-MA ACF-LPA, however, achieves accuracy figures of 0.793 and 0.622 for the same outlier ratios, outshining their results across all outlier conditions.

For the ACF component's ablation study, we substituted M-MA ACF with MA-ACF, M-RAW ACF, and RAW ACF. Without outliers, M-MA ACF performs slightly worse than MA-ACF but remains competitive. In the presence of outliers, M-MA ACF significantly outperforms other ACFs, especially as the outlier ratio increases. In the left peak analysis ablation study, we contrasted our algorithm with PAL's seasonality using M-MA ACF. With outlier ratios of 0 or 0.01, our algorithm marginally underperforms PAL, yet remains comparable. However, with outlier ratios of 0.03 or 0.05, our algorithm clearly surpasses PAL.

Table 4: Precision of ACF-LPA and other algorithms on CRAN dataset. The best results are highlighted in bold.

Algorithms	ACFs	OR = 0	OR = 0.01	OR = 0.03	OR = 0.05
ACF-LPA	RAW	0.415	0.402	0.402	0.268
	M-RAW	0.402	0.402	0.415	0.341
	MA	0.768	0.561	0.354	0.195
	M-MA	**0.805**	**0.793**	**0.659**	**0.622**
PAL	M-MA	**0.805**	**0.793**	0.634	0.500
Fisher's Test		0.390	0.366	0.293	0.220
Lomb–Scargle		0.549	0.537	0.488	0.402
findfrequency		0.451	0.378	0.268	0.171

The results for synthetic datasets, presented in Table 5, mirror those of the CRAN dataset. When the outlier ratio is zero, ACF-LPA employing M-MA ACF yields results comparable to PAL's seasonality test with M-MA ACF or MA ACF, and surpasses other algorithms. In the presence of outliers, our algorithm exceeds the performance of all others.

In the ACF component ablation study, without

Table 5: Precision of ACF-LPA and other algorithms on synthetic datasets. The best results are highlighted in bold.

Algorithms	ACFs	Synthetic Sinusoidal Time Series				Synthetic Square Time Series				Synthetic Triangle Time Series			
		OR = 0	OR = 0.01	OR = 0.03	OR = 0.05	OR = 0	OR = 0.01	OR = 0.03	OR = 0.05	OR = 0	OR = 0.01	OR = 0.03	OR = 0.05
ACF-LPA	RAW	0.091	0.090	0.068	0.062	0.235	0.233	0.178	0.153	0.071	0.077	0.059	0.047
	M-RAW	0.096	0.120	0.164	0.182	0.250	0.288	0.322	0.343	0.075	0.097	0.113	0.134
	MA	0.778	0.285	0.102	0.057	**1.000**	0.804	0.319	0.171	0.715	0.239	0.074	0.036
	M-MA	**0.881**	**0.935**	**0.922**	**0.862**	0.998	**0.993**	**0.992**	**0.985**	**0.841**	**0.903**	**0.852**	**0.724**
PAL	M-MA	0.880	0.891	0.722	0.576	0.954	0.875	0.763	0.657	0.836	0.838	0.650	0.462
Fisher's Test		0.212	0.217	0.207	0.208	0.267	0.273	0.269	0.265	0.172	0.173	0.171	0.179
Lomb–Scargle		0.483	0.480	0.462	0.477	0.646	0.638	0.606	0.609	0.391	0.393	0.382	0.375
findfrequency		0.431	0.356	0.287	0.216	0.516	0.454	0.365	0.287	0.351	0.291	0.200	0.168

outliers, M-MA ACF performance slightly trails MA ACF in synthetic square wave datasets, though it remains competitive. However, for synthetic sinusoidal and triangle wave datasets, M-MA ACF performs best. When outliers are present, especially in larger ratios, M-MA ACF significantly outperforms other ACFs in accuracy. In the left peak analysis ablation study, we juxtapose our algorithm with PAL's seasonality using M-MA ACF. Without outliers, our algorithm holds its own against PAL. However, when outliers are present, our algorithm outpaces PAL, with the advantage becoming more pronounced as the number of outliers increases.

The experimental results from both the CRAN and synthetic datasets demonstrate that the M-MA ACF-LPA algorithm outperforms or matches other algorithms when no outliers are present. Furthermore, the presence of outliers significantly enhances the superiority of the M-MA ACF-LPA, with its advantage becoming increasingly evident as the number of outliers increases.

5 CONCLUSIONS

In this paper, we present a novel robust autocorrelation function, M-MA ACF, designed for period detection in time series data. This function, derived from moving average and applying MAD filter to every cycle-subseries, exhibits robustness against both isolated and consecutive outliers. This robustness is substantiated through theoretical analysis and empirical testing on both real-world and synthetic datasets. M-MA ACF can enhance the performance of existing algorithms like PAL's seasonality test, ACF-Med, AUTOPERIOD, and SAZED. We also introduce a new algorithm, M-MA ACF-LPA, that builds on M-MA ACF and left peak analysis. Without outliers, the performance of the M-MA ACF-LPA algorithm is on par with or better than other algorithms in comparison. The presence of outliers, however, accentuates its superiority, with its advantage increasing proportional to the number of outliers. Although our proposed ACFs have a complexity of $O(N^2)$, parallel computation can be deployed for process optimization. Besides, our proposed ACF has the potential to improve the accuracy of many existing algorithms, and thus benefit various related applications.

REFERENCES

Breitenbach, T., Wilkusz, B., Rasbach, L., and Jahnke, P. (2023). On a method for detecting periods and repeating patterns in time series data with autocorrelation and function approximation. *Pattern Recognition*, 138:109355.

Buades, A., Coll, B., and Morel, J.-M. (2005). A non-local algorithm for image denoising. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 2, pages 60–65. IEEE.

Cleveland, R. B., Cleveland, W. S., McRae, J. E., and Terpenning, I. (1990). Stl: A seasonal-trend decomposition. *J. Off. Stat*, 6(1):3–73.

Fisher, R. A. (1929). Tests of significance in harmonic analysis. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 125(796):54–59.

Gao, J., Song, X., Wen, Q., Wang, P., Sun, L., and Xu, H. (2020). Robusttad: Robust time series anomaly detection via decomposition and convolutional neural networks. *arXiv preprint arXiv:2002.09545*.

Glynn, E. F., Chen, J., and Mushegian, A. R. (2006). Detecting periodic patterns in unevenly spaced gene expression time series using lomb–scargle periodograms. *Bioinformatics*, 22(3):310–316.

Hu, F., Smeaton, A. F., and Newman, E. (2014). Periodicity detection in lifelog data with missing and irregularly sampled data. In *2014 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 16–23. IEEE.

Hyndman, R. (2018). *Forecasting: principles and practice*. OTexts.

Hyndman, R. J. (2023). findfrequency: Find dominant frequency of a time series.

Hyndman, R. J. and Killick, R. (2023). Cran task view: Time series analysis.

Lomb, N. R. (1976). Least-squares frequency analysis of unequally spaced data. *Astrophysics and space science*, 39:447–462.

Parthasarathy, S., Mehta, S., and Srinivasan, S. (2006). Robust periodicity detection algorithms. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 874–875.

Puech, T., Boussard, M., D'Amato, A., and Millerand, G. (2020). A fully automated periodicity detection in time series. In *Advanced Analytics and Learning on Temporal Data: 4th ECML PKDD Workshop, AALTD 2019, Würzburg, Germany, September 20, 2019, Revised Selected Papers 4*, pages 43–54. Springer.

SAP (2024). Seasonality test in predictive analysis library (pal) of sap hana cloud and database.

Schmidl, S. (2023). Periodicity detection: Detect the dominant period in univariate, equidistant time series data.

Tolas, R., Portase, R., Iosif, A., and Potolea, R. (2021). Periodicity detection algorithm and applications on iot data. In *2021 20th International Symposium on Parallel and Distributed Computing (ISPDC)*, pages 81–88. IEEE.

Toller, M., Santos, T., and Kern, R. (2019). Sazed: parameter-free domain-agnostic season length estimation in time series data. *Data Mining and Knowledge Discovery*, 33(6):1775–1798.

Toller, M., Santos, T., and Kern, R. (2020). sazedr: Parameter-free domain-agnostic season length detection in time series.

Vlachos, M., Meek, C., Vagena, Z., and Gunopulos, D. (2004). Identifying similarities, periodicities and bursts for online search queries. In *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, pages 131–142.

Vlachos, M., Yu, P., and Castelli, V. (2005). On periodicity detection and structural periodic similarity. In *Proceedings of the 2005 SIAM international conference on data mining*, pages 449–460. SIAM.

Wang, J., Chen, T., and Huang, B. (2005). Cyclo-period estimation for discrete-time cyclo-stationary signals. *IEEE transactions on signal processing*, 54(1):83–94.

Wang, X., Smith, K., and Hyndman, R. (2006). Characteristic-based clustering for time series data. *Data mining and knowledge Discovery*, 13:335–364.

Wen, Q., He, K., Sun, L., Zhang, Y., Ke, M., and Xu, H. (2021). Robustperiod: Robust time-frequency mining for multiple periodicity detection. In *Proceedings of the 2021 international conference on management of data*, pages 2328–2337.

Wen, Q., Yang, L., and Sun, L. (2023). Robust dominant periodicity detection for time series with missing data. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Wichert, S., Fokianos, K., and Strimmer, K. (2004). Identifying periodically expressed transcripts in microarray time series data. *Bioinformatics*, 20(1):5–20.

Zhang, C., Zhou, T., Wen, Q., and Sun, L. (2022). Tfad: A decomposition time series anomaly detection architecture with time-frequency analysis. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 2497–2507.