



SAZED: parameter-free domain-agnostic season length estimation in time series data

Maximilian Toller¹ · Tiago Santos² · Roman Kern¹

Received: 26 November 2018 / Accepted: 19 July 2019
© The Author(s) 2019

Abstract

Season length estimation is the task of identifying the number of observations in the dominant repeating pattern of seasonal time series data. As such, it is a common pre-processing task crucial for various downstream applications. Inferring season length from a real-world time series is often challenging due to phenomena such as slightly varying period lengths and noise. These issues may, in turn, lead practitioners to dedicate considerable effort to preprocessing of time series data since existing approaches either require dedicated parameter-tuning or their performance is heavily domain-dependent. Hence, to address these challenges, we propose SAZED: spectral and average autocorrelation zero distance density. SAZED is a versatile ensemble of multiple, specialized time series season length estimation approaches. The combination of various base methods selected with respect to domain-agnostic criteria and a novel seasonality isolation technique, allow a broad applicability to real-world time series of varied properties. Further, SAZED is theoretically grounded and parameter-free, with a computational complexity of $\mathcal{O}(n \log n)$, which makes it applicable in practice. In our experiments, SAZED was statistically significantly better than every other method on at least one dataset. The datasets we used for the evaluation consist of time series data from various real-world domains, sterile synthetic test cases and synthetic data that were designed to be seasonal and yet have no finite statistical moments of any order.

Keywords Time series · Season length estimation · Ensemble methods

Responsible editor: Po-ling Loh, Evimaria Terzi, Antti Ukkonen, Karsten Borgwardt, and Katharina Heinrich.

✉ Tiago Santos
tsantos@iicm.edu

Extended author information available on the last page of the article

Published online: 26 July 2019

Springer

1 Introduction

In present days, an abundance of temporal data, such as time series or data streams, is produced by a vast number of sources, waiting to be analyzed. Seasonality is a common behavior of many time series datasets, which is generally interpreted in literature as a regular pattern occurring at a fixed frequency (Fuller 2009; Hyndman and Athanasopoulos 2018), such as daily, weekly, monthly or yearly. When working with such seasonal patterns, it is of crucial importance to know the number of observations within a seasonal pattern. Many algorithms for mining or forecasting seasonal patterns require this as an input parameter, e.g. function fitting (Jönsson and Eklundh 2002), seasonality clustering (Kumar et al. 2002), decomposition (Cleveland et al. 1990; Wang et al. 2006b), and embeddings (Cerqueira et al. 2017). However, only a small fraction of all time series data is labeled, and often additional work is required to determine the length of seasonal patterns. To that end, an automated approach for detecting the length of seasonal patterns would be beneficial for downstream tasks, such as forecasting, pattern recognition, classification and outlier detection in seasonal time series. Such methods are commonly applied in medicine, econometrics, engineering and many other domains. Consequently, a domain-agnostic approach for automated season length estimation would be of value to practitioners in many fields.

Baseline statistical methods address season length estimation by modeling time series data as cyclo-stationary processes and estimating the period of these processes, such as variability estimation (Wang et al. 2006a) or minimum description length (Ramírez et al. 2014). While such approaches are theoretically grounded, they are difficult to apply in practice since they impose assumptions and restrictions with regard to data or setup. These range from requiring human assistance for season length estimation to needing several time series realizations from the same process. Both of these can be problematic in an automated setting, in which data are available but cannot be generated on demand or ensured to follow the same process.

Machine learning techniques have been researched in the context of season length estimation, as exemplified by Pierson et al. (2018), who trained learning systems such as Hidden Markov Models on a labeled time series dataset and typically optimized them towards a local minimum. Subsequently, the learned decision weights were used to detect the season length of time series instances from a related dataset. While supervised machine learning has yielded outstanding results in this context, it has one inherent prerequisite: it requires a labeled dataset based on which it can learn to solve the estimation problem for the data at hand. Acquiring such datasets is associated with tedious and costly manual labeling and, if done improperly, carries the risk of poor generalization. To this end, it seems advantageous to have a complementary method in one's toolbox for detecting the season length of a phenomenon represented by a single univariate time series.

In this work, we introduce spectral and average autocorrelation zero distance density (SAZED), a parameter-free domain-agnostic ensemble method for season length estimation in a single univariate time series. We formally argue why our method works effectively with univariate seasonal time series and show that it improves

upon the existing baselines by a considerable margin. Additionally, all methods in the ensemble are theoretically grounded, enabling us to arrive at the following contributions:

- No dependency on parameters, empirical constants or domain-specific pre-processing,
- Coverage of a wide variety of time series properties with which the existing methods struggle,
- Worst-case computational complexity of $\mathcal{O}(n \log n)$.

Due to these characteristics our methods can be used in an automated setting. No parameters have to be tuned to achieve an acceptable performance in a wide variety of cases. The computational complexity is reasonably low, making our approach broadly applicable. Moreover, the combination of these properties allows our method to be applied in practical settings. An open-source implementation of our method is available in the R package `sizedR` (Toller et al. 2019) and we provide a script to reproduce our results.¹

2 Background

In the literature, season length estimation is mostly referred to as (segment) periodicity detection, which is the problem of splitting a sequence of observations into segments of equal length while maximizing their similarity. However, as Elfeke et al. (2005a) pointed out, many publications simply use the term periodicity detection, which has many other possible interpretations. For example, symbol periodicity refers to the more specific problem of finding a single particular type of observation that repeats at a specific interval.

More recent examples of detecting the periodicity of a single observation or event include the works by Yuan et al. (2017), Ghosh et al. (2017), and Pierson et al. (2018) and references therein. Yuan et al.'s method is based on an heuristic algorithm for maximizing a period pattern scoring function, which extracts multiple periods from incomplete and noisy data. A key aspect of Yuan et al.'s work, pursued in more detail by Lijffijt et al. (2012), is the extraction of periods at varying granularities. In this work, we focus on estimating a single *dominant* period. Ghosh et al. propose a particle-filter-based approach to estimating a period length and its rate of change. The Hidden Markov Chain-based approach presented by Pierson et al. recovers cycle length in human health-related applications, thereby coping with multivariate cyclic data with continuous and discrete dimensions. The works by these authors address challenges beyond the periodic pattern detection problem, such as shifting and noisy period lengths, online periodicity detection and clustering of time series via period length. While we also address some of the same real-world challenges tackled by these authors, e.g. noise and heterogeneous properties of seasonal data, our approach focuses on segment periodicity time series data rather than on symbol or event periodicity in event streams. An event may be periodic by itself without causing a sequence of observations to be overall periodic. Segment periodicity and in a wider sense sea-

¹ https://github.com/mtoller/autocorr_season_length_detection/blob/master/r/reproduceResults.R.

sonality assumes that the process that generates the time series data contains a singular influential component that causes periodic fluctuations in all observations.

For detecting time series segment periodicity, several different solutions have been proposed in various publications, e.g. a convolution-based algorithm (Elfeky et al. 2005a), suffix-trees (Rasheed et al. 2011) and spectral analysis (Vlachos et al. 2005). While these solutions work well in synthetic settings, their performance often drops in the presence of irregularities, such as noise or outliers, as pointed out by Elfeky et al. (2005b). To address this issue, Elfeky et al. suggested an algorithm based on dynamic time warping, which is resilient to various types of noise. However, all of the above-mentioned approaches still require non-trivial preprocessing steps, such as symbolization, or parametrization before they can work with raw data. This limits their general applicability since it is difficult to avoid overfitting when learning a parametrization from a dataset. Further, a parametrization might impose the user's bias onto the data, which can lead to misinterpretations of the data behavior (Keogh et al. 2004). To address this issue, a few works have focused on developing more robust approaches to segment periodicity detection, e.g. the analysis of autocorrelation peaks (Wang et al. 2006b), spectral density estimation (Hyndman 2012), and autocorrelation zero-distances (Toller and Kern 2017). However, while these approaches are non-parametric and require a small preprocessing effort, they still depend on internal empirical constants, which is essentially equivalent to using one default parameterization for any input. Our approach extends such methods, combining them with additional techniques in an ensemble setting to improve upon them.

3 Problem statement and notation

Seasonality is a behavior that many time series from domains such as econometrics or climatology exhibit.

An example of a seasonal time series is provided in Fig. 1, which features data with seasonal peaks and troughs. Informally, season length can be estimated as the number of observations between such peaks. We formalize this notion mathematically below.

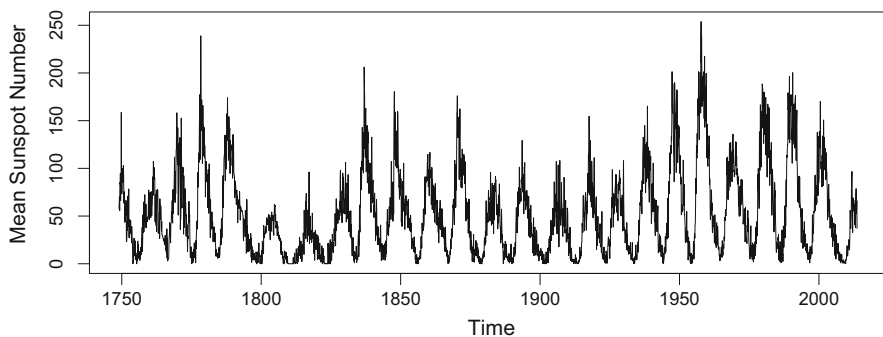


Fig. 1 A seasonal time series of monthly sunspots since 1750 (Andrews and Herzberg 2012). The periodic peaks and troughs correspond to a seasonality with almost constant season length of approximately 132 observations per season

Let $\{X_t | t \in \mathbb{Z}\}$ be a theoretic time series process of the form

$$X_t = S_t + D_t + Y_t \quad (1)$$

where $S_t = S_{t+m}$ is a deterministic seasonality with season length m , D_t a deterministic linear trend and Y_t the stochastic remainder. Further, let $x = \{x_1, x_2, \dots, x_n\}$ be a realization of process X . Season length estimation is the problem of finding $\hat{m}$ such that $S_t = S_{t+\hat{m}}$ given time series data x .

However, this definition is insufficient for practical applications since the theoretic process X that generated x is typically unknown. Therefore, we focus on seasonality of process realizations and define this seasonality as follows:

Definition 1 Time series data $\{x_t | t \in 1, \dots, n\}$ is seasonal if there exists $m \in \mathbb{Z}$ such that $x_t \approx x_{t+km}$ for all $t \in \mathbb{N}$, $k \in \mathbb{Z}$ with $t + km \in [1, n]$.

In other words, x is seasonal if it can be split into segments of length m that are all approximately equal to each other. One can summarize the subsequences of x in a matrix

$$\underline{x}_k := \begin{bmatrix} x^{(1)} \\ x^{(2)} \\ \vdots \\ x^{(k)} \end{bmatrix} := \begin{bmatrix} x_1 & x_2 & \dots & x_m \\ x_{m+1} & x_{m+2} & \dots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{(k-1)m+1} & x_{(k-1)m+2} & \dots & x_{km} \end{bmatrix} \quad (2)$$

for an arbitrary m , with $k = \lfloor \frac{n}{m} \rfloor$. The observations $x_{km+1 \leq t \leq n}$ are discarded and are not part of $\underline{x}_k$ if $n \not\equiv 0 \pmod{m}$.

To specify approximate equality of subsequences, the sample Pearson correlation coefficient ρ can be used. In this case, time series data x is considered seasonal if all subsequences $x^{(i)}$ and $x^{(j)}$ highly correlate with each other. With $R_{\underline{x}_k} \in [-1, 1]^{k \times k}$, we denote the matrix of correlations between all subsequences $x^{(i)}$ and $x^{(j)}$ of x . Based on these definitions, we formalize the season length estimation problem:

Definition 2 (Problem Statement) Given seasonal time series data $\{x_t | t \in 1, \dots, n\}$ and the correlation matrix of its segments $R_{\underline{x}_k}$, find season length $\hat{m}^*$ satisfying

$$\hat{m}^* = \min_m \operatorname{argmax}_m \min_{1 \leq i \leq k, i \leq j \leq k} R_{\underline{x}_k}(i, j). \quad (3)$$

Intuitively, the best season length $\hat{m}^*$ candidate is the one that maximizes the smallest correlation between subsequences of x of length $\hat{m}^*$, and, if there are several values which maximize this quantity, it is the smallest of these values. It is desirable to choose the smallest value, since this ensures that multiples of the correct season length are excluded, although they also maximize the minimum of the correlation matrix.

Table 1 provides a summary of the notation used in this work.

Table 1 Overview of the notation used in this work

Notation	Interpretation
t	Index of time
X	Time series process
S_t	Seasonal component of X
D_t	Trend component of X
x	Time series data, realization of X
n	Length of x
m	Season length of x
$\hat{m}$	Estimate of m
$\hat{m}^*$	Best estimate of m
ρ	Sample Pearson correlation coefficient
$\underline{x}_k$	Matrix of x split into k subsequences
$R_{\underline{x}_k}$	Correlation matrix of $\underline{x}_k$
S_X	Spectral density of X
$\hat{S}_x$	Periodogram of x
$\hat{R}_{xx}$	Sample autocorrelation
Ψ	Pairwise linear interpolation
$\nabla^d[f]$	d^{th} discrete difference of function f
z	Zeros of the autocorrelation function
Z	Zeros of the time series
$\mathcal{F}$	Discrete Fourier transform
$\mathcal{F}^{-1}$	Inverse discrete Fourier transform
$\hat{\mathcal{R}}_{xx}^\ell$	ℓ -fold composition of $\hat{\mathcal{R}}_{xx}$ with itself

4 Base methods

In this section, we address the theoretical foundation of the components of our ensemble method, which are also commonly used baseline algorithms for period length estimation in time series.

4.1 Preprocessing

Commonly, time series data contain a trend. In Eq. (1) this is reflected by the component D_t . Theoretically, D may be an arbitrary function of t , yet in time series analysis it is often assumed that D_t is a linear function (Hamilton 1994). We followed this assumption and detrend all time series data with ordinary least squares linear regression. Although there are other methods for achieving the same goal, such as filters, in our setting detrending must preserve the seasonal component S_t since otherwise no data will remain for estimating the season length m .

After removing the trend D_t , x is further normalized using $y_t = \frac{x_t - \hat{\mu}_x}{\hat{\sigma}_x}$, where $\hat{\mu}_x$ and $\hat{\sigma}_x$ are the sample mean and standard deviation of x , computed over all $t = 1, \dots, n$. Mathematically, $\hat{\mu}_x$ and $\hat{\sigma}_x$ may not represent the first and second

moment of process X , since no stationarity or ergodicity was assumed. However, we still performed the normalization, since it centers and scales data x , thus making the algorithms presented below more resilient to noise and outliers. In the remaining sections, it is assumed that x was preprocessed as described above.

4.2 Exhaustive estimation

An obvious solution to season length estimation as defined in Eq. (3) is to simply try out all $m < n$. The number of candidates can even be reduced to $m \in [2, \frac{n}{2}]$ since the subsequence matrix $\underline{x}_k$ becomes empty for larger m . A procedural description of this approach can be found in Algorithm 1.

Algorithm 1 Exhaustive season length estimation

Require: x
 $n \leftarrow \text{length}(x)$
for $m \leftarrow 2, 3, \dots, \lfloor \frac{n}{2} \rfloor$ **do**
 $k \leftarrow \lfloor \frac{n}{m} \rfloor$
 $\underline{x}_k \leftarrow \text{block}(x, k)$ //Eq. (2)
 $R_{\underline{x}_k} \leftarrow \text{cor}(\underline{x}_k)$ //cor := correlation measure Pearson ρ
 $\text{min_correlation}_m \leftarrow \min(R_{\underline{x}_k})$
end for
 $\text{solutions} \leftarrow \text{argmax}_m(\text{min_correlation})$
return $\min(\text{solutions})$

However, the exhaustive method has two drawbacks. First, for large n , looping over all m becomes infeasible. This reflects on both the computational complexity caused by the large parameter space of m , and the space complexity, since computing the correlation matrix $R_{\underline{x}_k}$ requires $\mathcal{O}(k^2)$ space, which is large for $m \ll n$. Second, as m grows, the likelihood of observing spurious correlations increases since the number of subsequences k decreases, allowing for a coincidentally large minimum of correlation matrix $R_{\underline{x}_k}$ due to the increased variance. This effect also makes the exhaustive algorithm susceptible to integer multiples of the correct season length $\hat{m}^*$, since it causes larger m to have slightly higher minimum correlation (although in theory all multiples of $\hat{m}^*$ should have the same minimum).

4.3 Spectral density estimation

Perhaps the most natural approach to season length estimation is Fourier analysis. Since seasonality is approximately periodic according to Definition 1, it is meaningful to analyze the spectral density $\mathcal{S}_X$, which can be estimated via periodogram $\hat{\mathcal{S}}$, defined as

$$\hat{\mathcal{S}}_x(\omega) = \frac{1}{n} \left| \sum_{j=1}^n x_j \exp(-i2\pi j\omega) \right|^2 \quad (4)$$

where $\exp(2\pi\omega j)$, $j \in 1, \dots, n$ are the fundamental frequencies of the Fourier transform. Spectral density $\mathcal{S}_X$ maps frequency ω to the density at which it occurs in time series X . Assuming that data x are sinusoidal, which is a strongly simplified model of seasonal data, then season length m corresponds to the period of the sinusoidal and its maximum likelihood estimate is given by Rife and Boorstyn (1974):

$$\hat{m} = \frac{1}{\operatorname{argmax}_{\omega} \hat{\mathcal{S}}_X(\omega)} \quad (5)$$

This agrees with our problem definition since for $x_t = \sin(\frac{2\pi}{m}t)$ all entries of the correlation matrix $R_{\underline{x}_k}$ will be 1, since

$$\begin{aligned} \sin\left(\frac{2\pi}{m}t\right) &= \sin\left(\frac{2\pi}{m}t + 2\pi\right) = \sin\left(\frac{2\pi}{m}(t+m)\right) \\ \implies \rho\left(\sin\left(\frac{2\pi}{m}t\right)^{(r)}, \sin\left(\frac{2\pi}{m}t\right)^{(s)}\right) &= 1 \quad \forall r, s \in 1, \dots, k \end{aligned} \quad (6)$$

The approach presented by Hyndman (2012) is very similar: an autoregressive process is fitted to the data and the periodogram of this process and its season length $\hat{m}$ are estimated with Eqs. (4) and (5) respectively. Using a periodogram to estimate the season length is restrictive, since this approach will fail for many non-sinusoidal signals, such as simple rectangular signals whose Fourier transform will have a maximum at frequency 0, leading to a division by 0 in Eq. (5). However, using spectral density estimates is certainly a valid approach to season length estimation, which has commonly been used in season length and periodicity estimation. Additionally, the periodogram can be estimated with a fast Fourier transform, allowing a computation in $\mathcal{O}(n \log n)$. A procedural description of this approach can be found in Algorithm 2.

Algorithm 2 Spectral Density Estimation

Require: x
 $\hat{\mathcal{S}} \leftarrow \text{FFT}(x)$
 $\hat{\mathcal{S}} \leftarrow |\hat{\mathcal{S}}|^2$
 $\hat{\omega} \leftarrow \operatorname{argmax}_{\omega} \hat{\mathcal{S}}(\omega)$
return $\hat{\omega}^{-1}$

4.4 Autocorrelation zero density

Another method for season length estimation is based on autocorrelation. The sample autocorrelation function $\hat{\mathcal{R}}_{xx}$ (ACF) of centered and scaled time series data can be computed as

$$\hat{\mathcal{R}}_{xx}(\tau) = \sum_{i=1}^{n-\tau} x_i x_{i+\tau} \quad (7)$$

where τ is the lag of interest. In theory, the sample ACF was designed to estimate the population autocorrelation of a covariance-stationary process, yet it can also be used for other purposes even if the assumption of covariance-stationarity is not met. As showed by Box et al. (2015), the sample ACF $\hat{\mathcal{R}}_{xx}$ of seasonal data will have peaks at regular intervals, where the length of the interval corresponds to season length m . Therefore, the sample ACF $\hat{\mathcal{R}}_{xx}$ can be used to estimate m . Let

$$\nabla^d[f](\tau) = \sum_{i=1}^d (-1)^i \binom{d}{i} f(\tau - i) \quad (8)$$

be the d^{th} finite difference of f , which approximates the d^{th} derivative of f . Assuming that the data are perfectly periodic with $x_t = x_{t+km}$, the theoretical ACF will be 1 for all integer multiples of m . Hence, $\hat{m}$ can be computed as

$$\hat{m} = \operatorname{argmin}_{\tau} \{ \tau : |\nabla^1[\hat{\mathcal{R}}_{xx}](\tau)| < \varepsilon, \nabla^2[\hat{\mathcal{R}}_{xx}](\tau) < 0 \} \quad (9)$$

where ε is a tolerance interval for the zeros of the first derivative, which is required since $\hat{\mathcal{R}}_{xx}$ is not continuous. Intuitively, Eq. (9) states that season length m corresponds to the lag τ in the sample ACF where its first peak occurs, and that this lag can be found using the standard optimization procedure of setting the first derivative to zero.

A disadvantage of this method is that the choice of ε will greatly influence the result. Too high values of ε will report peaks caused by noise, while too low values will lead to the peak of interest being missed. Further, since real data do not have infinite observations the sample ACF will decay as time lags increase (regardless of the data). Both of these aspects complicate identifying peaks in the sample ACF.

To overcome these problems, Toller and Kern (2017) developed a method based on ACF zeros instead of peaks for a more resilient computation. Intuitively, the distance between two zeros of a seasonally oscillating ACF corresponds to half the lag at which the ACF has its first periodic peak, as shown in Fig. 2.

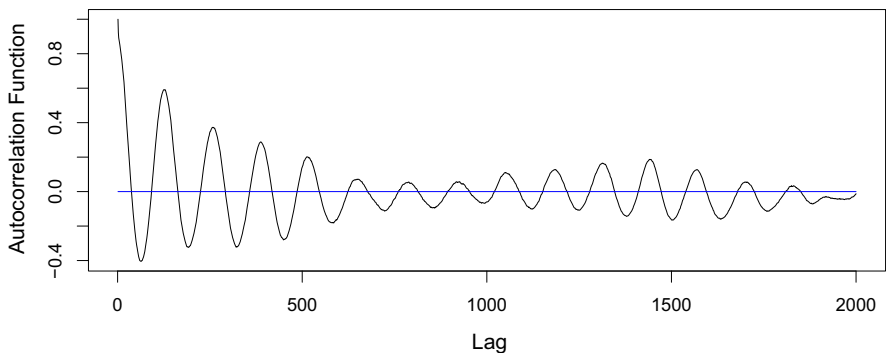


Fig. 2 Autocorrelation function of the sunspot data depicted in Fig. 1. Every positive peak is enclosed by two zero transitions (Color figure online)

Let $\{z = z_1, \dots, z_N : \Psi(\hat{\mathcal{R}}_{xx}, z) = 0\}$ be the zeros of the pairwise linearly interpolated sample ACF $\Psi(\hat{\mathcal{R}}_{xx})$. Since noise might as well cause zeros in the autocorrelation function, using the distance between two arbitrary consecutive zeros $2(z_i - z_{i+1})$ for season length estimation is unreliable. Rather, assuming that seasonality is more likely to cause zeros in the sample ACF than noise, it is possible to estimate the distribution of zero-to-zero distances via a kernel density estimate

$$K_z(r, h) = \frac{1}{h(N-1)} \sum_{i=1}^{N-1} \kappa\left(\frac{r - (z_{i+1} - z_i)}{h}\right) \quad (10)$$

where κ is a kernel function, and h the bandwidth. In this case, the season length estimate corresponds to half of the most dense zero-to-zero distance and is given by

$$\hat{m} = 2 \times \operatorname{argmax}_r K_z(r, h). \quad (11)$$

A disadvantage of kernel density estimation is that it requires two parameters that affect the outcome. For making a data-based selection of bandwidth h , one can use the derivative-based selector proposed by Sheather and Jones (1991). For kernel function κ , we suggest a Gaussian kernel. This is a sensible choice, assuming that the sum of all noise sources in the data converges to a Gaussian distribution according to the Central Limit Theorem. A procedural description of this approach can be found in Algorithm 3.

Algorithm 3 Autocorrelation Zero Density

Require: $\hat{\mathcal{R}}_{xx}$
 $n \leftarrow \text{length}(\hat{\mathcal{R}}_{xx})$
 $\psi \leftarrow \hat{\mathcal{R}}_{xx}$
for $i \leftarrow 1, \dots, n$ **do**
 if $\psi(i) > 0$ **then**
 $\psi(i) \leftarrow 1$
 else if $\psi(i) < 0$ **then**
 $\psi(i) \leftarrow -1$
 end if
end for
 $z \leftarrow \{z = z_1, \dots, z_r : \nabla^1[\psi] = 0\}$
 $\Psi \leftarrow \hat{\mathcal{R}}_{xx}(z) \times (-\hat{\mathcal{R}}_{xx}(z+1) - \hat{\mathcal{R}}_{xx}(z))^{-1}$
 $K_z \leftarrow \text{density}(\nabla^1[\Psi], \text{SJ}(\hat{\mathcal{R}}_{xx}))$ //SJ is the bandwidth selector proposed by Sheather and Jones (1991)
return $2 \times \operatorname{argmax}_r K_z(r)$

4.5 Average time series zero distances

A third method for season length estimation is based on the distances between zeros of the original time series x . Assuming that

- x is preprocessed as described above
- x has a single sinusoidal seasonality

- x has stochastic components whose combined variance is much smaller than the amplitude of the sinusoidal
- an arbitrary linear combination of the stochastic components of x converges towards a stable distribution as the number of summands grows

then the stochastic time-displacement of peaks, troughs and zero-crossings of the sinusoidal will follow the same stable distribution. This makes it possible to use the distribution's maximum likelihood estimator of the expected value to compute the most likely distance between two zero-crossings, which correspond to half of the season length if the above assumptions are met. A very commonly assumed stable distribution is the Gaussian distribution, and for this method one also assumes that zero-crossing displacements in x follow a Gaussian distribution. Since the maximum likelihood estimator of the expected value of the Gaussian distribution is the sample average, under the above assumptions season length $\hat{m}$ can be estimated based on the zeros of the pairwise interpolated time series $\{Z = Z_1, \dots, Z_r : \Psi(x, Z) = 0\}$ with

$$\hat{m} = \frac{2}{r-1} \sum_{i=1}^{r-1} \nabla^1[Z](i) = \frac{2}{r-1} (Z_r - Z_1) \quad (12)$$

Unlike autocorrelation zero distance densities, this method does not require input parameters such as bandwidth and a kernel function. However, the assumed distribution can also be viewed as a parameter. Altogether, average time series zero distances is a method with strongly simplifying assumptions, and it is bound to fail if its assumptions are not satisfied. A procedural description of this approach can be found in Algorithm 4.

Algorithm 4 Average Time Series Zero Distances

Require: x
 $n \leftarrow \text{length}(x)$
 $\psi \leftarrow x$
for $i \leftarrow 1, \dots, n$ **do**
 if $\psi(i) > 0$ **then**
 $\psi(i) \leftarrow 1$
 else if $\psi(i) < 0$ **then**
 $\psi(i) \leftarrow -1$
 end if
end for
 $Z \leftarrow \{Z = Z_1, \dots, Z_r : \nabla^1[\psi] = 0\}$
 $\Psi \leftarrow x_Z \times (-x_{Z+1} - x_Z)^{-1}$
return $\frac{2}{r-1} (\Psi_r - \Psi_1)$

5 Ensemble

The previous section presented three base methods that estimate season length as described in Eqs. (5), (11) and (12). In this section, we describe a method for com-

Table 2 Base methods

Name	Abbreviation	Defining equation
Spectral density estimation	S	(5)
Autocorrelation zero density	AZED	(11)
Average time series zero distances	ZE	(12)

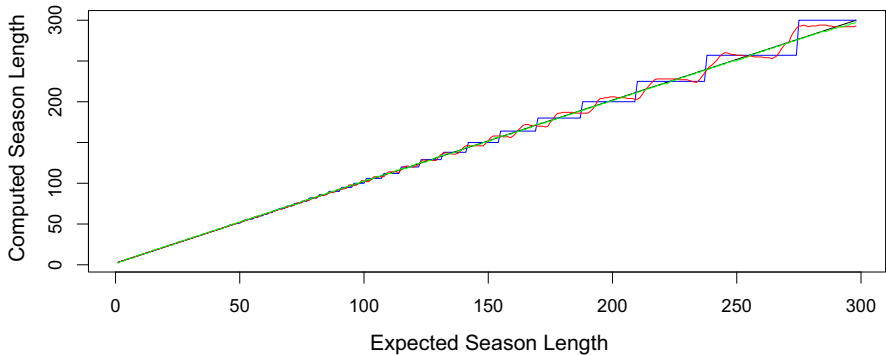


Fig. 3 A comparison of the base methods S (blue), AZED (red) and ZE (green) tested on sine waves with 1800 observations and varying periods (black). As the period to data length ratio increases, S and AZED become increasingly affected by sampling artifacts since the data are discrete and not continuous (Color figure online)

binning the base methods into a parameter-free season length estimation ensemble. An overview of the three methods is provided in Table 2.

5.1 Seasonality isolation

For purely sinusoidal data, all three base methods return the correct period. However, real data are discrete and finite, and not continuous, which is why in practice one will never encounter perfectly sinusoidal data. This may cause all three methods to fail even if the data are sinusoidal. We confirmed this in a simple experiment, where pure sine-waves of increasing period length with constant total length were presented to all three methods. The results of this experiment are depicted in Fig. 3.

A standard procedure for countering these effects is a sinc-interpolation, yet we have found a more promising approach based on autocorrelation composition.

In Eq. (7) we introduced one possible definition of the autocorrelation function (ACF). Another way to compute the ACF is based on the Wiener–Khinchin theorem, which states that the ACF can be computed based on the periodogram via an inverse discrete Fourier transform. Let $\mathcal{F}(x)$ denote the discrete Fourier transform of x and $\mathcal{F}^{-1}(x)$ the inverse discrete Fourier transform. Then, the (cyclical) sample autocorrelation function of x can also be computed as

$$\hat{\mathcal{R}}_{xx}(\tau) = \frac{1}{n^2 \text{Var}[x]} \mathcal{F}^{-1}(|\mathcal{F}(x)|^2) = \frac{1}{n^2 \text{Var}[x]} \sum_{l=1}^n \left| \sum_{j=1}^n x_j \exp(-i2\pi jl) \right|^2 \exp(i2\pi l\tau) \quad (13)$$

where $\text{Var}[x]$ is the sample variance of x . The computation of the autocorrelation according to this definition has no other effect on the frequencies in x other than squaring them. Hence, the spectral density-based method S is not affected by the computation of the autocorrelation. However, the zero-based methods AZED and ZE benefit from this since they assume that seasonality consists of a single, powerful sine-wave. Taking the square of the spectral density followed by a normalization will isolate a single sine-like behavior from the data since peak-frequencies will be more strongly affected by squaring than lower frequencies.

One can repeat this procedure and compute the autocorrelation function $\hat{\mathcal{R}}_{xx}^{\ell}(\tau)$ of the autocorrelation function of x . Let $\hat{\mathcal{R}}_{xx}^{\ell}$ denote the ℓ -fold composition of $\hat{\mathcal{R}}$ with itself. Effectively, computing $\hat{\mathcal{R}}_{xx}^{\ell}$ is similar to raising the periodogram $\hat{S}$ to the power of 2ℓ , which rapidly approaches zero for positive densities smaller than one. Normalization by the variance $\text{Var}[\hat{\mathcal{R}}_{xx}^{\ell-1}]$ prevents this, and allows a convergence towards a single sine-wave whose frequency is equivalent to the most powerful frequency in x . We have confirmed this in numerous practical simulations, and it appears intuitive from the theoretical formulation. However, formulating a simplified yet accurate closed form for $\hat{\mathcal{R}}_{xx}^{\ell}(\tau)$ is non-trivial due to the normalization step, and we therefore leave this for future work.

The spectral density estimation-based method S will remain unaffected by replacing x with $\hat{\mathcal{R}}_{xx}^{\ell}$, while the zero-based methods AZED and ZE will become more effective since more of their assumptions are met. Computing the autocorrelation zero density AZED even simplifies to the zero density of $\hat{\mathcal{R}}_{xx}^{\ell}$ without an additional computation of the ACF (ZED), since convergence is already reached in this case.

5.2 SAZED: spectral and average autocorrelation zero distance density

The main disadvantage of the exhaustive season length estimation method presented above was that it cannot be efficiently estimated. The three other base methods (S, AZED and ZE) can be computed efficiently via a fast Fourier transform in $\mathcal{O}(n \log n)$. However, these methods do not guarantee to return the correct season length if their assumptions are not met, and some of their assumptions are very restrictive.

A combination of methods, hereafter referred to as *ensemble*, may be a potential compromise between high computational complexity and strong assumptions. Instead of exhausting the parameter-space of season length m and evaluating all m according to Definition 2, one can only evaluate the season length candidates suggested by the three base methods. The ensemble then simply constructs the correlation matrix $R_{\underline{x}_k}$ for each of these candidates and checks which suggestion produces the largest minimum in the matrix. If several candidates result in the same highest minimum, then the smallest suggested season length $\hat{m}$ is returned, which is in line with our problem definition.

The ensemble is computationally efficient since a constant number of suggestions can be evaluated in $\mathcal{O}(n \log n)$ steps. Another advantage is that only the assumptions of one of S, AZED and ZE need to be met to guarantee a correct estimate. Moreover, adding further methods to the ensemble does not worsen the ensemble's estimates but rather simply increases computation time by a constant factor. Therefore, one can also

Table 3 Overview of the members of the SAZED ensemble

Method	Description
S	Spectral maximum of x
SA	Spectral maximum of $\hat{\mathcal{R}}_{xx}^\ell$
ZE	Average zero distances of x
AZE	Average zero distances of $\hat{\mathcal{R}}_{xx}^\ell$
ZED	Maximum zero distance density of x
AZED	Maximum zero distance density of $\hat{\mathcal{R}}_{xx}^\ell$
SAZED	Optimum of above suggestions per Definition 2

afford to include the seasonality isolation technique discussed above. Since the ℓ -fold self-composed autocorrelation $\hat{\mathcal{R}}_{xx}^\ell$ will converge exponentially, one can reasonably assume that $\ell \ll n$.

In short, we suggest the following ensemble:

- Collect season length estimates from raw data x with the periodogram (S), zero distance density (ZED) and average zero distances (ZE).
- Compute the ℓ -fold self-composed autocorrelation $\hat{\mathcal{R}}_{xx}^\ell$.
- Collect season length estimates from $\hat{\mathcal{R}}_{xx}^\ell$ with the same three estimators.
- For every unique estimate, compute the correlation matrix R_{x_k} .
- Return the estimate with produced the largest minimum in R_{x_k} .
- If multiple estimates are optimal, return the shortest estimate.

An overview of the ensemble is shown in Table 3 and a procedural description can be found in Algorithm 5.

5.3 Alternative ensembles

The ensemble presented above is only one way of combining the base methods. Alternative methods are, for instance, majority votes. There, the estimates from the ensemble members are collected and the most commonly suggested estimate is returned. However, in our setting, consensus-based ensembles have two clear drawbacks. First, if there is no clear consensus, one needs an additional method for breaking ties. We experimented with mean-based downsampling for this purpose, but this adds additional assumptions to the entire ensemble. Second, the most commonly suggested season length estimate does not have to be the best estimate in terms of the problem definition. This second drawback is also true for other ensemble techniques, such as weighted averages or Bayesian methods.

However, under several scenarios these alternative ensemble methods may be appropriate. For instance, Vlachos et al. (2005) computed a spectral density-based estimate of a time series' periodicity, which is comparable to season length estimation with S. Subsequently, they analyzed if a period suggested from the periodogram coincides with a peak in the autocorrelation function. Another situation that may require different ensemble methods is when one expects the data to contain multiple different seasonalities (Lijffijt et al. 2012; Yuan et al. 2017). The SAZED ensemble as pre-

Algorithm 5 SAZED

Require: x

$$\hat{\mathcal{R}}_{xx}^0 \leftarrow x$$

$$\hat{\mathcal{R}}_{xx}^1 \leftarrow \hat{\mathcal{R}}_{xx} // \text{Eq. (13)}$$

$$i \leftarrow 1$$
while $\hat{\mathcal{R}}_{xx}^i \neq \hat{\mathcal{R}}_{xx}^{i-1}$ **do**

$$\hat{\mathcal{R}}_{xx}^{i+1} \leftarrow \hat{\mathcal{R}}_{\hat{\mathcal{R}}_{xx} \times \hat{\mathcal{R}}_{xx}^i} // \text{Eq. (13) with } x_t \text{ replaced by } \hat{\mathcal{R}}_{xx}^i$$

$$i \leftarrow i + 1$$
end while

$$\hat{m}_1 \leftarrow S(x)$$

$$\hat{m}_2 \leftarrow S(\hat{\mathcal{R}}_{xx}^i)$$

$$\hat{m}_3 \leftarrow \text{ZED}(x)$$

$$\hat{m}_4 \leftarrow \text{ZED}(\hat{\mathcal{R}}_{xx}^i)$$

$$\hat{m}_5 \leftarrow \text{ZE}(x)$$

$$\hat{m}_6 \leftarrow \text{ZE}(\hat{\mathcal{R}}_{xx}^i)$$
for $m = \hat{m}_1, \dots, \hat{m}_6$ **do**

$$k \leftarrow \lfloor \frac{n}{m} \rfloor$$

$$\underline{x}_k \leftarrow \text{block}(x, k) // \text{Eq. (2)}$$

$$R_{\underline{x}_k} \leftarrow \text{cor}(\underline{x}_k) // \text{cor} := \text{correlation measure Pearson } \rho$$

$$\text{min_correlation}_m \leftarrow \min(R_{\underline{x}_k})$$
end for

$$\text{solutions} \leftarrow \text{argmax}_m(\text{min_correlation})$$
return $\min(\text{solutions})$

sented above was not designed for this. If its members suggest different “correct” season lengths, it will return the shortest solution if all solutions have exactly the same minimum in the correlation matrix. This is unlikely due to numerical inaccuracies, and we expect SAZED to behave similarly to exhaustive evaluation under such scenarios.

6 Experimental study

In our experiments, we evaluate the performance of SAZED and variations thereof against commonly used and state-of-the-art baselines using both real-world and synthetic datasets. Subsequently, we discuss our results in terms of the performance of each algorithm.

6.1 CRAN dataset

For our dataset selection, we start by extracting all open-source univariate time series of packages listed in the “Time Series Data” section of the CRAN Task View on Time Series Analysis,² a comprehensive collection of R packages on time series analysis. The time series included in those packages span across a wide variety of application domains, ranging from economic indicators, such as employment rates or retail sales, to environmental measurements, such as pollution levels or the number of sunspots. We consider only time series of the R object class “ts”. Further, we discard time

² <https://cran.r-project.org/web/views/TimeSeries.html>, retrieved in March 2019.

Table 4 Packages and names of the time series included in the evaluation for the first dataset

R Package	Time series names
AER (Kleiber and Zeileis 2008)	BondYield, DutchSales, UKNonDurables
astsa (Stoffer 2016)	birth, cmort, flu, gas, hor, part, prodn, qinfl, qintr, rec, so2, soi, sunspotz, tempr, unemp, UnempRate
expsmooth (Hyndman 2015)	bonds, cangas, enplanements, frexport, mcopper, ukcars, usgdp, utility, vehicles, visitors
fma (Hyndman 2017a)	airpass, beer, bricksq, condmilk, dole, elec, fancy, hsales, hsales2, invent15, labour, milk, motion, pigs, plastics, pollution, qelec, qsales, shampoo, ukdeaths, usdeaths, uselec, writing
fpp (Hyndman 2013)	cafe, euretail
fpp2 (Hyndman 2017b)	a10, ausbeer, auscafe, ausgdp, austourists, debitcards, elecequip, gasoline, h02, hyndsight, qauselec, qcement, qgas, usmelec
TSA (Chan and Ripley 2012)	airmiles, beersales, co2, flow, hours, milk, JJ, oilfilters, prescrip, prey.eq, retail, tempdub, wages, winnebago

series objects without a given “frequency” parameter (in R, this corresponds to a frequency equal to one), since we take this given frequency as the ground truth seasonality value to evaluate our approach. However, since some of the time series with given frequency may not exhibit any seasonal patterns (irrespective of the frequency parameter), we manually inspected all time series with given frequency to check if there are any such cases. Specifically, three annotators independently annotated all time series with given frequency as being seasonal or not. We report a substantial inter-rater agreement for this annotation task (Fleiss’ $\kappa = .77$) and we thus discard only time series which all three annotators agreed to be not seasonal. This procedure results in a dataset of 82 univariate time series to be evaluated. The lengths of these time series range from 16 to 3024 observations (mean 290), and their season length ranges from 2 to 52 observations (mean and mode both equal to 12). For the sake of completeness, we list the names of the packages and time series of the CRAN dataset in Table 4.

6.2 SL dataset

To make the performance of SAZED comparable to other season length estimation algorithms, we included the test data used by Toller and Kern (2017) in the evaluation. This SeasonLength (SL) dataset consist of 125 synthetic and 40 real-world time series. The synthetic time series were designed to assess various aspects of estimation robustness, ranging from noise and outlier resilience over high result domain variance to time series without seasonality. This dataset captures characteristics that are different from those found in the CRAN dataset described in Sect. 6.1. The number of observations of time series in this dataset ranges from 20 to 327,680 (mean 7,586), and the season lengths range from 3 to 65,536 (mean 1,341). Further, anomalies, novelties and complex trends are just as common as in this dataset as sterile cosine-like time series behavior.

6.3 Cauchy dataset

In addition to the CRAN and SL datasets, we synthetically generated a third dataset consisting of 100 randomly sampled sine waves with an added linear trend and additive noise sampled from a Cauchy distribution. In particular, the Cauchy Dataset was generated as follows:

- Select period and amplitude from the interval $[3, 1000]$ uniformly at random.
- Select a phase shift from the interval $[-\pi, \pi]$ uniformly at random.
- Generate a sin wave with $n = 3000$ observations and the above properties.
- Select a random linear trend slope from a standard Gaussian distribution and generate a linear function with that slope and $n = 3000$ observations.
- Generate a random Autoregressive process of order 1 (AR(1) (Hamilton 1994)) $n = 3000$ observations and with coefficient $\phi \sim U(0, 1)$ and random innovations $\iota \sim C(0, 1)$, where U is the uniform distribution and C the Cauchy distribution.
- Add trend and the AR process to the sine-wave.

Since the Cauchy-distribution is a statistical edge case with no finite moments of any order, many of the assumptions made by SAZED and competing methods are not met. Beyond these methods, we envision this challenging dataset as a stress test also for future seasonality estimation methods to improve upon. However, in this Cauchy dataset (as is the case of the previous ones), when the data are plotted at an appropriate resolution, finding the correct season length is trivial for humans. We exemplify the shape of time series in the Cauchy dataset in Fig. 4.

6.4 Setup

Our evaluation setup was designed to directly compare SAZED with *findfrequency* by Hyndman (2017b), and the *seasonLength* algorithm for season length detection by Toller and Kern (2017). Using the previously described datasets, we compare the algorithms' performance with respect to two measures and three different settings. The first evaluation measure considered is raw accuracy of season length estimation, i.e. the number of time series in which a season length estimation algorithm detected the correct season length divided by the total number of estimated season lengths. The second evaluation measure we consider is based on the absolute distance between the algorithms' estimates and the correct season length. This absolute distance also serves as the input for a Friedman's rank test (Demšar 2006), which ranks the algorithms and displays the results in a critical difference plot. In the first setting, season length suggestions are only considered correct if they are equal to the reference value (cf. $\pm 0\%$ in results Table 5), while in the second setting there is a tolerance interval of $\pm 20\%$. In other words, a season length of a time series in this second setting is considered correctly detected if the estimated value is within a range of $\pm 20\%$ of the reference value. In the third setting, we count how often an algorithm returned an integer multiple of the reference value (cf. "mult" in Table 5). All three settings were applied for all three datasets. Furthermore, we distill SAZED's performance by component, evaluating how each member of the SAZED ensemble contributes to its overall performance.

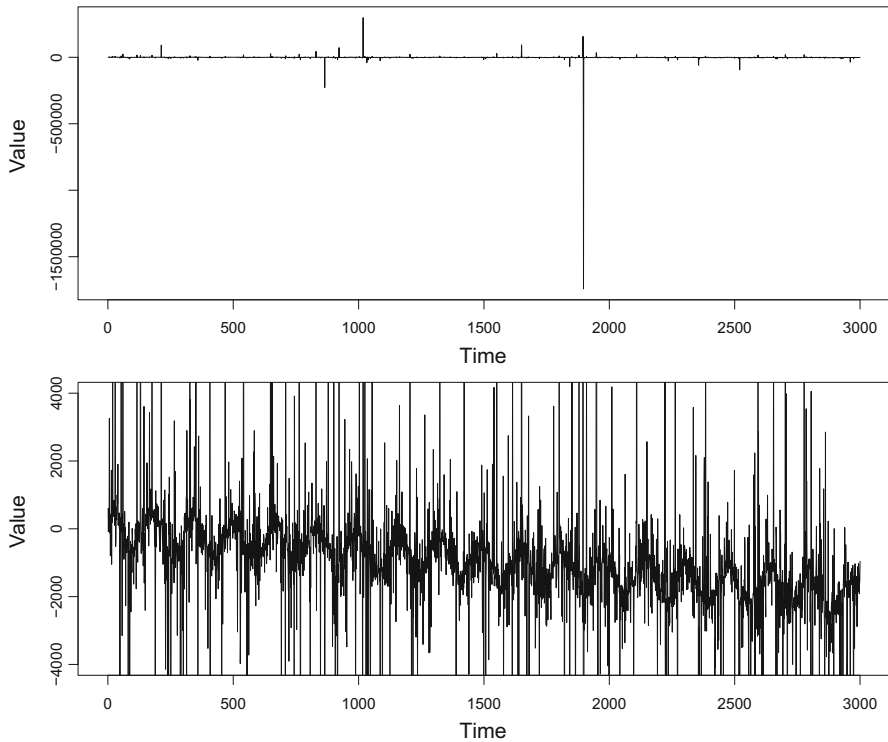


Fig. 4 An example of time series data from the Cauchy dataset. The top plot depicts a full view of the data, while the axis limits of the bottom plot were adjusted to make the underlying seasonality visible

6.5 Results

Table 5 provides the season length estimation accuracy of *findfrequency*, *seasonLength*, $\text{SAZED}_{\text{opt}}$ (Sect. 5.2) and $\text{SAZED}_{\text{maj}}$ (Sect. 5.3), as well as SAZED's individual members on time series from all of the above-described datasets. There is no singular best performing method: $\text{SAZED}_{\text{opt}}$ and $\text{SAZED}_{\text{maj}}$ share the first place, performing similarly in terms of estimation accuracy, while almost always scoring higher than all other methods.

In the Friedman's rank tests, $\text{SAZED}_{\text{opt}}$ has the highest rank on all three datasets. It is also significantly better than every other method on at least one dataset, as shown in Figs. 5, 6 and 7. Within the SAZED ensemble, no single member is significantly better than all other members on all datasets.

6.6 Discussion

The above results indicate that both variants of SAZED performed overall better than *findFrequency* and *seasonLength*.

The performance accuracy of *findFrequency* appears to be greatly affected by non-Gaussian noise. This was to be expected since the method fits an AR(1) process

Table 5 Season length estimation accuracy of *findFrequency*, *seasonLength*, SAZED and SAZED's components

Method	Accuracy								
	<i>CRAN dataset</i>			<i>SL dataset</i>			<i>Cauchy dataset</i>		
	$\pm 0\%$	$\pm 20\%$	mult	$\pm 0\%$	$\pm 20\%$	mult	$\pm 0\%$	$\pm 20\%$	mult
<i>findFrequency</i>	36	<i>40</i>	1	55	85	2	0	0	0
<i>seasonLength</i>	12	28	6	52	<i>121</i>	7	0	8	0
SAZED _{opt}	43	51	6	72	120	10	1	27	0
SAZED _{maj}	36	<i>40</i>	26	73	122	<i>21</i>	3	23	0
S	34	39	24	66	116	25	1	26	0
SA	34	<i>40</i>	22	65	117	25	1	21	0
ZED	10	16	2	36	76	6	0	0	0
AZED	30	39	7	58	<i>121</i>	6	3	22	0
ZE	17	29	5	33	63	10	0	4	0
AZE	<i>37</i>	39	10	53	117	11	2	22	0

The highest and second highest scores in each column are given in bold and italic, respectively. $\pm 0\%$ indicates the raw accuracy score per dataset, whereas $\pm 20\%$ states that the results were also considered correct if they were within a 20% tolerance interval around the correct value. The column “mult” lists the number of times an integer multiple of the correct solution was returned. SAZED_{opt} and SAZED_{maj} feature the overall highest estimation accuracy

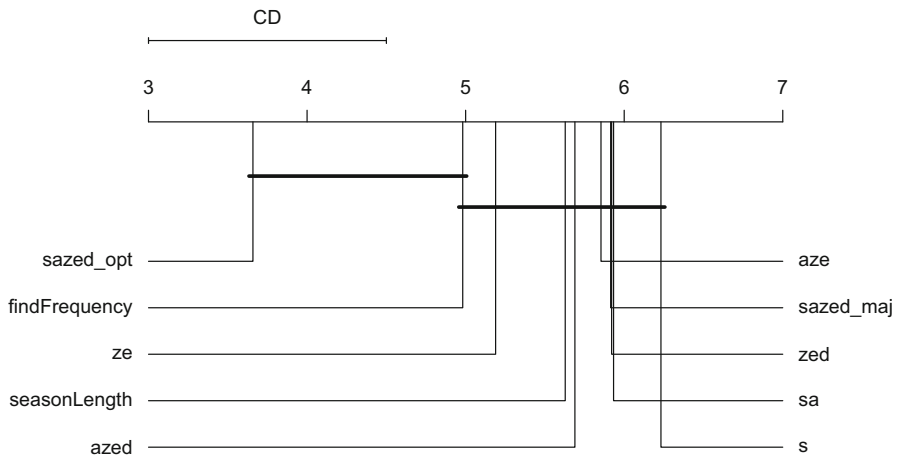


Fig. 5 Critical distance plot for the CRAN dataset. A critical difference plot ranks the algorithms based on their performance as measured by the distance between their estimated seasonality lengths and the actual ones. The critical difference plot depicts average ranks in order, where lower is better and the horizontal bars highlight no significant difference in the compared pairs. SAZED_{opt} is significantly better than SAZED_{maj} and *seasonLength*, but not significantly better than *findFrequency*

to the data during preprocessing and assumes that the process' residuals follow a Gaussian distribution. Further, it appears to benefit less from a tolerance interval around the reference value than all other methods. This suggests that the output of

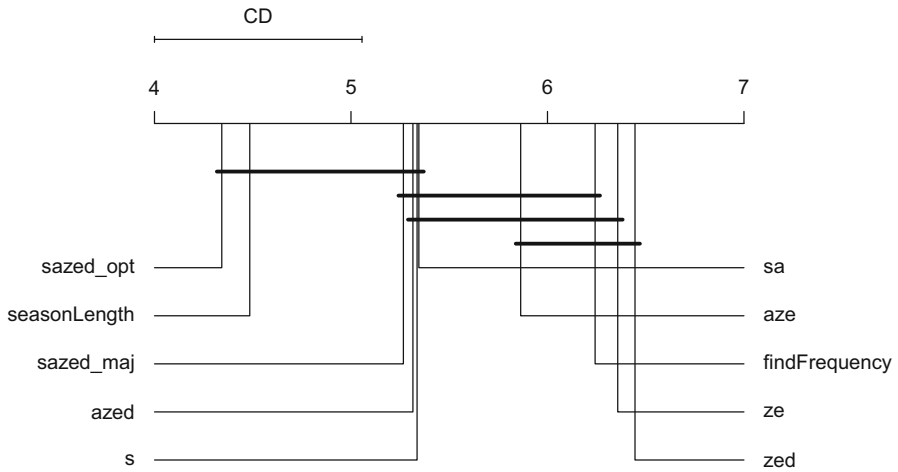


Fig. 6 Critical distance plot for the SL dataset. $\text{SAZED}_{\text{opt}}$ is significantly better than findFrequency , but not significantly better than $\text{SAZED}_{\text{maj}}$ and seasonLength . $\text{SAZED}_{\text{maj}}$ is not significantly better than findFrequency

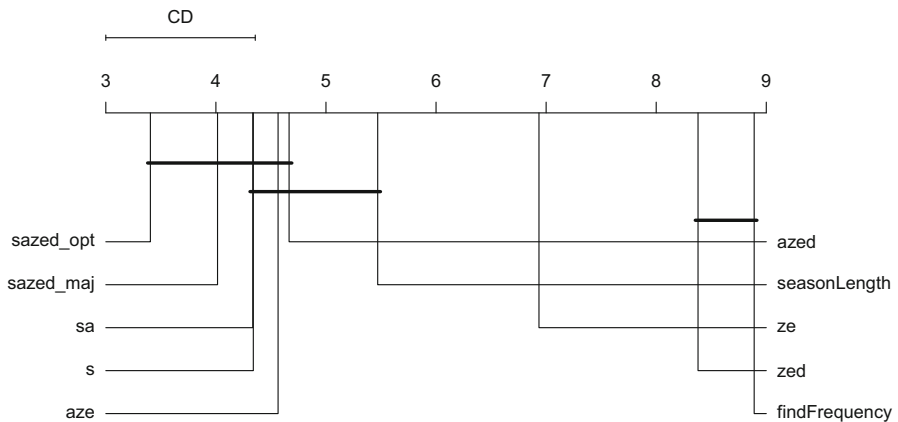


Fig. 7 Critical distance plot for the Cauchy dataset. Both variants of SAZED are significantly better than findFrequency , but not significantly better than seasonLength

this method is likely either “correct” or “incorrect” and seldom “almost correct”. Its results for the Cauchy dataset make this fact particularly evident, since all other methods, except ZED, greatly benefited from the more lenient performance measure. Regarding findFrequency ’s ranking, the difference from its rank in the CRAN dataset when compared to the other datasets can be explained by the large differences in the input and output domains of these datasets. It appears that findFrequency rarely suggests long season lengths and is less suited for long time series. This weakness is likely caused by the fitting of an AR(1) process, since this method otherwise would be identical to the ensemble member S.

The high score achieved by *seasonLength* on the SL dataset in the $\pm 20\%$ setting does not compensate for its otherwise below-average performance. The results of the experimental study even suggest that *seasonLength* might overfit in this one particular pair of evaluation methodology and dataset since the method's performance is low in all other cases. It appears that although this method is parameter-free, it contains hidden assumptions that make it unreliable in a domain-agnostic setting. The most problematic step in the *seasonLength* algorithm is likely the application of a low-pass filter since this implicitly assumes that either the input is noisy or else its season length is unaffected by filtering. This does not imply that low-pass filters cannot be beneficial for the purpose of season length estimation. However, as mentioned in Sect. 5.1, we believe that the ℓ -fold self-composition of the autocorrelation function has a similar effect while being parameter-free and not assuming any particular distribution.

A direct comparison between SAZED_{opt} and SAZED_{maj} indicates their similar performance on both datasets, although it also seems that SAZED_{maj} is the less reliable than SAZED_{opt} . This becomes evident when one also adds the members S and SA to the comparison. In all settings, SAZED_{maj} performed similar to these two methods. Since S and SA commonly return the same result, this frequently forms a majority when there otherwise would be none. SAZED_{opt} has no such deficiency: all unique member suggestions are evaluated according to the problem definition and duplicate suggestions make no difference.

S and SA appear to return an integer multiple of the correct solution particularly often, and both methods even appear to outperform SAZED in the SL dataset. However, SAZED_{maj} still has a higher combined accuracy if one counts these multiples as correct solutions and adds them to the accurate $\pm 0\%$ results: e.g., in the SL dataset, SAZED_{maj} and S correctly estimate season length in a total of 94 and 91 cases, respectively. However, we also attribute these performance differences between SAZED and its ensemble members in the “mult” accuracy measure to the fact that SAZED per definition (cf. Problem Statement in Definition 2) optimizes season length estimation towards shorter season lengths (and thus not its multiples).

The fact that no algorithm returned a multiple of a correct solution in the Cauchy dataset is likely due to the overall bad performance across all algorithms on this dataset. It appears that the infinite variances generated by heavy-tail distributions deviate too much from the methods' assumptions. Developing a season length estimator that remains robust under such conditions seems a promising task for future work.

7 Conclusion

Time series season length estimation is a common pre-processing task in time series analysis. Improvements to existing algorithms can benefit a wide array of downstream applications and, in general, facilitate future analysis of seasonal data. In this work, we introduced an ensemble method, SAZED, for detecting the season length of seasonal time series. SAZED supports practitioners in their time series analysis pipelines by providing a simple interface, which does not require parameter-tuning

and domain-specific pre-processing. Moreover, SAZED is theoretically grounded, features encouraging theoretical computational complexity of $\mathcal{O}(n \log n)$ and copes with a broad variety of time series properties, such as different period lengths and noise. In an experimental study with both synthetic and real datasets, our method compares favorably to state-of-the-art alternatives.

Our approach focused on estimating a single season length in one time series. Extending our approach to estimate multiple season lengths from a time series (e.g. daily and monthly temperature cycles) is an interesting avenue for future work. Further future work includes the development of a season length estimator that is resilient to data contamination from various sources of non-Gaussian noise. Note that we did not provide formal justification for our repeated application of the autocorrelation function. Presenting a closed form for this iterated function is additional future work. Finally, another interesting extension to SAZED that could benefit practitioners would be adding a module to automatically detect if an input time series is seasonal or contains enough of a seasonal pattern for SAZED to output sensible results.

Acknowledgements Open access funding provided by Graz University of Technology. We thank the anonymous reviewers for their valuable feedback on the manuscript. Our work was partly funded by the iDev40 project. The iDev40 project has received funding from the ECSEL Joint Undertaking (JU) under Grant Agreement No. 783163. The JU receives support from the European Union's Horizon 2020 research and innovation programme. It is co-funded by the consortium members, grants from Austria, Germany, Belgium, Italy, Spain and Romania. Tiago Santos is a recipient of a DOC Fellowship of the Austrian Academy of Sciences at the Institute of Interactive Systems and Data Science of the Graz University of Technology.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Andrews DF, Herzberg AM (2012) Data: a collection of problems from many fields for the student and research worker. Springer, Berlin
- Box GE, Jenkins GM, Reinsel GC, Ljung GM (2015) Time series analysis: forecasting and control. Wiley, London
- Cerqueira V, Torgo L, Pinto F, Soares C (2017) Arbitrated ensemble for time series forecasting. In: Joint European conference on machine learning and knowledge discovery in databases, Springer, Berlin, pp 478–494
- Chan KS, Ripley B (2012) TSA: time series analysis. <https://CRAN.R-project.org/package=TSA>, R package version 1.01
- Cleveland RB, Cleveland WS, McRae JE, Terpenning I (1990) STL: a seasonal-trend decomposition. *J Off Stat* 6(1):3–73
- Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. *J Mach Learn Res* 7(Jan):1–30
- Elfeky MG, Aref WG, Elmagarmid AK (2005a) Periodicity detection in time series databases. *IEEE Trans Knowl Data Eng* 17(7):875–887
- Elfeky MG, Aref WG, Elmagarmid AK (2005b) WARP: time warping for periodicity detection. In: Data mining, fifth IEEE international conference on, IEEE, pp 8–pp
- Fuller WA (2009) Introduction to statistical time series, vol 428. Wiley, London
- Ghosh A, Lucas C, Sarkar R (2017) Finding periodic discrete events in noisy streams. In: Proceedings of the 2017 ACM on conference on information and knowledge management, ACM, pp 627–636

- Hamilton JD (1994) Time series analysis, vol 2. Princeton University Press, Princeton
- Hyndman RJ (2012) Measuring time series characteristics. <https://robjhyndman.com/hyndsight/tseriescharacteristics/>, Accessed 21 Feb 2018
- Hyndman RJ (2013) FPP: data for “Forecasting: principles and practice”. <https://CRAN.R-project.org/package=fpp>, R package version 0.5
- Hyndman RJ (2015) expsmooth: Data Sets from “Forecasting with exponential smoothing”. <https://CRAN.R-project.org/package=expsmooth>, R package version 2.3
- Hyndman RJ (2017a) FMA: data sets from “Forecasting: methods and applications” by Makridakis, Wheelwright & Hyndman (1998). <https://CRAN.R-project.org/package=fma>, R package version 2.3
- Hyndman RJ (2017b) FPP2: data for “Forecasting: principles and practice” (2nd edition). <https://CRAN.R-project.org/package=fpp2>, R package version 2.1
- Hyndman RJ, Athanasopoulos G (2018) Forecasting: principles and practice. OTexts
- Jönsson P, Eklundh L (2002) Seasonality extraction by function fitting to time-series of satellite sensor data. *IEEE Trans Geosci Remote Sens* 40(8):1824–1832
- Keogh E, Lonardi S, Ratanamahatana CA (2004) Towards parameter-free data mining. In: Proceedings of the tenth ACM SIGKDD international conference on knowledge discovery and data mining, ACM, pp 206–215
- Kleiber C, Zeileis A (2008) Applied econometrics with R. Springer, New York, <https://CRAN.R-project.org/package=AER>, ISBN 978-0-387-77316-2
- Kumar M, Patel NR, Woo J (2002) Clustering seasonality patterns in the presence of errors. In: Proceedings of the eighth ACM SIGKDD international conference on knowledge discovery and data mining. ACM, pp 557–563
- Lijffijt J, Papapetrou P, Puolamäki K (2012) Size matters: finding the most informative set of window lengths. In: Joint European conference on machine learning and knowledge discovery in databases. Springer, Berlin, pp 451–466
- Pierson E, Althoff T, Leskovec J (2018) Modeling individual cyclic variation in human behavior. In: Proceedings of the 2018 world wide web conference on world wide web, international world wide web conferences steering committee, pp 107–116
- Ramírez D, Schreier PJ, Vía J, Santamaría I, Scharf LL (2014) A regularized maximum likelihood estimator for the period of a cyclostationary process. 48th Asilomar conference on signals, Systems and Computers, IEEE, pp 1972–1976
- Rasheed F, Alshalalfa M, Alhaji R (2011) Efficient periodicity mining in time series databases using suffix trees. *IEEE Trans Knowl Data Eng* 23(1):79–94
- Rife D, Boorstyn R (1974) Single tone parameter estimation from discrete-time observations. *IEEE Trans Inf Theory* 20(5):591–598
- Sheather SJ, Jones MC (1991) A reliable data-based bandwidth selection method for kernel density estimation. *J R Stat Soc: Ser B (Methodological)* 53(3):683–690
- Stoffer D (2016) astsa: Applied statistical time series analysis. <https://CRAN.R-project.org/package=astsa>, R package version 1.7
- Toller M, Kern R (2017) Robust parameter-free season length detection in time series. In: Proceedings of the 3rd SIGKDD workshop on mining and learning from time series
- Toller M, Santos T, Kern R (2019) sizedR: parameter-free domain-agnostic season length detection in time series. <https://CRAN.R-project.org/package=sizedR>, R package version 2.0.0
- Vlachos M, Yu P, Castelli V (2005) On periodicity detection and structural periodic similarity. In: Proceedings of the 2005 SIAM international conference on data mining, SIAM, pp 449–460
- Wang J, Chen T, Huang B (2006a) Cyclo-period estimation for discrete-time cyclo-stationary signals. *IEEE Trans Signal Process* 54(1):83–94
- Wang X, Smith K, Hyndman RJ (2006b) Characteristic-based clustering for time series data. *Data Min Knowl Discov* 13(3):335–364
- Yuan Q, Shang J, Cao X, Zhang C, Geng X, Han J (2017) Detecting multiple periods and periodic patterns in event time sequences. In: Proceedings of the 2017 ACM on conference on information and knowledge management, ACM, pp 617–626

Affiliations

Maximilian Toller¹ · Tiago Santos²  · Roman Kern¹

Maximilian Toller
mtoller@know-center.at

Roman Kern
rkern@know-center.at

¹ Know-Center GmbH, Inffeldgasse 13, 6th floor, 8010 Graz, Austria

² Institute of Interactive Systems and Data Science, Graz University of Technology, Inffeldgasse 16c/I, 8010 Graz, Austria