British Journal of Applied Science & Technology
13(6): 1-11, 2016, Article no.BJAST.22726
ISSN: 2231-0843, NLM ID: 101664541

SCIENCEDOMAIN *international*
www.sciencedomain.org

Area 1 of Approximate Entropy as a Fast and Robust Tool to Address Temporal Organization

José Eduardo Soubhia Natali[1] and José Guilherme Chaui-Berlinck[1*]

[1]Laboratório de Energética e Fisiologia Teórica, Instituto de Biociências/Universidade de São Paulo, Rua do Matão tr.14, n.321 – Butantã/São Paulo, CEP: 05508-090, SP, Brazil.

Authors' contributions

This work was carried out in collaboration between both authors. Both authors contributed equally to the study, writing and approved the final manuscript.

Article Information

DOI: 10.9734/BJAST/2016/22726
Editor(s):
(1) Qing-Wen Wang, Department of Mathematics, Shanghai University, P.R. China.
Reviewers:
(1) Chen Gao, Tufts University, USA.
(2) S. Zimeras, University of the Aegean, Greece.
Complete Peer review History: http://sciencedomain.org/review-history/12742

Original Research Article

Received 22nd October 2015
Accepted 30th November 2015
Published 19th December 2015

ABSTRACT

Aims: To evaluate the consistency and robustness of an informational entropy analytical tool derived from Approximate Entropy (ApEn).

Study Design: A set of *in machina* time-series of known properties were generated to test and compare the proposed tool with the standard ApEn and with peak-ApEn.

Place and Duration of Study: Laboratory of Energetics and Theoretical Physiology, Dept. Physiology, Biosciences Institute, University of São Paulo. From April 2014 to May 2015.

Methodology: The proposed tool consists in obtaining a detailed tolerance vector with more than 100 values and, then, to compute ApEn for window m = 1 for each one of these tolerance values. This creates a curve that is numerically integrated using a normalized tolerance vector as the basis, thus obtaining the area under the curve of m = 1 ApEn (a1ApEn). In order to make comparisons, 17 time-series from different generating processes were constructed using Matlab R2013a. Employing the above-cited analytical tools, we approached the following queries: (a) for a given process, how variable is the estimator value? (b) is a1ApEn more consistent than peak-ApEn in classifying different processes?

Results: The answer for (a) is that, in relation to ApEn, the variance of a1ApEn is significantly lower in 16 cases (all $P < .01$, F-test for sample variance), and we explain why the one exception occurs. In relation to peak-ApEn, the variance is lower for all 17 series (all $P < .01$). The answer for

**Corresponding author: E-mail: jgcb@usp.br;*

(b) is that a1ApEn is able to correct inconsistencies found when using peak-ApEn (all $P < .01$, Student's t-test).

Conclusion: The proposed tool, the area under the curve for ApEn of window 1 (a1ApEn) is objective and more consistent than both the ApEn and the peak-ApEn estimators.

Keywords: Algorithms; computer-assisted numerical analysis; information theory; approximate entropy; consistency; objectivity.

1. INTRODUCTION

Approximate Entropy (ApEn) is a widely employed tool to characterize temporal organization in time-series (**S**). This method seeks to estimate the degree of organization by counting the number of equal events (matches) of a sub-vector *i* of size *m* along the original vector, where the distance between two sub-vectors is given by the Heaviside function for a certain tolerance *r*. More details can be found in [1,2], but the central idea is to count all the matches (#) for a certain *i, m* and *r* as:

$$C_i^m(r) = \frac{\#_i^m}{N - m + 1} \quad (1)$$

From the counts of each sub-vector *i*, another value, ϕ, is computed:

$$\phi^m(r) = \frac{1}{N - m + 1} \sum_{1}^{N-m+1} \ln\left(C_i^m\right) \quad (2)$$

Finally, ApEn is obtained through:

$$\text{ApEn}(m,r,N) = \phi^m(r) - \phi^{m+1}(r) \quad (3)$$

On the one hand, ApEn putatively achieves its objective in many cases, as, for example, prediction of survivability through body temperature regularity and estimation of machine health via analysis of vibration in rolling bearings [3–9]. On the other hand, drawbacks in the estimator are present [2,10,11] and, consequently, there is room for improvement.

A first issue is what we call as "positional sensitivity": two arbitrarily chosen fragments of a larger original data may possess very distinct ApEn values. This problem holds true even for highly organized series such as the sum of two sine waves (Fig. 1A), where it is possible to observe a twofold increase in ApEn values in different sampling intervals of the original data set (Fig. 1B). This problem is usually mitigated, as shown in Fig. 1B, by the use of a moving window to obtain the mean ApEn of a given time-series [6,8,9].

A second, and much more important issue, is the lack of an objective procedure to obtain the value of the estimator. ApEn relies on two arbitrary choices of parameters, namely, the window size of comparison, *m*, and the tolerance for distinguishing two vectors as non-equals, *r* (equations 1-3). This arbitrariness is a huge shortcoming in this tool since two very distinct series can be classified differently depending on the choice of the parameters [2]. A well known variation of ApEn, sample entropy (developed by Richman & Moorman [10]), also from suffers from this weakness. There are two attempts to overcome such a problem, as presented below.

One is based on the computation of ApEn for a large set of tolerance values in order to obtain the highest (peak) ApEn for a given *m* (peak-ApEn - [11–13]). The logic behind this approach is clear when comparing peak ApEn values in different time-series (Fig. 2). Due to the formulation of equation 3, small tolerances are associated with small ApEn values (since the counting is as small for $m = k$ as it is for $m = k+1$). Therefore, as the tolerance increases, ApEn values rise to a peak and then decreases with further tolerance increases (since for large tolerances all sub-vectors would be considered as equal for both $m = k$ and $m = k+1$, resulting in ApEn $\rightarrow$ 0). Therefore, it is possible to observe that the use of a single value of *r*, as suggested by Pincus [1] (e.g. 0.15 as illustrated in Fig. 2) can lead to spotting different regions of the ApEn curves of different time-series.

The other alternative approach is based on a double summation of ApEn values along all suitable *m* and *r*, resulting in a pseudo-volume below the surface thus obtained (vApEn, developed by Santos et al. [2]). vApEn is much more robust than ApEn and Sample ApEn. However, it is extremely demanding on computational time/resources, and turns out prohibitive for series containing more than 400 points even in powerful conventional computers.

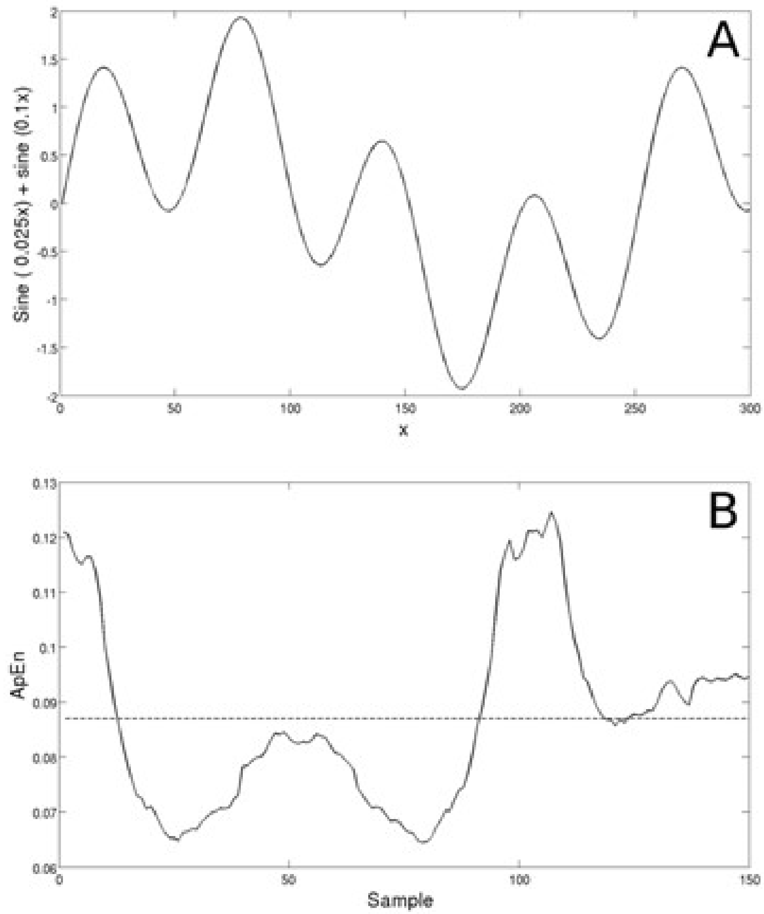

Fig. 1. Sum of two sine waves and the corresponding moving ApEn

(A) Sum of two sine waves, one with frequency of 0.025 Hz and the other with frequency of 0.1 Hz, as a function of time (x – in arbitrary units). (B) Using a 150 points window, ApEn values were sequentially obtained along the original series. the dotted line represents the mean value. Notice the near two-fold variation in ApEn values

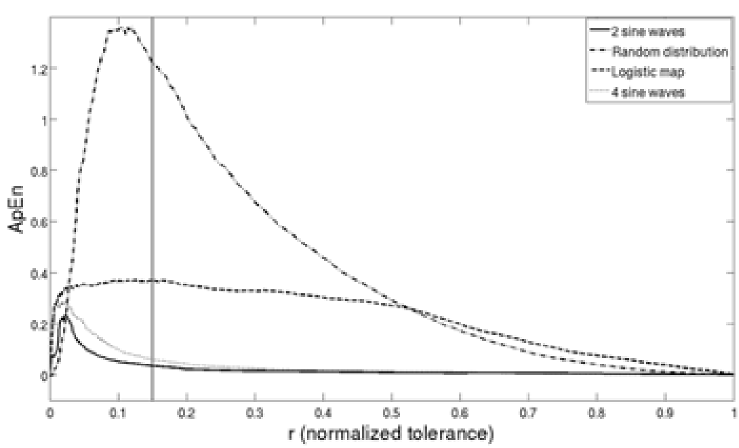

Fig. 2. ApEn (m = 2) along tolerances (r – normalized, see text) for different generating process

The vertical line denotes the typically used r = 0.15. Processes: sum of 2 sine waves (2S_a); uniformly distributed random numbers (U.D.R.N.); Logistic map (L3.7) and sum of 4 sine waves (S4). See Table 2 for nomenclature

Table 1. Some examples of subjectivity and inconsistencies found in the peak-ApEn estimator

Time-series size N	Window size m	Process	
A		*4S*	*SAT_a*
210	1	0.756 (0.054)*	0.324 (0.015)
210	3	0.192 (0.021)	0.310 (0.012)*
B		*AR_a*	*AR_b*
210	2	1.231 (0.014)*	1.198 (0.018)
300	2	1.317 (0.032)	1.327 (0.020)*
C		*1S_b*	*L3.7*
210	1	0.737 (0.085)*	0.488 (0.006)
240	1	0.733 (0.088)*	0.487 (0.006)
270	1	0.761 (0.067)*	0.488 (0.007)

*Subset A: classification changes with the change in the window size m. Subset B: classification changes with the change in the size N of the time-series. Subset C: a highly organized series classified as more complex (less organized) than a much more variable one. the values of peak-ApEn are presented as means (standard deviations) of 2N samples. * indicates a significant greater value with P < .01. See Table 2 for nomenclature*

Likewise, peak-ApEn is much more robust than ApEn and Sample ApEn. Nevertheless, some inconsistencies still persist as we describe next (and exemplify in Table 1). Firstly, there remains subjectivity in the choice of the value of *m*, since the classification may change depending on this parameter (Table 1 subset A). Secondly, there is a high dependency on the size of the vector analyzed and reversed results are obtained with small differences in size (Table 1 subset B). Finally, well-organized time-series (e.g., a sine wave) analyzed through only one period may present higher peak-ApEn values than a more variable series (Table 1 subset C).

In short, peak-ApEn and vApEn are much more reliable than ApEn, but the tools deserve further improvement.

2. A NEW TOOL: A1APEN

Here we propose an approach that might be considered as a step forward in relation to peak-ApEn and a step backward in relation to vApEn. a1ApEn is based on the construction of the area under the curve of ApEn versus tolerance *r** (see Fig. 2) and is defined for a time-series **S** of size N as:

$$\mathrm{a1ApEn}(\mathbf{S}) = \int_0^1 \mathrm{ApEn}\left(m = 1, r^*, N\right) dr^* \quad (4)$$

Where the integral stands for a numerical integration. The meaning of *r** is given in section 2.1. Why the use of the window *m = 1* is presented in section 2.2.

2.1 The Tolerance Vector *r* and the Normalized Tolerance Vector *r**

The first crucial step to obtain an accurate a1ApEn value is to construct a detailed tolerance vector, ***r***. The usual practice to establish a value of tolerance for comparison is to compute it as a fraction of the standard deviation of the time-series (see [1]). For instance, some figure between 15% and 25% of the standard deviation is the most typical choice. Here, we do not use the standard deviation as a milestone to compute *r* values.

Because the Heaviside distance d between two vectors $i = (i_1, i_2, \ldots, i_n)$ and $j = (j_1, j_2, \ldots, j_n)$ is given as $d(i,j) = \max(\mathrm{absolute}(i_k - j_k))$, $k = 1,2 \ldots n$, then, for the entire time-series data, there would be a pair of points (henceforth we will use "point" to refer to a certain datum value in the time-series **S**) that has a minimal distance greater than zero, and another pair that has a maximal distance (this will be absolute(max(**S**) – min(**S**))[1]. Why we are considering a minimal distance greater than zero becomes clear shortly.

The detailed tolerance vector **r**. Initially, the time-series is sorted (ascendingly) and the absolute difference between each pair of consecutive sorted points is used to create a delta-vector, D. From the delta-vector, the zeros are excluded, and D has size n (unknown before these procedures). Then, the tolerance *r* vector is constructed. The first value of ***r*** is zero.

[1] An exception for that is a binary series, since for r lower than some critical value, no matches can occur, while for r higher than that critical value, all the sub-vectors are equal; and ApEn should not be employed to characterize a binary series at all.

Therefore, for r_1 = $\boldsymbol{r}(1)$ = 0, only completely equal sub-vectors will count as a match. There are two paths to construct the subsequent *r* values.

If the time-series size N ≤ 300, ***r*** will have size = n+1, and

$$\mathbf{r}(x) = \sum_{z=1}^{x-1} D(z)$$

x = 2,3, … n+1, i.e., each value in ***r*** is the sum of the previous ones in D. Notice that the last value of the tolerance vector is *r*(n+1) = absolute (max(S) – min(S)), i.e., the maximal possible Heaviside distance among all the pairings in the original time-series.

If N > 300, then the first 51 values of ***r*** are obtained as described above. Then, depending on the fraction of the maximal distance that the sum of these 51th values reaches, different partitioning procedures are taken in order to populate the remaining of the tolerance vector ***r***. The main focus is to have a large number of *r* values up to the range of 35% of the maximal distance, since the ApEn versus *r** curve has its most significant part within this range (personal observations - see Fig. 2 as an example). At the same time, from 35% to 100% of the maximal distance, fewer values are needed, so computational time should not be too much affected. The ApEn values are computed using each *r* from the tolerance vector.

Then, the area is obtained through a numerical integration (eq. 4) using a normalized tolerance vector r* = *r*/max(*r*) as the basis. *This is very important in order to have comparisons on the same foot among different time-series*. Otherwise, high amplitude series would end up, inherently, with higher a1ApEn values.

2.2 The Window Size *m = 1*

If the window size is kept fixed, a1ApEn has no subjectivity in the choice of parameters for the analysis, a much desired condition. In this sense, the window size 1 is chosen because it can be shown that it results in the largest area compared to *m* = 2, 3, 4, … up to m << N. In other words, *m* = 1 gives a complete objectivity for the a1ApEn measure because one can fix such a window size beforehand, having the knowledge that this choice will result in the largest area for the time-series under analysis. We show this result below. Equation 2 can be rewritten as:

$$\phi(m,r) = \frac{1}{N-m+1} \ln\left[\prod_{1}^{N-m+1} \frac{\#_i(m)}{N-m+1}\right] \tag{5a}$$

or:

$$\phi(m,r) = \frac{\ln\left[\prod_{1}^{N-m+1} \#_i(m)\right]}{N-m+1} - \ln(N-m+1) \tag{5b}$$

Let X(m) be a mean count. Substituting this mean count in place of the true counts $\#_i$(m):

$$\ln\left[\prod \#_i(m)\right] = \ln\left[X(m)\right]^{N-m+1} \tag{6}$$

Then, inserting (6) in equation (5b), we obtain a mean ϕ, related to the mean counting:

$$\bar{\phi}(m,r) = \ln\left[X(m)\right] - \ln(N-m+1) \tag{7}$$

Consequently, from equation (3):

$$ApEn(m,r) \cong \bar{\phi}^{m}(r) - \bar{\phi}^{m+1}(r) \tag{8}$$

Replacing (7) in (8) results in an ApEn value related to the mean counts in each window:

$$\overline{ApEn}(m,r) = \ln\left[\frac{X(m)\cdot(N-m)}{X(m+1)\cdot(N-m+1)}\right] \tag{9}$$

Let us call $\overline{ApEn}$ as the expected ApEn value. If N→ ∞ is considered, then:

$$\overline{ApEn}(m,r) = \ln\left[\frac{X(m)}{X(m+1)}\right] \tag{10}$$

It is clear that the expected ApEn depends on how X(*m*) increases in proportion to X(*m+1*). It is not possible to define the rule for such a relation, but it is known that it must respect the following conditions: *r** = 0 implies X(m,0) = 1 and *r** = 1 implies X(m,1) = N. It is interesting that even an arbitrary rule may still give valuable information about the behavior of the curve ApEn versus tolerance. Consider the following formulation that obeys the above conditions and is able to reproduce the curves in Fig. 2:

$$X(m,r) = 1 + (N-1)\cdot r^{q(m)} \tag{11}$$

With this rule, it is possible to observe that if q(*m*) possess a linear behavior (i.e., q(*m*)=a+b·*m*), $\overline{ApEn}$ (*m* = 1) encompasses the $\overline{ApEn}$ for all other windows sizes (see Fig. 3). Furthermore, the only situation in which an area computed for $m = 2$ is greater than an area computed for $m = 1$ is when $q(3) / q(2) \gg q(2) / q(1)$. Generally, this last situation is not expected for $m << N$ since, for each inclusion of a new dimension in the state-space (i.e., m, $m+1$, $m+2$ ...), almost-always there would be a proportional decrease in the number of counts (C_i). This is particularly accurate for less organized time-series, given that the probability of decreasing the number of counts remains the same with the addition of a new dimension. Therefore, we are lead to conclude that the area under the curve of ApEn versus tolerance r for $m = 1$ is appropriate, remaining no subjectivity in the choice of the parameters for analysis.

2.3 Testing a1ApEn

Seventeen time-series from different generating processes were created *in machina* (Matlab R2013a). Table 2 presents a description of the generating processes.

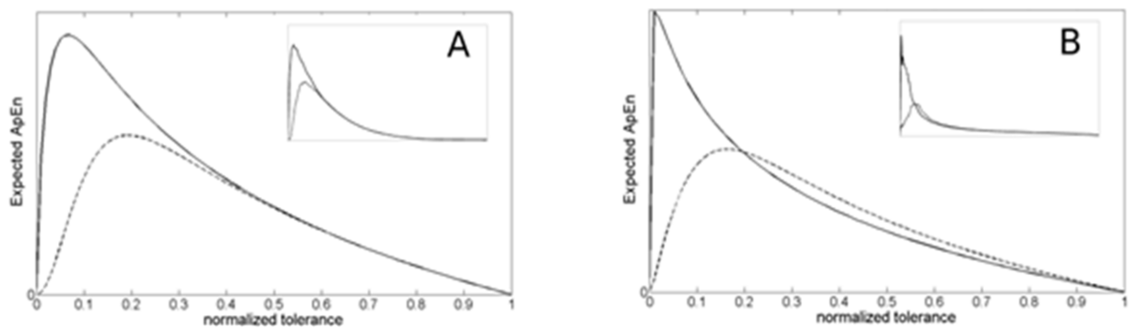

Fig. 3. Expected ApEn as a function of tolerance (equation (11))

Solid lines: m = 1; dotted lines: m = 2. (A) Factor q(1) = 1; q(2) = 2; q(3) = 3. (B) Factor q(1) = 0.25; q(2) = 1.5; q(3) = 3. The insets contain real ApEn values for normally distributed random numbers (in A) and a sine wave (in B). Notice the resemblance of the expected curves and the real ones. N = 200

Table 2. Designations, descriptions and some estimators of the prototypical processes (in alphabetical order)

Designation	Process/Description	Mean	Var	Min	Max
1S_a	Complete deterministic process of a sine wave of angular speed (ω) = 0.025, amplitude (A) = 1	0.029	0.502	-1.000	1.000
1S_b	Complete deterministic process of a sine wave; ω = 0.100, A = 1	0.001	0.498	-1.000	1.000
2S_a	Complete deterministic process formed by the sum of two sine waves: ω1 = 0.025, A = 0.2; ω2 = 0.100, A = 1	0.007	0.521	-1.185	1.185
2S_b	Complete deterministic process formed by the sum of two sine waves: ω1 = 0.025, A = 0.1; ω2 = 0.100, A = 1	0.030	1.015	-1.928	1.928
2S_c	Complete deterministic process formed by the sum of two sine waves: ω1 = 0.025, A = 1; ω2 = 0.100, A = 0.2	0.029	0.525	-1.142	1.142
4S	Complete deterministic process formed by the sum of four sine waves: ω1 = 0.025; ω2 = 0.100; ω3 = 0.051; ω4 = 0.078. A = 1 for all frequencies.	0.050	2.112	-3.394	3.241
AR_a	Second-order autoregressive model (parameters: 0.1; 0.499)	0.006	0.016	-0.538	0.377
AR_b	First-order autoregressive model (parameter: 0.4499)	0.002	0.003	-0.189	0.170
MA_a	5^{th}-order moving average model (parameters: 0.4, 0.2, 0.1, 0.1, 0.5)	0.085	0.457	-2.002	2.057

Designation	Process/Description	Mean	Var	Min	Max
MA_b	5th-order moving average model (parameters: 0.1, 0.7, 0.9, 0.4, 0.2)	0.025	1.640	-3.772	3.932
SAT_a	Skewed asymmetric tent map (parameters: a = 0.2, b = 0.4). Chaos region.	0.528	0.071	0.004	0.997
SAT_b	Skewed asymmetric tent map (parameters: a = 0.6, b = 0.1). Chaos region.	0.550	0.076	0.002	0.999
SAT_c	Skewed asymmetric tent map (parameters: a = 0.041, b = 0.9). Quasi-2 periodic solution.	0.501	0.211	0.000	1.000
L3.6	Logistic map (parameter = 3.6). Quasi-2 periodic solution.	0.647	0.049	0.324	0.900
L3.7	Logistic map (parameter = 3.7). Pomeau–Manneville scenario (transition to chaos due to intermittency).	0.667	0.042	0.257	0.925
N.D.R.N.	Normally distributed (pseudo)random numbers with zero mean and standard deviation of 1.	0.035	0.988	-3.797	3.321
U.D.R.N.	Uniformly distributed (pseudo)random numbers in the [0,1] interval.	0.500	0.083	0.000	1.000

Mean: mean value of the time-series; Var: variance; Min and Max: minimum and maximum values in the series. Notice that in the moving average model "MA_a", the extremities (i.e., the more recent and the more past events) have a higher weight in relation to model "MA_b"

For the purposes of testing the "positional sensitivity", each generated series had 360 points and a moving sampling window of N = 180 run along these 360 points. Therefore, we obtained 181 samples of each original series, and each sample had its ApEn (r^* = 0.15), peak-ApEn and a1ApEn values computed. The variances of the 181 values obtained for each estimator were, then, compared using a F-test for sample variance.

For the purposes of testing the consistency of the tool, we generated time-series of varying sizes (2N = 240, 420 and 600). A moving window of half size (N = 120, 210 and 300) run along the series. Therefore, for each N, we computed N values of the peak-ApEn and a1ApEn estimators. Two things are expected: a beforehand known well-behaved series should have lower estimators values, and, a series classified as less-organized for a given N should be classified as less-organized for a different N when compared to another series.

3. RESULTS AND DISCUSSION

3.1 Results of the Positional Sensitivity

The issue of positional sensitivity is still present in the a1ApEn. However, the variance of the moving sampling windows is significantly lower (all $P < .01$) than the ones obtained using peak-ApEn for all the 17 data sets analyzed. Regarding ApEn, the variance of a1ApEn was significantly lower in 16 cases ($P < .01$, Fig. 4A). The only exception was the tent map C (Fig. 4B). This time-series is poorly characterized by ApEn using $r > 0.15$ (Fig. 4C), and all moving sampling windows had a very low and similar ApEn value ($\cong 1.56 \times 10^{-5}$) and, therefore, a close to zero variance resulted. Hence, this exception reinforces the benefits of the a1ApEn approach indeed.

3.2 Results of the Consistency of a1ApEn

Tables 3 and 4 present mean values of peak-ApEn and a1ApEn, respectively, obtained for each process. The series are classified ascendingly accordingly to their mean values for N = 120. Table 5 presents the pairing comparison between the classification given by peak-ApEn and a1ApEn. Coincidences are highlighted. Notice that the extremities of the classifications are coincident between peak-ApEn and a1ApEn.

Table 6 shows a symbolic pattern of changes in classifications that would occur using peak-ApEn if, instead of sorting the processes using the 120 points size (Table 3), one had chosen the results from N = 210 or N = 300. There are two main columns, denoted by Δ_1 and Δ_2. Δ_1 is the difference between a line "j" and the line "j-1", in Table 3. Δ_2 is the difference between a line "j" and the line "j-2", in Table 3. The "+" symbol indicates that the process in line "j" maintains its classification in relation to the preceding process when 210 or 300 points are analyzed. On the other hand, the "-" symbol indicates that a change in classification would occur. Similarly, Table 7 shows the symbolic pattern of changes in classifications for a1ApEn.

Two points are to be noted. Firstly, changes in classification using peak-ApEn are much more pronounced than using a1ApEn. In fact, while the latter gives no second-degree changes (Δ_2), the former presents two changes in Δ_2. Secondly, while most of the changes in classification by peak-ApEn occur for non-coincident processes (Table 5), changes using a1ApEn occur at the more disorganized processes extremity. We discuss the importance of this fact shortly.

3.3 Discussion

ApEn is a wide employed estimator of organization (complexity) of time-series (cf., Introduction). As discussed in [14], ApEn was devised as an alternative method to approach short time-series (< 1,000 data points) originated from unknown underlying processes. At that time, there was a struggle in empirical time-series analysis regarding the recognition of strange attractors (deterministic chaos) from other processes and ApEn proved a valuable tool as an estimator of the rate of (informational) entropy production of a Markov chain approximating a given process (see [14]). However, as pointed out in a number of studies (cf. Introduction), this estimator suffers from consistency and objectivity. Fig. 2 illustrates well these problems and more robust tools are, therefore, relevant. In this study, we present a new estimator that fulfil this task, i.e., has no subjectivity and, at the same time, has consistency. To obtain the estimator, one constructs, numerically, the area under the curve of ApEn values for *m = 1,* obtained in a detailed range of tolerances, along a normalized tolerance vector. To supply the tool with a complete objective procedure, we show that almost-always the area obtained with the window size *m = 1* will be the largest one in relation to other window sizes greater than 1. Thus, in face of this, we name the estimator as a1ApEn.

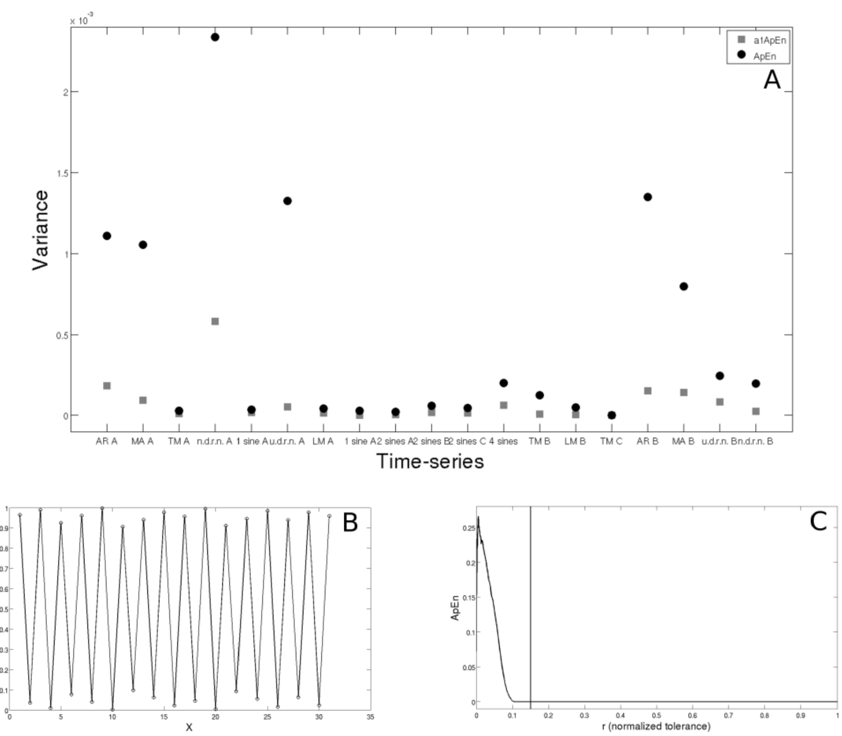

Fig. 4. Comparison of the variance of 17 data-sets with ApEn and a1ApEn

(A) Variance of 17 different time-series from a moving ApEn and a moving a1ApEn of 180 points over an original data set of 360 points. Notice that the variance of a1ApEn is smaller for all time-series but SAT_c. (B) A sample of 20 consecutive points from the skewed asymmetric tent map SAT_c. (C) ApEn of SAT_c for window size m=2 for all possible normalized tolerances r. Notice that the typically utilized r (0.15, the vertical line) does not properly cover informative values of ApEn*

Table 3. Mean peak-ApEn for time-series of a given process with size N

Process	Size (N)		
	120	**210**	**300**
SAT_c	0.252 (0.013)	0.256 (0.011)	0.257 (0.005)
1S_a	0.305 (0.198)	0.559 (0.078)	0.651 (0.023)
SAT_a	0.306 (0.010)	0.324 (0.017)	0.338 (0.007)
2S_c	0.318 (0.094)	0.482 (0.061)	0.548 (0.028)
L3.6	0.442 (0.017)	0.446 (0.006)	0.437 (0.011)
L3.7	0.478 (0.018)	0.488 (0.006)	0.487 (0.007)
SAT_b	0.586 (0.014)	0.606 (0.009)	0.621 (0.006)
2S_a	0.717 (0.028)	0.922 (0.041)	1.014 (0.037)
4S	0.722 (0.082)	0.756 (0.055)	0.952 (0.043)
1S_b	0.732 (0.070)	0.737 (0.085)	0.835 (0.033)
2S_b	0.787 (0.078)	0.953 (0.033)	1.034 (0.016)
MA_b	1.589 (0.031)	1.793 (0.019)	1.951 (0.023)
AR_a	1.670 (0.025)	1.951 (0.024)	2.157 (0.015)
AR_b	1.733 (0.020)	1.947 (0.022)	2.138 (0.039)
N.D.R.N.	1.749 (0.025)	2.052 (0.020)	2.202 (0.016)
MA_a	1.793 (0.022)	2.060 (0.017)	2.231 (0.020)
U.D.R.N.	1.902 (0.029)	2.165 (0.024)	2.333 (0.015)

Results sorted ascendingly by the values obtained for N = 120. Samples = N for each N (see text). Parenthesis: standard deviations. Processes as in Table 2

Table 4. Mean a1ApEn for time-series of a given process with size N

Process	Size (N)		
	120	**210**	**300**
SAT_c	0.011 (0.0005)	0.012 (0.0003)	0.012 (0.0003)
1S_a	0.011 (0.009)	0.021 (0.002)	0.025 (0.001)
2S_c	0.020 (0.009)	0.024 (0.003)	0.027 (0.001)
4S	0.050 (0.018)	0.041 (0.009)	0.046 (0.003)
2S_b	0.059 (0.011)	0.054 (0.003)	0.054 (0.001)
2S_a	0.076 (0.004)	0.075 (0.002)	0.076 (0.001)
1S_b	0.079 (0.001)	0.080 (0.001)	0.081 (0.001)
SAT_a	0.135 (0.007)	0.136 (0.004)	0.140 (0.002)
L3.6	0.143 (0.003)	0.149 (0.004)	0.153 (0.003)
L3.7	0.240 (0.003)	0.242 (0.002)	0.243 (0.001)
MA_b	0.258 (0.009)	0.240 (0.009)	0.240 (0.004)
SAT_b	0.277 (0.003)	0.277 (0.004)	0.274 (0.005)
AR_a	0.312 (0.022)	0.325 (0.014)	0.326 (0.012)
AR_b	0.319 (0.013)	0.302 (0.005)	0.306 (0.007)
N.D.R.N.	0.350 (0.015)	0.355 (0.010)	0.358 (0.012)
MA_a	0.353 (0.014)	0.352 (0.007)	0.334 (0.019)
U.D.R.N.	0.546 (0.006)	0.558 (0.007)	0.568 (0.008)

Results sorted ascendingly by the values obtained for N = 120. Samples = N for each N (see text). Parenthesis: standard deviations. Processes as in Table 2

It is expected that a time-series from a given generating process could have different values of a certain estimator depending on the sample interval. Our first goal was to show that a1ApEn is less sensitive to the sample interval than ApEn. Thus, the tool is not even objective and more consistent, but it is also more precise. Next, we proceed to show that the new estimator is more consistent than, and without the subjectivities of, another estimator derived from ApEn as well, namely, peak-ApEn. Some of the problems with this estimator are illustrate in Table 1.

The results from peak-ApEn ($m = 1$) and a1ApEn are shown in Tables 3 and 4, respectively. There, the time-series were sorted accordingly to the values obtained for each estimator with N = 120.

The first important point to be noted is that the extremities (i.e., low and high values of the estimators) are coincident. This indicates that, in general, these two tools are able to, apparently, recognize time organization in similar ways (Table 5).

Table 5. Comparison of the classification given by peak-ApEn and a1ApEn as presented in Tables 3 and 4

peak-ApEn	a1ApEn
SAT_c	SAT_c
1S_a	1S_a
SAT_a	2S_c
2S_c	4S
L3.6	2S_b
L3.7	2S_a
SAT_b	1S_b
2S_a	SAT_a
4S	L3.6
1S_b	L3.7
2S_b	MA_b
MA_b	SAT_b
AR_a	AR_a
AR_b	AR_b
N.D.R.N.	N.D.R.N.
MA_a	MA_a
U.D.R.N.	U.D.R.N.

Coincidences are highlighted in light orange

Table 6. Ordered by peak-ApEn Deltas 1 and 2

	Δ_1		Δ_2	
	210	**300**	**210**	**300**
SAT_c				
1S_a	+	+		
SAT_a	-	-	+	+
2S_c	+	+	-	-
L3.6	-	-	+	+
L3.7	+	+	+	-
SAT_b	+	+	+	+
2S_a	+	+	+	+
4S	-	-	+	+
1S_b	-	-	-	-
2S_b	+	+	+	+
MA_b	+	+	+	+
AR_a	+	+	+	+
AR_b	-	-	+	+
N.D.R.N.	+	+	+	+
MA_a	+	+	+	+
U.D.R.N.	+	+	+	+

On the other hand, when we address the issue of changes in classification that would occur if another size was employed, we find out that peak-ApEn presents much more changes than a1ApEn (Tables 6 and 7). This feature highlights that a1ApEn is a more robust tool for analysis.

As important as the issue of changes in classification above, is another fact. Despite of the size N, a1ApEn segregates the organized deterministic process of sine waves from the maps, while peak-ApEn mixes up the ordering of these different processes. As we stated, this is a very relevant issue since, even for small series (N = 120), a1ApEn can correctly identify different generating processes. At the same time, it should be noted that both tools correctly classify the random processes as the less organized ones.

Finally, plain inspection of Tables 3 and 4 reveals another significant result. As can be observed, the standard deviations of a1ApEn are all less elevated than those of peak-ApEn, with a 4.8-times lower median. This implies, as in the case of positional sensitivity, that a1ApEn is much less prone to variations than peak-ApEn when evaluating the organization of a time-series.

Table 7. Ordered by a1ApEn Deltas 1 and 2

	Δ_1		Δ_2	
	210	**300**	**210**	**300**
SAT_c				
1S_a	+	+		
2S_c	+	+	+	+
4S	+	+	+	+
2S_b	+	+	+	+
2S_a	+	+	+	+
1S_b	+	+	+	+
SAT_a	+	+	+	+
L3.6	+	+	+	+
L3.7	+	+	+	+
MA_b	-	-	+	+
SAT_b	+	+	+	+
AR_a	+	+	+	+
AR_b	-	-	+	+
N.D.R.N.	+	+	+	+
MA_a	-	-	+	+
U.D.R.N.	+	+	+	+

4. CONCLUSION

The analytical tool a1ApEn is consistent and has a completely objective procedure to address time-series temporal organization. The tool is able to discriminate adequately different generating processes and presents less variance than ApEn and peak-ApEn.

ACKNOWLEDGEMENTS

This study was funding by a research grant from Fundação de Amparo à Pesquisa do Estado de São Paulo – FAPESP #2014/08842-3. JESN was supported by a PhD fellowship from Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (Capes). The sponsors had no involvement in any stage of the study/ manuscript.

COMPETING INTERESTS

Authors have declared that no competing interests exist.

REFERENCES

1. Pincus SM. Approximate entropy as a measure of system complexity. Proc. Natl. Acad. Sci. 1991;88(6):2297–2301.
2. Santos BT, Martins RA, Natali JES, Rodrigues VH, Marques FS, Chauí-Berlinck JG. Consistency in approximate entropy given by a volumetric estimate. Chaos, Solitons & Fractals. 2009;42(1): 322–334.
3. Rukhin AL. Approximate entropy for testing randomness. J. Appl. Probab. 2000;37(1): 88–100.
4. Cuesta D, Varela M, Miró P, Galdós P, Abásolo D, Hornero R, Aboy M. Predicting survival in critical patients by use of body temperature regularity measurement based on approximate entropy. Med. Biol. Eng. Comput., 2007;45(7):671–8.
5. Yan R, Gao RX. Approximate entropy as a diagnostic tool for machine health monitoring. Mech. Syst. Signal Process. 2007;21(2):824–839.
6. Zhou P, Barkhaus PE, Zhang X, Rymer WZ. Characterizing the complexity of spontaneous motor unit patterns of amyotrophic lateral sclerosis using approximate entropy. J. Neural Eng. 2011; 8(6):066010.
7. Shin D-G, Yoo C-S, Yi S-H, Bae J-H, Kim Y-J, Park J-S, Hong G-R. Prediction of paroxysmal atrial fibrillation using nonlinear analysis of the R-R interval dynamics before the spontaneous onset of atrial fibrillation. Circ. J. 2006;70(1):94–99.
8. Hu X, Miller C, Vespa P, Bergsneider M. Adaptive computation of approximate entropy and its application in integrative analysis of irregularity of heart rate variability and intracranial pressure signals. Med. Eng. Phys. 2008;30(5):631–639.
9. Chen L, Luo W, Deng Y, Wang Z, Zeng S. Characterizing the complexity of spontaneous electrical signals in cultured neuronal networks using approximate entropy. IEEE Trans. Inf. Technol. Biomed. 2009;13(3):405–10.
10. Richman JS, Moorman JR. Physiological time-series analysis using approximate entropy and sample entropy. Am. J. Physiol. Heart Circ. Physiol. 2000;278(6): H2039–H2049.
11. Lu S, Chen X, Kanters JK, Solomon IC, Chon KH. Automatic selection of the threshold value R for approximate entropy. IEEE Trans. Biomed. Eng. 2008;55(8): 1966–72.
12. Chon K, Scully CG, Lu S. Approximate entropy for all signals. IEEE Eng. Med. Biol. Mag. 2009;28(6):18–23.
13. Castiglioni P, Di Rienzo M. How the threshold ‘r’ influences approximate entropy analysis of heart-rate variability. Comput. Cardiol. 2008;7:561–564.
14. Pincus SM, Huang W. Approximate entropy: Statistical properties and applications. Commun. Stat. - Theory Methods. 1992;21(11):3061–3077.

Peer-review history:
The peer review history for this paper can be accessed here:
http://sciencedomain.org/review-history/12742