On Abrupt Changes of the Approximate and Sample Entropy Values in Supercomputer Power Consumption

Jiří Tomčala
IT4Innovations, VSB - Technical University of Ostrava, 17.listopadu 2172/15, 70833 Ostrava-Poruba, Czech Republic
E-mail: jiri.tomcala@vsb.cz

Abstract. When calculating the Approximate Entropy (ApEn) and the Sample Entropy (SampEn) of a time series, representing the course of supercomputer power consumption, seemingly unexplained abrupt changes sometimes appear. These sudden changes occur when calculating the course of ApEn and SampEn values using a floating time window and can be manifested also in other discrete-valued time series, such as the development of prices on stock exchanges, time series from birth-death models, count-data time series, etc. These abrupt changes are not continuous and it is clear that they do not reflect real changes in the complexity degree of the analyzed time series. This can cause erroneous detection of a change in the degree of complexity of the time series, which can trigger a false alarm that something unusual is happening in the complex system being monitored (human brain, engine gearbox, financial market, supercomputer infrastructure, etc.). This work reveals in detail the mechanism of this phenomenon and also proposes measures to prevent its occurrence.

1. Introduction
One way to determine the status of a complex system is to analyze the time series that the system produces. In many cases, it is desirable to know the momentary degree of complexity of this system. A change in this degree of complexity is in many of these complex systems a sign that important changes are taking place inside.

An example of such a complex system is the human brain, where a decrease in the complexity of the measured electroencephalogram (EEG) indicates the onset of an epileptic seizure. A completely different example can be an engine gearbox, or in general any mechanical machine containing moving parts, where an increase in complexity of the measured vibration time series indicates an emerging mechanical failure.

In recent years, the Approximate Entropy (ApEn) [1] and especially the Sample Entropy (SampEn) [2] have been widely used algorithms to determine the degree of complexity of a time series. Recently, their accelerated modifications the Fast Approximate Entropy (FastApEn) and the Fast Sample Entropy (FastSampEn) have been introduced in [3]. They are designed for very fast detection of a change in the time series degree of complexity.

The term entropy is commonly used in many works such as [4], [5], [6], or [7]. However, it should be noted here that none of above mentioned algorithms calculates entropy in the mathematical sense, but only in various ways determine the degree of complexity of the analyzed...
time series. In addition, the resulting values of these algorithms are slightly different for the very same time series. This logically follows from the fact that each of these algorithms uses a different formula to calculate the result. Importantly, however, all of these algorithms can detect an increase and decrease in the level of complexity of the observed time series. For those interested, a detailed description of all these algorithms can be found in [3].

So from the above it is clear that the detection of a change in the complexity degree of the observed time series is very important. However, in the case of discrete-valued time series [8], sudden, step changes can occur in the calculations of the course of ApEn, SampEn, FastApEn, FastSampEn, which can be mistaken for changes in the complexity of these time series when automatically processing the results. An illustrative example of this phenomenon is shown in Figure 1, where the time series of the supercomputer power consumption is analyzed. The origin and conditions of this phenomenon are explained in the following sections.

At the end of this introduction, it would be appropriate to mention which time series are actually referred to as discrete-valued. These are time series that contain only values that are an integer multiple of some constant \emph{value discretization step}, which is the minimum value by which the value of a time series can change. An example of such a time series is shown in Figure 2.

At first glance, a chaotic cluster of data, but after magnification it turns out that it is a discrete-valued time series. This is again a time series of energy consumption of the supercomputer, which is displayed in a slightly different way and the discretization was caused by a measurement system that rounded the values of this time series to whole kilowatts.

In general, rounding is probably the most common cause of discrete-valued time series creation, however, some time series are discrete-valued due to the way they are generated. These are, for example, so-called count-data time series, where the system itself, which produces these discrete-valued time series, is discrete. Examples of these count-data time series could be time series of the development of the number of patients, the number of some product per hour, the number of cells in the monitored space, etc.

Of course, from a certain point of view, it could be said that all time series are discrete-valued time series, because they are all somehow rounded. This is true, but as will be shown later, if
Figure 2: Supercomputer power consumption as an example of a discrete-valued time series. When magnified, the discretization step of its value can be seen. The time series was normalized.

the discretization step of a time series value is much smaller (at least a thousand times) than the standard deviation of any floating time window used, this undesirable phenomenon should not occur.

2. Approximate and Sample Entropy

The original purpose of the approximate entropy was the analysis of medical data such as heart rate, neuroactivity, etc. Later, this algorithm was used to calculate the time series degree of complexity also in the field of finance, psychology, etc. Its value is calculated according to the following formula:

\[ ApEn(x, m, r) = \]

\[ = \frac{1}{N - m + 1} \left[ \sum_{i=1}^{N-m+1} \log \left( \frac{|j_i|}{N - m + 1} \right) \right] - \frac{1}{N - m} \left[ \sum_{i=1}^{N-m} \log \left( \frac{|k_i|}{N - m} \right) \right], \]

(1)

where

\[ j_i = \{ \xi \mid \| y_i - y_{i+1} \| \leq r \land \xi \in (1, N - m + 1) \}, \]

\[ k_i = \{ \xi \mid \| z_i - z_{i+1} \| \leq r \land \xi \in (1, N - m) \}, \]

\[ y_i = [x_i, x_{i+1}, ..., x_{i+m-1}], \quad z_i = [x_i, x_{i+1}, ..., x_{i+m}], \quad N = |x|. \]

This Equation (1) looks for similar sub-sequences \( y_i \) resp. \( z_i \) of lengths \( m \) resp. \( m + 1 \).

Currently, the more widely used algorithm for time series complexity analysis is Sample Entropy. This algorithm is a bit simpler than the previous Approximate Entropy algorithm. In 2000, Richman and Moorman proposed this algorithm to determine the degree of the physiological time series complexity.

The definition of Sample Entropy is:

\[ SampEn(x, m, r) = \log \left( \frac{\sum_{i=1}^{N-m+1} |h_i|}{\sum_{i=1}^{N-m} |a_i|} \right), \]

(2)
where
\[ b_i = \{ \xi \mid \|y_i - y_\xi\| \leq r \land \xi \in \{1, N - m + 1\} \setminus i \}, \]
\[ a_i = \{ \xi \mid \|z_i - z_\xi\| \leq r \land \xi \in \{1, N - m\} \setminus i \}, \]
\[ y_i = [x_i, x_{i+1}, ..., x_{i+m-1}], \quad z_i = [x_i, x_{i+1}, ..., x_{i+m}], \quad N = |x|. \]

Note that sets \( b_i \) and \( a_i \) are different from sets \( j_i \) and \( k_i \) of Approximate Entropy. They do not contain index \( i \) so there is a possibility that the sum of all \( |a_i| \) can be zero.

3. Explanation of the abrupt changes occurrence

To understand the mechanism of this phenomenon, it is necessary to understand the algorithm for calculating the value of ApEn, SampEn, FastApEn, and FastSampEn.

Simply put, all of these algorithms are finding out the number of mutually similar subsequences in the floating time window of the analyzed time series. The similarity of these subsequences is by these algorithms determined according to the fact that the maximum norm of their mutual distance is not greater than a predetermined value of the parameter \( r \) (radius).

In the scientific literature, it is recommended to set this parameter \( r \) to a multiple of the standard deviation. Most often, the value of \( r \) is set to 0.2 times the standard deviation, which is also the default value for the functions for calculating the above-mentioned algorithms, which are included in the R [9] software packages TSEntropies [10] and pracma [11].

Since the value of \( r \) is in this way tied to the standard deviation, which changes as the time window is shifted, the boundaries of the region in which the subsequences are considered to be similar also change. In most cases, nothing special happens with such a change in this region of similarity, but if the time series is discrete-valued and the boundary of the similarity region just exceeds an integer multiple of the discretization step of the time series value, a massive increase or decrease in number of similar subsequences may occur. In that case, there would also be a jump in the calculated value of the algorithm used, and thus the abrupt change phenomenon shown in Figure 1 would occur.

If an ordinary time series with a very small value discretization step shifts the boundary of the similarity region over an integer multiple of this value discretization step, there will be no massive change in the number of similar subsequences because the number of elements of such a time series with the same value is very small and therefore there are also very small number of subsequences with the same norm of distance.

However, if a discrete-valued time series with a big value discretization step shifts the boundary of the similarity region over an integer multiple of this value discretization step, there may occur a massive change in the number of similar subsequences because the number of elements of such a time series with the same value is big, hence there are also a lot of subsequences with the same norm of distance. In such a case, a very small change in the parameter \( r \) can cause a really big change in the number of similar sub-sequences and consequently also a big change in the resulting value of any of the above-mentioned algorithms. If such a small change in the parameter \( r \) is a result of a shift of the floating time window by one element of the time series, then the sudden jump will occur.

Both of these cases are illustrated in Figure 3, where the content of the previous paragraphs is clarified. This figure shows the change in the number of only a few similar sub-sequences, which would not cause a large resulting change, however, it should be noted that this search takes place for all sub-sequences, which may be hundreds or thousands, so as a result the total change in the number of similar sub-sequences can be massive.

The next Figure 4 shows the course of 0.2 times the standard deviation of the analyzed time series together with the marked levels of the multiple of the value discretization step. At times where 0.2 times the standard deviation reaches a value of some multiple of the value discretization step, sudden jumps in the ApEn and SampEn values should occur. It is clear
Figure 3: Comparison of what happens when the parameter \( r \) changes for an ordinary time series (3a) and for a discrete-value time series (3b). Dots represent sub-sequences of 2 samples length. Squares indicate the boundaries of the sub-sequence similarity region. While in the case 3a the change of the parameter \( r \) (from \( r_1 \) to \( r_2 \)) increased the number of similar sub-sequences by 1, in the case 3b the very same change increased the number of similar sub-sequences by 6. By the way, the discrete-value time series in 3b is in fact strongly rounded ordinary time series from 3a.

from Figure 1 that this is indeed the case, which confirms the above hypothesis of the origin of these abrupt changes. As can be seen from Figure 4, in our example of a supercomputer power consumption time series, the critical level is 0.00954, which is four times its value discretization step. The jumps in Figure 1 correspond exactly to the times when this level intersects the course of the graph in Figure 4.

4. How to prevent it

Probably the easiest way to prevent this abrupt changes phenomenon is to set the parameter \( r \) to some constant value for all floating time windows. The similarity region will then be the same for all calculations and its boundary will never exceed some multiple of the value discretization step. This is the best solution for time series whose standard deviation does not change much. In this case, it is advisable to set the parameter \( r \) for all floating time windows to 0.2 times the standard deviation of the whole time series.

This method of prevention was chosen when recalculating the course of the ApEn and SampEn value of the supercomputer power consumption time series, and the result is shown in Figure 5.

Another possibility is to reduce the value discretization step of the time series to a sufficiently low value. This can be done by interpolating the existing values of the time series, between which new values are inserted, corresponding to the expected course of the time series at a given location. This would reduce the smallest possible change in the entire time series and the jumps in the ApEn and SampEn values should then be smaller. The interpolation can then be linear, polynomial, spline, etc.

In this case of the supercomputer power consumption time series the linear interpolation was chosen. This interpolation was performed twice, each time with a different level of refinement.
Figure 4: The course of 0.2 times the standard deviation of the analyzed time series together with the marked levels of the multiple of the value discretization step. The floating time window was 1440 minutes.

Figure 5: The resulting course of the ApEn and SampEn value of the supercomputer power consumption time series after recalculation at a constant value of the parameter $r$ equal to 0.2 times the standard deviation of the whole time series. The floating time window was again 1440 minutes.

with a different number of inserted values. First, one value was inserted between each of the two existing samples, doubling the number of time series samples, and the result of the recalculation is presented in Figure 6.

In the second interpolation, seven values were inserted between each of the two existing samples, so that the number of time series samples increased eightfold and the result of the recalculation is shown in Figure 7.
Figure 6: The resulting course of the ApEn and SampEn value of the interpolated supercomputer power consumption time series. During interpolation, one value was inserted between each of the two existing samples. The floating time window was 1440 minutes.

Figure 7: The resulting course of the ApEn and SampEn value of the interpolated supercomputer power consumption time series. During interpolation, seven values were inserted between each of the two existing samples. The floating time window was 1440 minutes.

5. Conclusion

Based on the comparison of Figures 1 and 4, it can be said that the origin of these abrupt changes in the values of ApEn and SampEn was reliably elucidated in this work. What is not so clear is how to prevent it as effectively as possible.

The method of the constant parameter $r$, the result of which is shown in Figure 5, is well applicable to time series with not very variable standard deviation. This is also the case for the analyzed supercomputer power consumption time series and the result in Figure 5 is satisfactory.
But what if the value of the standard deviation changes significantly and another method will need to be used. Looking at the results of the second method (Figures 6 and 7), which reduces the value discretization step, it can be seen that the more values inserted by linear interpolation, the more the sudden jumps in the ApEn and SampEn values are reduced. When interpolated by doubling the number of samples in Figure 6, these sudden jumps are approximately half. Even, in the case of an eight-fold number of samples in Figure 7, these sudden changes hardly occur.

However, at the same time, the use of this second method also reduces the complexity of the analyzed time series, and thus the overall decrease in ApEn and SampEn values can be seen in Figures 6 and 7. This is quite expected, because in linear interpolation values are added on the basis of a certain, predefined rule, which increase the predictability of the time series and according to [12] this fact must then cause a decrease in its complexity.

The bigger disadvantage associated with the method of reducing the value discretization step is the greater computational demand and thus the longer computational time. This is, of course, due to the fact that the interpolated time series has a higher time density of samples, so that when analyzing the same time period, a larger amount of data needs to be processed.

Based on the above, it would probably be best to propose a new method that would be suitable for all time series and at the same time its computational requirement would be comparable to the calculation of the ApEn, SampEn, FastApEn, or FastSampEn course. This could be a suggestion for a possible direction for further work, following on from this article.

Finally, it should be noted that since the FastApEn and FastSampEn algorithms (mentioned at the beginning) are based on similar principles as ApEn and SampEn, the content of this work also applies to these accelerated versions. For the sake of clarity, however, only the results concerning the original versions of ApEn and SampEn were presented in this work. All calculations of these algorithms were performed using the TSEntropies package, the practical use of which can be found in [13].

Acknowledgments
This work was supported by The Ministry of Education, Youth and Sports from the Large Infrastructures for Research, Experimental Development, and Innovations project “e-INFRA CZ – LM2018140” and by SGC grant No. SP2020/137 “Dynamic system theory and its application in engineering”, VSB - Technical University of Ostrava, Czech Republic.

References
[1] Pincus S M 1991 Approximate entropy as a measure of system complexity Proc. Natl. Acad. Sci. 88 pp 2297–2301
[2] Richman J S and Moorman J R 2000 Physiological time-series analysis using approximate entropy and sample entropy Am. J. Physiol. Heart Circ. Physiol. 278 pp 2039–49
[3] Tomčála J 2018 Acceleration of time series entropy algorithms J Supercomput 75 pp 143–54
[4] Lampart M and Nagová J 2020 Movement Characteristics of a Model with Circular Equilibrium Chaos and Complex Systems, Springer Proceedings in Complexity ed Stavrinides S and Ozer M (Springer, Cham) pp 45–55
[5] Lampart M and Raith P 2010 Topological entropy for set valued maps Nonlinear Anal Theory Methods Appl 73 pp 1533–37
[6] Lampart M and Zapoměl J 2020 Dynamics of a non-autonomous double pendulum model forced by biharmonic excitation with soft stops Nonlinear Dyn. 99 pp 1909–21
[7] Halflar R 2019 Dynamical properties of Beeler–Reuter cardiac cell model with respect to stimulation parameters Int. J Comput Math 97 pp 498–507
[8] Weiss C H 2018 An Introduction to Discrete-Valued Time Series (John Wiley & Sons Ltd.) p 304
[9] R Core Team 2019 R: A Language and Environment for Statistical Computing R Foundation for Statistical Computing Available online on https://www.R-project.org (accessed 1 Sep 2020)
[10] Tomčála J 2018 TSEntropies: Time Series Entropies R package Available online on https://CRAN.R-project.org/package=TSEntropies (accessed 1 Sep 2020)
[11] Borchers H W 2019 Pracma: Practical Numerical Math Functions R package
Available online on https://CRAN.R-project.org/package=pracma (accessed 1 Sep 2020)
[12] Tomčala J 2020 Predictability and Entropy of Supercomputer Infrastructure Consumption Chaos and
Complex Systems, Springer Proceedings in Complexity ed Stavrinides S and Ozer M (Springer, Cham)
pp 59–66
[13] Tomčala J 2020 New Fast ApEn and SampEn Entropy Algorithms Implementation and Their Application
to Supercomputer Power Consumption Entropy 22 p 863