



HAL
open science

(Multiscale) Cross-Entropy Methods: A Review

Antoine Jamin, Anne Humeau-Heurtier

► **To cite this version:**

Antoine Jamin, Anne Humeau-Heurtier. (Multiscale) Cross-Entropy Methods: A Review. Entropy, 2020, 22 (1), pp.45. <10.3390/e22010045>. <hal-02428477>

HAL Id: hal-02428477

<https://univ-angers.hal.science/hal-02428477v1>

Submitted on 15 Feb 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY 4.0 - Attribution - International License

Review

(Multiscale) Cross-Entropy Methods: A Review

Antoine Jamin ^{1,2,*}  and Anne Humeau-Heurtier ² 

¹ COTTOS Médical, Allée du 9 novembre 1989, 49240 Avrillé, France

² LARIS – Laboratoire Angevin de Recherche en Ingénierie des Systèmes, Univesity Angers, 62 avenue Notre-Dame du Lac, 49000 Angers, France; anne.humeau@univ-angers.fr

* Correspondence: antoine.jamin@cottos.fr; Tel.: +33-252605954

Received: 9 December 2019; Accepted: 26 December 2019; Published: 29 December 2019

Abstract: Cross-entropy was introduced in 1996 to quantify the degree of asynchronism between two time series. In 2009, a multiscale cross-entropy measure was proposed to analyze the dynamical characteristics of the coupling behavior between two sequences on multiple scales. Since their introductions, many improvements and other methods have been developed. In this review we offer a state-of-the-art on cross-entropy measures and their multiscale approaches.

Keywords: cross-entropy; multiscale cross-entropy; asynchrony; complexity; coupling; cross-sample entropy; cross-approximate entropy; cross-distribution entropy; cross-fuzzy entropy; cross-conditional entropy

1. Introduction

To quantify the asynchronism between two time series, Pincus and Singer have adapted the approximate entropy algorithm to a cross-approximate entropy (cross-ApEn) method [1]. Then, other cross-entropy methods—that improve the cross-ApEn—have been developed [2–7]. Furthermore, additional cross-entropy methods have been introduced to quantify the degree of coupling between two signals, or the complexity between two cross-sequences [8–10]. Cross-entropy methods have recently been used in different research fields, including medicine [5,11,12], mechanics [13], and finance [7,10].

The multiscale approach of entropy measures was proposed by Costa et al. in 2002 to analyze the complexity of a time series [14]. In 2009, Yan et al. proposed a multiscale approach for cross-entropy methods to quantify the dynamical characteristics of coupling behavior between two sequences on multiple scale factors [15]. Then, other multiscale procedures have been published with different cross-entropy methods [16,17]. Multiscale cross-entropy methods have recently been used in different research fields, including medicine [18–21], finance [6,9], civil engineering [22], and the environment [23].

Cross-entropy methods and their multiscale approaches are used to obtain information on the possible relationship between two time series. For example, Wei et al. applied percussion entropy to the amplitude of digital volume pulse signals and changes in R-R intervals of successive cardiac cycles for assessing baroreflex sensitivity [18]. Results showed that the method is able to identify the markers of diabetes by the nonlinear coupling behavior of the two cardiovascular time series. Moreover, Zhu and Song computed cross-fuzzy entropy on a vibration time series to assess the bearing performance degradation process of motor [13]. Results showed that the method detects trend for bearing degradation process over the whole lifetime. In addition, Wang et al. applied multiscale cross-trend sample entropy to analyze the asynchrony between air quality impact factors (fine particulate matters, nitrogen dioxide, . . .), and air quality index (AQI) in different regions of China [23]. Results showed that the degree of synchrony between fine particulate matter and AQI is

higher than the other air quality impact factor which reveals that fine particulate matter has become the main source of air pollution in China.

Our paper presents a state-of-the-art in three sections: First, the cross-entropy methods are introduced. We detail, in the second section, different multiscale procedures. A multiscale cross-entropy generalization is presented and other specific multiscale cross-entropy algorithms are proposed in the third section.

2. Cross-Entropy Methods

In this section, we classify cross-entropy methods according to their entropy measures: Cross-approximate entropy, cross-sample entropy, and cross-distribution entropy. Other methods that use different cross-entropy-based measures are also detailed. Table 1 shows the twelve measures that are detailed in this section.

Table 1. Cross-entropy measures, in chronological order, that are presented in this review. Authors, year, reference, and section location are indicated for each item.

Method	Authors	Year	Ref.	Section
Cross-approximate entropy	Pincus and Singer	1996	[1]	Section 2.1.1
Cross-conditional entropy	Porta et al.	1999	[8]	Section 2.4.1
Cross-sample entropy	Richman and Moorman	2000	[2]	Section 2.2.1
Cross-fuzzy entropy	Xie et al.	2010	[3]	Section 2.4.2
Modified cross-sample entropy	Yin and Shang	2015	[4]	Section 2.2.2
Binarized cross-approximate entropy	Škorić et al.	2017	[5]	Section 2.1.2
Modified cross-sample entropy based on symbolic representation and similarity	Wu et al.	2018	[6]	Section 2.2.3
Kronecker-delta based cross-sample entropy	He et al.	2018	[7]	Section 2.2.4
Permutation based cross-sample entropy	He et al.	2018	[7]	Section 2.2.5
Cross-distribution entropy	Wang and Shang	2018	[9]	Section 2.3.1
Permutation cross-distribution entropy	He et al.	2019	[10]	Section 2.3.2
Cross-trend sample entropy	Wang et al.	2019	[23]	Section 2.2.6
Joint permutation entropy	Yin et al.	2019	[24]	Section 2.4.3

2.1. Cross-Approximate Entropy-Based Measures

2.1.1. Cross-Approximate Entropy

Cross-approximate entropy (cross-ApEn), introduced by Pincus and Singer [1], allows to quantify asynchrony between two time series. For two vectors \mathbf{u} and \mathbf{v} of length N , cross-ApEn is computed as:

$$\text{cross-ApEn}(m,r,N)(\mathbf{v}||\mathbf{u}) = \Phi^m(r)(\mathbf{v}||\mathbf{u}) - \Phi^{m+1}(r)(\mathbf{v}||\mathbf{u}), \quad (1)$$

where $\Phi^m(r)(\mathbf{v}||\mathbf{u}) = \frac{1}{N-m+1} \sum_{i=1}^{N-m+1} \log C_i^m(r)(\mathbf{v}||\mathbf{u})$ and $C_i^m(r)(\mathbf{v}||\mathbf{u})$ is the number of sequences, of m consecutive points, of \mathbf{u} that are approximately (within a resolution r) the same as sequences, of the same length, of \mathbf{v} . One major drawback of this approach is that $C_i^m(r)(\mathbf{v}||\mathbf{u})$ should not be equal to zero. This is why cross-ApEn is not really adapted for a short time series. Furthermore, it is direction-dependent because often $\Phi^m(r)(\mathbf{v}||\mathbf{u})$ is generally not equal to its direction conjugate $\Phi^m(r)(\mathbf{u}||\mathbf{v})$ [2]. The value of cross-ApEn computed from two signals can be interpreted as a degree of synchrony or mutual relationship.

2.1.2. Binarized Cross-Approximate Entropy

Binarized cross-approximate entropy (XBinEn), introduced by Škorić et al. [5] in 2017, is an evolution of cross-ApEn to quantify the similarity between two time series. It has the advantage of being faster than cross-ApEn. XBinEn encodes a time series divided into vectors of length m . For two vectors \mathbf{u} and \mathbf{v} of length N , the XBinEn algorithm follows these six steps:

1. Binary encoding series are obtained as:

$$x_i = \begin{cases} 0 & \text{if } u_{i+1} - u_i \leq 0 \\ 1 & \text{if } u_{i+1} - u_i > 0 \end{cases}, \quad y_i = \begin{cases} 0 & \text{if } v_{i+1} - v_i \leq 0 \\ 1 & \text{if } v_{i+1} - v_i > 0 \end{cases}, \quad (2)$$

where $i = 1, 2, \dots, N - 1$, $x_i \in \mathbf{X}_m^{(i)} = [x_i, x_{i+t}, \dots, x_{i+(m-1)t}]$, and $y_i \in \mathbf{Y}_m^{(i)} = [y_i, y_{i+t}, \dots, y_{i+(m-1)t}]$. The time lag t allows a vector decorrelation to be performed;

2. Vector histograms $N_X^{(m)}(k)$ and $N_Y^{(m)}(n)$ are computed as:

$$N_X^{(m)}(k) = \sum_{i=1}^{N-(m-1)t} I\left\{\sum_{l=0}^{m-1} x_{i+l \cdot t} \times 2^l = k\right\}, \quad N_Y^{(m)}(n) = \sum_{j=1}^{N-(m-1)t} I\left\{\sum_{l=0}^{m-1} y_{j+l \cdot t} \times 2^l = n\right\}, \quad (3)$$

where $k, n = 0, 1, \dots, 2^m - 1$, and $I\{\cdot\}$ is a function that is equal to 1 if the indicated condition is fulfilled;

3. The probability mass functions are obtained as:

$$P_X^{(m)}(k) = \frac{N_X^{(m)}(k)}{N - (m - 1)t}, \quad P_Y^{(m)}(n) = \frac{N_Y^{(m)}(n)}{N - (m - 1)t}, \quad (4)$$

where $k, n = 0, 1, \dots, 2^m - 1$;

4. A distance measure is applied:

$$d(\mathbf{X}_m^{(i)}, \mathbf{Y}_m^{(j)}) = \sum_{k=0}^{m-1} I\{x_{i+k \cdot t} \neq y_{j+k \cdot t}\}, \quad (5)$$

where $i, j = 1, \dots, N - (m - 1)t$;

5. The probability $p_k^m(r)$ that a vector is within the distance r from a particular vector is estimated:

$$p_k^m(r) = \Pr\{d(\mathbf{X}_m^{(k)}, \mathbf{Y}_m) \leq r\}; \quad (6)$$

6. XBinEn is finally obtained as:

$$\text{XBinEn}(m, r, N, t) = \Phi^{(m)}(r, N, t) - \Phi^{(m+1)}(r, N, t), \quad (7)$$

where $\Phi^{(m)}(r, N, t) = \sum_{k=0}^{2^m-1} P_X^{(m)}(k) \cdot \ln(p_k^m(r))$.

This method gives almost the same results as cross-ApEn for a non-short time series. However, it is computationally more efficient than cross-ApEn. Its main disadvantage is that it cannot identify small signal changes. XBinEn is adapted to environments where processor resources and energy are limited but it is not a substitute to cross-ApEn [5]. It is proposed when the cross-ApEn procedure cannot be applied. The value of XBinEn computed from two signals can be interpreted as a degree of relationship between a related pair of time series.

2.2. Cross-Sample Entropy-Based Measures

2.2.1. Cross-Sample Entropy

Cross-sample entropy (cross-SampEn) quantifies the degree of asynchronism of two time series. This method was introduced by Richman and Moorman in 2000 to improve the cross-ApEn limitations (see Section 2.1.1) [2]. Cross-SampEn is a conditional probability measure that quantifies the probability that a sequence of m consecutive points (called sample) of a time series \mathbf{u} —that matches another

sequence of the same length of another time series \mathbf{v} —will still match the other sequence when their length is increased by one sample ($m + 1$). For two vectors \mathbf{u} and \mathbf{v} , cross-SampEn is computed as:

$$\text{cross-SampEn}(m, r, N)(\mathbf{v}||\mathbf{u}) = -\ln \frac{A^m(r)(\mathbf{v}||\mathbf{u})}{B^m(r)(\mathbf{v}||\mathbf{u})}, \quad (8)$$

where m is the sample length, N is the vectors (\mathbf{u} and \mathbf{v}) length, $A^m(r)(\mathbf{v}||\mathbf{u})$ and $B^m(r)(\mathbf{v}||\mathbf{u})$ are, respectively, the probability that a sequence of \mathbf{u} and a sequence of \mathbf{v} will match for $m + 1$ and m points (within a tolerance r).

For two time series \mathbf{u} and \mathbf{v} of length N , cross-SampEn can also be described as:

$$\text{cross-SampEn}(\mathbf{u}, \mathbf{v}, m, r, N) = -\ln \frac{n^{(m+1)}}{n^{(m)}}, \quad (9)$$

where $n^{(m)}$ represents the total number of sequences of m consecutive points of \mathbf{u} that match with other sequences of m consecutive points of \mathbf{v} .

The main difference between cross-ApEn and cross-SampEn is that cross-SampEn shows relative consistency whereas cross-ApEn does not. Unlike cross-ApEn, cross-SampEn is not direction-dependent. However, cross-SampEn generates, sometimes, undefined values for short time series. The value of cross-SampEn computed from two time series can be interpreted as a measure of similarity of the two time series.

2.2.2. Modified Cross-Sample Entropy

Modified cross-sample entropy (mCSE), introduced by Yin and Shang in 2015, has been developed to detect the asynchrony of a financial time series [4]. Inspired by the generalized sample entropy, proposed by Silva and Murta, Jr. [25], the authors proposed to adapt this method to cross-SampEn. The method combines cross-SampEn and nonadditive statistics. For two vectors \mathbf{u} and \mathbf{v} of length N , mCSE is computed as:

$$\text{mCSE}(m, r, N) = -\log_q \frac{\sum_{i=1}^{N-m} n_i^{(m+1)}}{\sum_{i=1}^{N-m} n_i^{(m)}}, \quad (10)$$

where m is the sample length, q is the entropic index, and $n_i^{(m)}$ is the number of times that the distance between vectors $\mathbf{y}_m = \{v(i), v(i+1), \dots, v(i+m-1) : 1 \leq i \leq N-m+1\}$ and $\mathbf{x}_m = \{u(i), u(i+1), \dots, u(i+m-1) : 1 \leq i \leq N-m+1\}$ is less than or equal to the tolerance r . The distance is calculated with $d(x_m(i), y_m(i)) = \max\{|u(i+k) - v(j+k)| : 0 \leq k \leq m-1\}$.

The value of mCSE computed from two time series can be interpreted as a degree of synchrony between the two time series and it can illustrate some intrinsic relations between the two time series.

2.2.3. Modified Cross-Sample Entropy Based on Symbolic Representation and Similarity

Modified cross-sample entropy based on symbolic representation and similarity (MCSEBSS), introduced by Wu et al. in 2018, has been developed to quantify the degree of asynchrony of two financial time series with various trends (stock markets from different areas [6]). In comparison with cross-SampEn, this method reduces the probability of including undefined entropies and it is more robust to noise. For two vectors \mathbf{u} and \mathbf{v} of length N , MCSEBSS is computed as:

$$\text{MCSEBSS}(\mathbf{u}, \mathbf{v}, m, r, N) = -\ln \frac{n^{(m+1)}}{n^{(m)}}, \quad (11)$$

where m is the sample length and $n^{(m)}$ is the number of template matches by comparing $s(\mathbf{u}_m(i), \mathbf{v}_m(j))$ and r . For $\mathbf{u}_m = \{u(i+k)\}$ and $\mathbf{v}_m = \{v(i+k)\}$ ($0 \leq k \leq m-1$ and $1 \leq i \leq N-m$), the similarity function $s(u_m(i), v_m(j))$ is calculated as:

$$s(u_m(i), v_m(j)) = \frac{\# \text{ of } 1 \text{ in } \text{count}(i, j)}{m}, \quad 1 \leq i, j \leq N - m, \tag{12}$$

where $\text{count}(i, j)$ is obtained by the function f defined as:

$$f = \begin{cases} 1 & \text{if } u_m(i+k) = v_m(j+k) \\ 0 & \text{if } u_m(i+k) \neq v_m(j+k) \end{cases}, \quad 0 \leq k \leq m-1. \tag{13}$$

The parameter r must be fixed between $\frac{m-n}{m+1}$ and $\frac{m-n}{m}$, where n is the maximum number of zeros obtained with $\text{count}(i, j)$ to consider \mathbf{u} and \mathbf{v} similar.

The value of MCSEBSS computed from two time series can be interpreted as a degree of asynchrony of the two time series. A low cross-entropy value indicates a strong synchrony between two signals.

2.2.4. Kronecker-Delta-Based Cross-Sample Entropy

The Kronecker-delta-based cross-sample entropy (KCSE), introduced by He et al. in 2018, has been developed to define the dissimilarity between two time series [7]. KCSE is based on the Kronecker-delta function $\delta_{x,y}$ that returns 1 if two variables are equal and 0 otherwise. For two vectors \mathbf{u} and \mathbf{v} of length N , KCSE is calculated as:

$$\text{KCSE}(m) = -\ln \frac{B^{m+1}}{B^m}, \tag{14}$$

where $B^m = \frac{\sum_{i=1}^{N-m+1} \text{KrD}_{u_m(i), v_m(i)}}{N-m+1}$ and $B^{m+1} = \frac{\sum_{i=1}^{N-m} \text{KrD}_{u_{m+1}(i), v_{m+1}(i)}}{N-m}$. The dissimilarity, between $\mathbf{u}_m(i) = [u(i), u(i+1), \dots, u(i+m-1)]$ and $\mathbf{v}_m(i) = [v(i), v(i+1), \dots, v(i+m-1)]$, is calculated as:

$$\text{KrD}_{u_m(i), v_m(i)} = \frac{\delta_{u_m(i), v_m(i)} + \delta_{u_m(i+1), v_m(i+1)} + \dots + \delta_{u_m(i+m-1), v_m(i+m-1)}}{n}. \tag{15}$$

Authors show that KCSE is better to classify financial data than multidimensional scaling based on the Chebyshev distance method [7]. The value of KSCE computed from two time series can be interpreted as a degree of irregularity between the two time series.

2.2.5. Permutation-Based Cross-Sample Entropy

The permutation-based cross-sample entropy (PCSE), introduced by He et al. in 2018, is quite similar to KCSE (see Section 2.2.4) [7]. A permutation step has only been added. For two vectors \mathbf{u} and \mathbf{v} of length N , PCSE is calculated as:

$$\text{PCSE}(m) = -\ln \frac{B^{m+1}}{B^m}, \tag{16}$$

where $B^m = \frac{\sum_{i=1}^{N-m+1} \text{KrD}_{\text{permu}X_m(i), \text{permu}Y_m(i)}}{N-m+1}$ and $B^{m+1} = \frac{\sum_{i=1}^{N-m} \text{KrD}_{\text{permu}X_{m+1}(i), \text{permu}Y_{m+1}(i)}}{N-m}$. The KrD function is defined in Section 2.2.4. The two vectors $\text{permu}X_m(i)$ and $\text{permu}Y_m(i)$ are obtained by a permutation algorithm defined with the permutation entropy [26]. The Video S1 shows an example of a permutation algorithm.

PCSE shows better results than KCSE for synthetic data (ARFIMA model) [7]. However, the two approaches give the same results for financial data [7]. Authors show that KCSE is better to classify financial data than multidimensional scaling based on the Chebyshev distance method [7]. The value

of PCSE computed from two time series can be interpreted as degree of irregularity between the two time series.

2.2.6. Cross-Trend Sample Entropy

Inspired by MCSEBSS (see Section 2.2.3), Wng et al. developed the cross-trend sample entropy (CTSE) to quantify the synchronism between two time series with strong trends [23]. For two time series \mathbf{u} and \mathbf{v} of length N , CTSE is calculated with the following four steps algorithm:

1. The two time series are symbolized as:

$$U(j) = \begin{cases} 1 & \text{if } \tilde{u}(j) > u(j) \\ 0 & \text{otherwise} \end{cases}, \quad V(j) = \begin{cases} 1 & \text{if } \tilde{v}(j) > v(j) \\ 0 & \text{otherwise} \end{cases}, \quad 1 \leq j \leq N, \quad (17)$$

where \tilde{u} and \tilde{v} are, respectively, the trend of \mathbf{u} and \mathbf{v} obtained by polynomial fitting (linear, quadratic or higher order).

2. The template vectors \mathbf{u}_m and \mathbf{v}_m are constructed as:

$$\mathbf{u}_m(i) = \{U(i+k)\}, \quad \mathbf{v}_m(i) = \{V(i+k)\}, \quad (18)$$

where $0 \leq k \leq m-1$ and $1 \leq i \leq N-m$.

3. The similarity between $x_m(i)$ and $y_m(i)$ is calculated as:

$$d(x_m(i), y_m(i)) = \frac{\# \text{ of } 1 \text{ in } C_m(i)}{m}, \quad 1 \leq i \leq N-m, \quad (19)$$

where the i -th symbol vector C_m is determined with f , a symbolic function between two template vectors \mathbf{u}_m and \mathbf{v}_m , as:

$$f = \begin{cases} 1 & \text{if } u_m(i+k) = v_m(i+k) \\ 0 & \text{otherwise} \end{cases}, \quad 0 \leq k \leq m-1. \quad (20)$$

4. CTSE is finally computed as:

$$\text{CTSE}(u, v, r, N) = -\ln \frac{n^{(m+1)}}{n^{(m)}}, \quad (21)$$

where $n^{(m)}$ is obtained by comparing $d(x_m(i), y_m(j))$ within a tolerance r for $1 \leq i \leq N-m$.

CTSE has two advantages over MCSEBSS: It is more sensitive to the difference of dynamical characteristic between two signals, and it works well with signals with trends (linear, quadratic, cubic, and sinusoidal) [23]. The value of CTSE computed from two time series can be interpreted as an indicator of dynamical structure regarding the two time series with potential trends.

2.3. Cross-Distribution Entropy-Based Measures

2.3.1. Cross-Distribution Entropy

In 2018, Wang and Shang introduced the cross-distribution entropy (cross-DistEn) to quantify the complexity between two cross-sequences [9]. To generalize the standard statistical mechanics, the authors replaced the standard distribution entropy (DistEn) based on Shannon entropy by DistEn based on Tsallis entropy [9]. The authors showed that cross-DistEn better illustrates the relationships between two vectors than cross-SampEn does [9]. For two times series \mathbf{u} and \mathbf{v} of length N cross-DistEn follow these four steps:

1. The state-space is reconstructed by building $(N - m + 1)$ vectors $\mathbf{X}(i)$ and $\mathbf{Y}(i)$ with $\mathbf{X}(i) = \{u(i), u(i + 1), \dots, u(i + m - 1)\}$, $1 \leq i \leq N - m$, and $\mathbf{Y}(j) = \{v(j), v(j + 1), \dots, v(j + m - 1)\}$, $1 \leq j \leq N - m$. m is the intended size of the vectors $\mathbf{X}(i)$ and $\mathbf{Y}(i)$;
2. The distance matrix is built by defining the distance matrix $\mathbf{D} = \{d_{ij}\}$ with d_{ij} being the Chebyshev distance between the vectors $\mathbf{X}(i)$ and $\mathbf{Y}(j)$ defined as:

$$d_{ij} = \max\{|u_{i+k}^{(\tau)} - v_{j+k}^{\tau}|, 0 \leq k \leq m - 1\}; \quad (22)$$

3. The probability density is estimated by computing the empirical probability density function of the matrix \mathbf{D} by applying the histogram approach. If the histogram has M bins, the probability of each bin will be P_t with $1 \leq t \leq M$;
4. The cross-distribution entropy based on the Tsallis entropy is computed as:

$$\text{crossDistEn}(\mathbf{u}, \mathbf{v}) = \frac{1}{\ln(a)} \frac{1}{q-1} \left(1 - \sum_{t=1}^M P_t^q\right), \quad (23)$$

where q is the order of the Tsallis entropy and a the logarithm base of the entropy computation.

The main advantage of cross-DistEn is that it is adapted for short time series. With financial data, cross-DistEn illustrates better the relationship between signals than cross-SampEn [9]. The value of cross-DistEn computed from two time series can be interpreted as a degree of linkage of the two time series.

2.3.2. Permutation Cross-Distribution Entropy

The permutation cross-distribution entropy (PCDE), introduced by He et al. in 2019, is a variant of cross-DistEn (see Section 2.3.1) [10]. The permutation allows to characterize fluctuations and prevents the impact of spatial distances on results. The PCDE algorithm is the same as the one of cross-DistEn, detailed in Section 2.3.1. However, an additional step is added before step 2 to permute $\mathbf{X}(i)$ and $\mathbf{Y}(j)$ with the permutation algorithm mentioned in Section 2.2.5. The distance matrix is therefore constructed with the permuted vectors. The value of PCDE computed from two time series can be interpreted as a degree of dissimilarity between the two time series.

2.4. Other Cross-Entropy-Based Measures

2.4.1. Cross-Conditional Entropy

Cross-conditional entropy (CCE), introduced by Porta et al. in 1919, quantifies the degree of coupling between two signals [8]. A corrected conditional entropy has been introduced to improve the approximate entropy that suffers from limitations when a finite number of sample is considered [27]. CCE is an adaptation of the corrected conditional entropy. For two signals $\mathbf{u} = \{u(i), i = 1, \dots, N\}$ and $\mathbf{v} = \{v(i), i = 1, \dots, N\}$, CCE is computed as:

$$\text{CCE}_{\mathbf{v}/\mathbf{u}}(L) = - \sum_{L-1} p(\mathbf{u}_{L-1}) \sum_{i/L-1} p(v(i)/\mathbf{u}_{L-1}) \times \log p(v(i)/\mathbf{u}_{L-1}), \quad (24)$$

where L is the length of the pattern extracted to be compared, $p(\mathbf{u}_{L-1})$ is the joint probability of the pattern $\mathbf{u}_{L-1}(i) = (u(i), u_{L-1}(i - 1))$, and $p(v(i)/\mathbf{u}_{L-1})$ is the probability of the sample $v(i)$ given the pattern $\mathbf{u}_{L-1}(i)$. If a mixed pattern, composed by $L - 1$ samples, of \mathbf{u} and \mathbf{v} : $(v(i), u(i), \dots, u(i - L + 2)) = (v(i), \mathbf{u}_{L-1})$, is defined and with the Shannon entropy $E(\mathbf{u}_L) = - \sum_L p(\mathbf{u}_L) \log p(\mathbf{u}_L)$, CCE can also be described as:

$$\text{CCE}_{\mathbf{v}/\mathbf{u}}(L) = E(v(i), \mathbf{u}_{L-1}) - E(\mathbf{u}_{L-1}). \quad (25)$$

For a limited amount of samples, the approximation of CCE always decreases to zero while increasing L . To solve this problem, a modification has been introduced as:

$$CCE_{\mathbf{v}/\mathbf{u}}(L) = \widehat{CCE}_{\mathbf{v}/\mathbf{u}}(L) + \text{perc}_{\mathbf{v}/\mathbf{u}}(L) \times \widehat{E}(\mathbf{v}), \tag{26}$$

where $\text{perc}_{\mathbf{v}/\mathbf{u}}$ is the ratio of mixed patterns found only once over the total number of mixed patterns, $\widehat{CCE}_{\mathbf{v}/\mathbf{u}}(L)$ and $\widehat{E}(\mathbf{v})$ are, respectively, the estimates of the $CCE_{\mathbf{v}/\mathbf{u}}(L)$ and $E(\mathbf{v})$ based on the considered limited dataset.

CCE can be defined as a measure of unpredictability of one signal when the second is observed because it quantifies the amount of information carried by one signal which cannot be derived from the other. It is not fully a measure of synchronization. The main disadvantage of CCE is that it is not totally adapted for short time series.

2.4.2. Cross-Fuzzy Entropy

Cross-fuzzy entropy (C-FuzzyEn), introduced by Xie et al. in 2010 [3], is an adaptation of fuzzy entropy, introduced by Chen et al. [28], that quantifies the synchrony or similarity of patterns between two signals [3]. C-FuzzyEn is an improvement of cross-SampEn that is more adapted for short time series and more robust to noise. For two times series \mathbf{u} and \mathbf{v} of length N , C-FuzzyEn is obtained with the following three steps algorithm:

1. The distance d_{ij}^m between \mathbf{X}_i^m and \mathbf{Y}_j^m is computed as:

$$d_{ij}^m = d[\mathbf{X}_i^m, \mathbf{Y}_j^m] = \max_{k \in (0, m-1)} |u(i+k) - \bar{u}(i) - v(j+k) - \bar{v}(i)|, \tag{27}$$

where m is the number of consecutive data to compare, $\mathbf{X}_i^m = \{u(i), u(i+1), \dots, u(i+m-1)\} - \bar{u}(i)$, and $\mathbf{Y}_j^m = \{v(j), v(j+1), \dots, v(j+m-1)\} - \bar{v}(j)$. $\bar{u}(i)$ and $\bar{v}(j)$ are calculated as: $\bar{u}(i) = \frac{1}{m} \sum_{l=0}^{m-1} u(i+l)$, and $\bar{v}(j) = \frac{1}{m} \sum_{l=0}^{m-1} v(j+l)$;

2. The synchrony or similarity degree D_{ij}^m is computed as: $D_{ij}^m = \mu(d_{ij}^m, n, r)$, where $\mu(d_{ij}^m, n, r)$ is the fuzzy function obtained as:

$$\mu(d_{ij}^m, n, r) = \exp - \frac{(d_{ij}^m)^n}{r}, \tag{28}$$

where r and n determine the width and the gradient of the boundary of the exponential function, respectively;

3. Finally, C-FuzzyEn is computed as:

$$\text{C-FuzzyEn}(m, n, r) = \ln \Phi^m - \ln \Phi^{m+1}, \tag{29}$$

where $\Phi^m = \frac{1}{N-m} \sum_{i=1}^{N-m} (\frac{1}{N-m} \sum_{j=1}^{N-m} D_{ij}^m)$, and $\Phi^{m+1} = \frac{1}{N-m} \sum_{i=1}^{N-m} (\frac{1}{N-m} \sum_{j=1}^{N-m} D_{ij}^{m+1})$.

The value of C-FuzzyEn computed from two time series can be interpreted as the synchronicity of patterns.

2.4.3. Joint-Permutation Entropy

Joint permutation entropy (JPE), introduced by Yin et al. in 2019, quantifies the synchronism between two time series. It is based on permutation entropy that consists of comparing neighboring values of each point and mapping them to ordinal patterns to quantify the complexity of a signal [26]. For two signals \mathbf{u} and \mathbf{v} , JPE is computed as the Shannon entropy of the $d! \times d!$ distinct motif combinations $\{(\pi_i^{d,t}, \pi_j^{d,t})\}$:

$$\text{JPE}(d, t) = - \sum_{i,j: (\pi_i^{d,t}, \pi_j^{d,t})} p(\pi_i^{d,t}, \pi_j^{d,t}) \cdot \ln p(\pi_i^{d,t}, \pi_j^{d,t}), \tag{30}$$

where d is the embedded dimension and $p(\pi_i^d, \pi_j^d)$ is the joint probability of $\{(\pi_i^{d,t}, \pi_j^{d,t})\}$ appearing in the $\mathbf{X}_l^{d,t} = \{u_l, u_{l+t}, \dots, u_{l+(d-1)t}\}$ and $\mathbf{Y}_l^{d,t} = \{v_l, v_{l+t}, \dots, v_{l+(d-1)t}\}$ and it is defined as:

$$p(\pi_i^{d,t}, \pi_j^{d,t}) = \frac{||l : l \leq T, \text{type}(\mathbf{X}_l^{d,t}, \mathbf{Y}_l^{d,t}) = (\pi_i^{d,t}, \pi_j^{d,t})||}{T}, \tag{31}$$

where $T = N - (d - 1)t$, $\text{type}(\cdot)$ corresponds to the map from pattern space to symbol space, and $|| \cdot ||$ corresponds to the cardinality of a set.

The main advantages of JPE are the simplicity, the robustness, and the low computational cost. The value of JPE computed from two time series can be interpreted as a degree of correlation between the two time series [29].

3. Multiscale Procedures

To study entropy or cross-entropy measures of time series across scales, a multiscale procedure can be used. In this part we detail, in chronological order, three multiscale methods: The coarse-grained, the time-shift, and the composite coarse-grained approaches.

3.1. Coarse-Graining Procedure

In 2002 Costa et al. introduced the coarse-graining procedure to analyze the complexity, defined by the analysis of the irregularity through scale factors [14]. This method is an improvement, more adapted for a biological time series, of the coarse-graining procedure introduced by Zhang [30]. This procedure has been used in multiscale entropy and cross-entropy methods [6,9,15,20,31–33]. For each scale factor, this procedure derives a set of vectors illustrating the system dynamics. For a monovariate discrete signal \mathbf{x} of length N , the coarse-grained time series $\mathbf{y}^{(\tau)}$ is calculated as:

$$y_j^{(\tau)} = \frac{1}{\tau} \sum_{i=(j-1)\tau+1}^{j\tau} x_i, \tag{32}$$

where τ is the scale factor and $1 \leq j \leq \frac{N}{\tau}$. The length of the coarse-grained vector is $\frac{N}{\tau}$. An example of coarse-graining procedure is presented in Figure 1A.

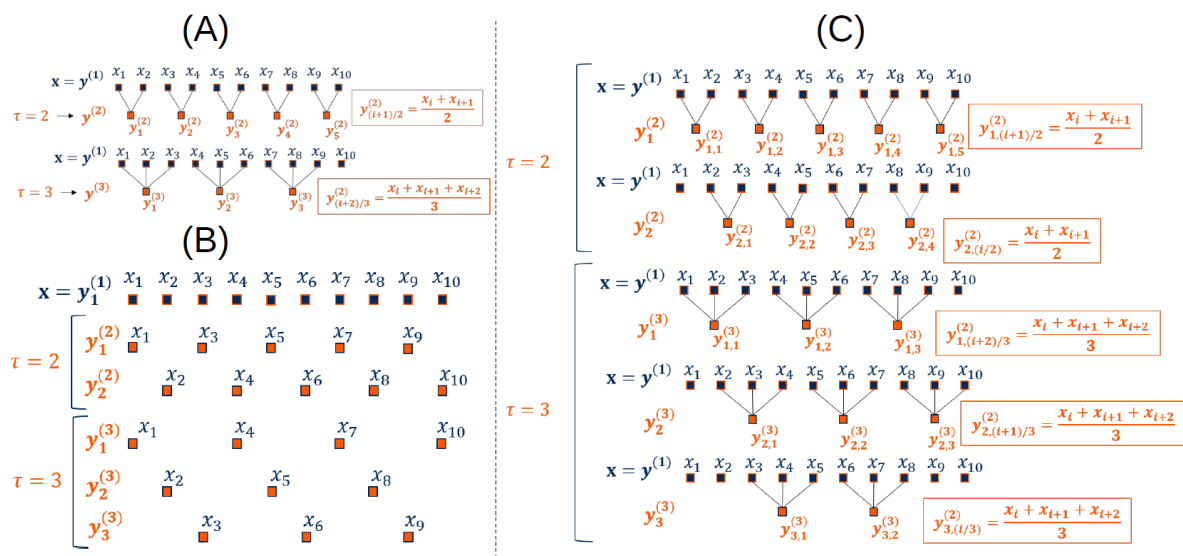


Figure 1. Examples of multiscale procedures for the ten first points of a time series \mathbf{x} . (A) represents the coarse-graining procedure (modified from [34]), (B) shows the time-shift procedure, and (C) illustrates the composite coarse-graining procedure (modified from [35]).

3.2. Time-Shift Procedure

As for the coarse-grained procedure, the time-shift procedure is used to decompose a signal through different scale factors and to perform a multiscale analysis. While coarse-graining procedure uses the averaging of time series on several interval scales, the time-shift procedure applies time shifting in time series. The main disadvantage of a coarse-graining procedure is the loss of pattern information hidden in the time series. To overcome this limitation, Pham used the Higuchi's fractal dimension (HFD) [36] and proposed a new multiscale analysis [37]. The time-shift procedure illustrates the fractal dimension of a signal. This method has been recently used with entropy and cross-entropy measures [17,37–39]. HFD shows stable numerical results for stationary, non-stationary, deterministic, and stochastic time series [40]. For a monovariate discrete signal \mathbf{x} of length N , the β time-shift signal $\mathbf{y}_\beta^{(\tau)}$ is calculated as:

$$\mathbf{y}_\beta^{(\tau)} = (x_\beta, x_{\beta+\tau}, \dots, x_{\beta+\lfloor \frac{N-\beta}{\tau} \rfloor \tau}). \quad (33)$$

For each time scale τ , β time-shift time series are computed ($\beta = 1, 2, \dots, \tau$). An illustration of the time-shift procedure is presented in Figure 1B.

3.3. Composite Coarse-Graining Procedure

The coarse-graining procedure, introduced by Costa et al. [14], increases the variance of estimated entropy values at large scale. To overcome this limitation, by Wu et al. introduced in 2013 a composite coarse-graining procedure [35]. This method has been used with entropy and cross-entropy measures [16,32]. For a monovariate discrete signal \mathbf{x} of length N , the k -th composite coarse-grained time series $\mathbf{y}_k^{(\tau)}$ is computed as:

$$y_{k,j} = \frac{1}{\tau} \sum_{i=(j-1)\tau+k}^{j\tau+k-1} x_i, \quad (34)$$

where $1 \leq j \leq \frac{N}{\tau}$. For each time scale τ , k composite coarse-grained time series are computed ($1 \leq k \leq \tau$). An illustration of the composite coarse-graining procedure is presented in Figure 1C.

4. Multiscale Cross-Entropy Methods

4.1. Generalization

Multiscale cross-entropy (MCE) methods consist of applying a cross-entropy measure for each scale factor obtained by a specific procedure. For each scale factor τ , MCE is computed as:

$$MCE(\mathbf{X}^{(\tau)}, \mathbf{Y}^{(\tau)}) = \frac{1}{k} \sum_{\beta=1}^k \text{crossEn}(\mathbf{X}_\beta^{(\tau)}, \mathbf{Y}_\beta^{(\tau)}), \quad (35)$$

where $\mathbf{X}^{(\tau)}$ and $\mathbf{Y}^{(\tau)}$ are computed with a multiscale procedure (see Section 3), k is the number of time series that are generated by the multiscale procedure ($k = 1$ for the coarse-graining procedure and $k = \tau$ for the time-shift and the composite coarse-graining procedures), and crossEn is the cross-entropy method used (see Section 2). Table 2 shows the multiscale cross-entropy methods that can be generalized with Equation (35). Before the computation of MCE, a pre-treatment can be performed. For example, the asymmetric multiscale cross-SampEn (AMCSE) method [33] decomposes each signal into two, one for the positive trends and the other for the negative trends, before applying a coarse-graining procedure and cross-SampEn.

Table 2. Multiscale cross-entropy methods, in chronological order, that can be generalized with Equation (35). For each method, the multiscale procedure and the cross-entropy measure used and the reference are mentioned.

Method	Multiscale Procedure	Cross-entropy Measure	Reference
Multiscale cross-SampEn	Coarse-grained	cross-SampEn	Yan et al., 2009 [15]
Multiscale cross-ApEn	Coarse-grained	cross-ApEn	Wu et al., 2013 [31]
Asymmetric multiscale cross-SampEn	Coarse-grained	cross-SampEn	Yin and Shang, 2015 [33]
Composite multiscale cross-SampEn	Composite coarse-grained	cross-SampEn	Yin et al., 2016 [16]
Multiscale cross-DistEn	Coarse-grained	cross-DistEn	Wang and Shang, 2018 [9]
Modified multiscale cross-SampEn based on symbolic representation and similarity	Coarse-grained	MCSEBSS	Wu et al., 2018 [6]
Modified multiscale cross-SampEn	Coarse-grained	mCSE	Castiglioni et al., 2019 [20]
Time-shift multiscale cross-SampEn	Time-shift	cross-SampEn	Jamin et al., 2019 [17]
Time-shift multiscale cross-DistEn	Time-shift	cross-DistEn	Jamin et al., 2019 [17]
Multiscale cross-trend SampEn	Coarse-grained	CTSE	Wang et al., 2019 [23]
Multiscale joint permutation entropy	Coarse-grained	JPE	Yin et al., 2019 [24]

4.2. Particular Cases

Some multiscale cross-entropy methods cannot follow the generalization previously introduced. In this part we detail three particular methods, in chronological order: The adaptive multiscale cross-SampEn, the refined composite multiscale cross-SampEn, and the percussion entropy.

4.2.1. Adaptive Multiscale Cross-Sample Entropy

The adaptive multiscale cross-sample entropy (AMCSE), introduced by Hu and Liang in 2011, assesses the nonlinear interdependency between different visual cortical areas [41]. The method uses the multivariate empirical mode decomposition (MEMD), introduced by Rehman and Mandic [42], to decompose two time series into intrinsic mode functions (IMFs) that represent the oscillation mode embedded in the data. For two time series \mathbf{u} and \mathbf{v} , AMCSE is calculated with the following three steps algorithm:

1. The MEMD on \mathbf{u} and \mathbf{v} is performed to obtain N IMFs;
2. The scales of data are computed in two directions, fine-to-coarse S_{f2c}^τ and coarse-to-fine S_{c2f}^τ , with the following two equations:

$$S_{f2c}^\tau = \sum_{i=\tau}^N \text{IMF}_i, \quad (\tau \leq N), \quad (36)$$

$$S_{c2f}^\tau = \sum_{i=1}^{N+1-\tau} \text{IMF}_i, \quad (\tau \leq N). \quad (37)$$

The two directions can be used separately or used in tandem to reveal the underlying dynamics of complex time series;

3. For each scale factor τ , the cross-SampEn (see Section 2.2.1) is applied between the two scales of data (S_{f2c}^τ and S_{c2f}^τ) extracted from vectors \mathbf{u} and \mathbf{v} .

4.2.2. Refined Composite Multiscale Cross-Sample Entropy

Yin et al. introduced in 2016 the composite multiscale cross-sample entropy (CMCSE) that follows the generalization (see Section 4.1), where the composite coarse-graining procedure and cross-SampEn are used [16]. The main disadvantage of this method is that cross-SampEn generates some undefined values when the number of matched sample is zero. To overcome this limitation, Yin et al. introduced

the refined CMCSE (RCMCSE). This method leads to better results with short time series. For two times series \mathbf{u} and \mathbf{v} of length N , RCMSE is computed with the following three steps algorithm:

1. Coarse-grained time series are obtained with the composite coarse-graining procedure detailed in Section 3.3;
2. For a scale factor τ , the number of matched vector pairs, $n_{k,\tau}^m$ and $n_{k,\tau}^{m+1}$, are calculated for all coarse-grained vectors
3. For each scale factor τ , RCMCSE is computed as:

$$\text{RCMCSE}(\mathbf{u}, \mathbf{v}, \tau, m, r) = -\ln \frac{\sum_{k=1}^{\tau} n_{k,\tau}^{m+1}}{\sum_{k=1}^{\tau} n_{k,\tau}^m}, \tag{38}$$

where m is the dimension and of the matched vector pairs and r is the distance tolerance for the matched vector pairs.

4.2.3. Percussion Entropy

Wu et al. introduced, in 2013, the multiscale small-scale entropy index (MEI_{SS}) that is obtained by summing the values of entropy for the first five scale factors [43]. Percussion entropy, introduced by Wei et al. in 2019, allows one to quantify a percussion entropy index (PEI) [18]. The method has been introduced to assess baroreflex sensitivity. PEI compares the similarity in tendency of change between two time-series. This index has been compared to MEI_{SS} . For two time series \mathbf{u} and \mathbf{v} of length N , PEI is computed with the following three steps algorithm:

1. A binary transformation of \mathbf{u} and \mathbf{v} is used to obtain $\mathbf{x} = \{x_1, x_2, \dots, x_{N-1}\}$ and $\mathbf{y} = \{y_1, y_2, \dots, y_{N-1}\}$:

$$x_i = \begin{cases} 0 & u(i+1) \leq u(i) \\ 1 & u(i+1) > u(i) \end{cases} \quad y_i = \begin{cases} 0 & v(i+1) \leq v(i) \\ 1 & v(i+1) > v(i) \end{cases}; \tag{39}$$

2. The percussion rate for each scale factor τ is computed as:

$$P_{\tau}^m = \frac{1}{n - m - \tau + 1} \sum_{i=1}^{n-m-\tau+1} \text{count}(i), \tag{40}$$

where m is the embedded dimension vectors and $\text{count}(i)$ represents the match number between $\mathbf{A}(i) = \{x_i, x_{i+1}, \dots, x_{i+m-1}\}$ and $\mathbf{B}(i + \tau) = \{y_{i+\tau}, y_{i+\tau+1}, \dots, y_{i+\tau+m-1}\}$;

3. PEI is calculated as:

$$\text{PEI}(m, n_{\tau}) = \phi^m - \phi^{m+1}, \tag{41}$$

where $\phi^m = \ln \sum_{\tau=1}^{n_{\tau}} P_{\tau}^m$ and n_{τ} is the number of scales to consider. Wei et al. [18] have chosen $n_{\tau} = 5$ in accordance with MEI_{SS} .

This algorithm is a generalization of the method developed by Wei et al. [18] for a specific time series, amplitudes of successive digital volume pulse signals and changes in R-R intervals of successive cardiac cycles. At the moment, it has not been used to process other kinds of signals.

5. Conclusions

In this review we proposed a state-of-the-art of cross-entropy measures, multiscale procedures, and multiscale cross-entropy methods. Multiscale cross-entropy methods offer other interesting perspectives for time series analysis. Furthermore, all the cross-entropy methods, detailed in this review, can be translated into multiscale cross-entropy methods with the multiscale procedures presented in this review.

Supplementary Materials: The following are available online at <http://www.mdpi.com/1099-4300/22/1/45/s1>, Video S1: Permutation entropy—An example to obtain permutation vectors.

Author Contributions: Investigation, A.J. and A.H.-H.; supervision, A.H.-H.; writing—original draft, A.J.; writing—review and editing, A.H.-H. All authors have read and agreed to the published version of the manuscript.

Funding: A CIFRE grant N°2017/1165 was awarded by ANRT to the company COTTOS Médical to support the work of graduate student A.J.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Pincus, S.; Singer, B.H. Randomness and degrees of irregularity. *Proc. Natl. Acad. Sci. USA* **1996**, *93*, 2083–2088. [[CrossRef](#)] [[PubMed](#)]
2. Richman, J.S.; Moorman, J.R. Physiological time-series analysis using approximate entropy and sample entropy. *Am. J. Physiol. Heart Circ. Physiol.* **2000**, *278*, H2039–H2049. [[CrossRef](#)] [[PubMed](#)]
3. Xie, H.B.; Zheng, Y.P.; Guo, J.Y.; Chen, X. Cross-fuzzy entropy: A new method to test pattern synchrony of bivariate time series. *Inf. Sci.* **2010**, *180*, 1715–1724. [[CrossRef](#)]
4. Yin, Y.; Shang, P. Modified cross sample entropy and surrogate data analysis method for financial time series. *Phys. A Stat. Mech. Appl.* **2015**, *433*, 17–25. [[CrossRef](#)]
5. Škorić, T.; Mohamoud, O.; Milovanovic, B.; Japundzic-Zigon, N.; Bajic, D. Binarized cross-approximate entropy in crowdsensing environment. *Comput. Biol. Med.* **2017**, *80*, 137–147. [[CrossRef](#)]
6. Wu, Y.; Shang, P.; Li, Y. Multiscale sample entropy and cross-sample entropy based on symbolic representation and similarity of stock markets. *Commun. Nonlinear Sci. Numer. Simul.* **2018**, *56*, 49–61. [[CrossRef](#)]
7. He, J.; Shang, P.; Xiong, H. Multidimensional scaling analysis of financial time series based on modified cross-sample entropy methods. *Phys. A Stat. Mech. Appl.* **2018**, *500*, 210–221. [[CrossRef](#)]
8. Porta, A.; Baselli, G.; Lombardi, F.; Montano, N.; Malliani, A.; Cerutti, S. Conditional entropy approach for the evaluation of the coupling strength. *Biol. Cybern.* **1999**, *81*, 119–129. [[CrossRef](#)]
9. Wang, Y.; Shang, P. Analysis of financial stock markets through the multiscale cross-distribution entropy based on the Tsallis entropy. *Nonlinear Dyn.* **2018**, *94*, 1361–1376. [[CrossRef](#)]
10. He, J.; Shang, P.; Zhang, Y. PID: A PDF-induced distance based on permutation cross-distribution entropy. *Nonlinear Dyn.* **2019**, *97*, 1329–1342. [[CrossRef](#)]
11. Škorić, T. Automatic Determination of Cross-Approximate Entropy Parameters for Cardiovascular Time Series. In Proceedings of the 2018 26th Telecommunications Forum (TELFOR), Belgrade, Serbia, 20–21 November 2018; pp. 1–8. [[CrossRef](#)]
12. Wang, R.; Li, D.; Wang, J.; Cai, L.; Shi, L. Synchrony analysis using different cross-entropy measures of the electroencephalograph activity in Alzheimer’s disease. In Proceedings of the 2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Datong, China, 15–17 October 2016; pp. 1541–1545. [[CrossRef](#)]
13. Zhu, K.; Song, X. Cross-fuzzy entropy-based approach for performance degradation assessment of rolling element bearings. *Proc. Inst. Mech. Eng. E J. Process Mech. Eng.* **2018**, *232*, 173–185. [[CrossRef](#)]
14. Costa, M.; Goldberger, A.L.; Peng, C.K. Multiscale Entropy Analysis of Complex Physiologic Time Series. *Phys. Rev. Lett.* **2002**, *89*, 068102. [[CrossRef](#)] [[PubMed](#)]
15. Yan, R.; Yang, Z.; Zhang, T. Multiscale Cross Entropy: A Novel Algorithm for Analyzing Two Time Series. In Proceedings of the 2009 Fifth International Conference on Natural Computation, Tianjin, China, 14–16 August 2009; Volume 1, pp. 411–413, ISSN 2157-9555, 2157-9563. [[CrossRef](#)]
16. Yin, Y.; Shang, P.; Feng, G. Modified multiscale cross-sample entropy for complex time series. *Appl. Math. Comput.* **2016**, *289*, 98–110. [[CrossRef](#)]
17. Jamin, A.; Duval, G.; Annweiler, C.; Abraham, P.; Humeau-Heurtier, A. A Novel Multiscale Cross-Entropy Method Applied to Navigation Data Acquired with a Bike Simulator. In Proceedings of the 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Berlin, Germany, 23–27 July 2019; pp. 733–736, ISSN 1558-4615, 1557-170X. [[CrossRef](#)]
18. Wei, H.C.; Xiao, M.X.; Ta, N.; Wu, H.T.; Sun, C.K. Assessment of Diabetic Autonomic Nervous Dysfunction with a Novel Percussion Entropy Approach. *Complexity* **2019**, *2019*, 11. [[CrossRef](#)]

19. Xiao, M.X.; Lu, C.H.; Ta, N.; Jiang, W.W.; Tang, X.J.; Wu, H.T. Application of a Speedy Modified Entropy Method in Assessing the Complexity of Baroreflex Sensitivity for Age-Controlled Healthy and Diabetic Subjects. *Entropy* **2019**, *21*. [[CrossRef](#)]
20. Castiglioni, P.; Parati, G.; Faini, A. Information-Domain Analysis of Cardiovascular Complexity: Night and Day Modulations of Entropy and the Effects of Hypertension. *Entropy* **2019**, *21*, 550. [[CrossRef](#)]
21. Xiao, M.X.; Wei, H.C.; Xu, Y.J.; Wu, H.T.; Sun, C.K. Combination of R-R Interval and Crest Time in Assessing Complexity Using Multiscale Cross-Approximate Entropy in Normal and Diabetic Subjects. *Entropy* **2018**, *20*. [[CrossRef](#)]
22. Lin, T.K.; Chien, Y.H. Performance Evaluation of an Entropy-Based Structural Health Monitoring System Utilizing Composite Multiscale Cross-Sample Entropy. *Entropy* **2019**, *21*. [[CrossRef](#)]
23. Wang, F.; Zhao, W.; Jiang, S. Detecting asynchrony of two series using multiscale cross-trend sample entropy. *Nonlinear Dyn.* **2019**. [[CrossRef](#)]
24. Yin, Y.; Shang, P.; Ahn, A.C.; Peng, C.K. Multiscale joint permutation entropy for complex time series. *Phys. A Stat. Mech. Appl.* **2019**, *515*, 388–402. [[CrossRef](#)]
25. Silva, L.E.V.; Murta, L.O., Jr. Evaluation of physiologic complexity in time series using generalized sample entropy and surrogate data analysis. *Chaos* **2012**, *22*, 043105. [[CrossRef](#)] [[PubMed](#)]
26. Bandt, C.; Pompe, B. Permutation Entropy: A Natural Complexity Measure for Time Series. *Phys. Rev. Lett.* **2002**, *88*, 174102. [[CrossRef](#)] [[PubMed](#)]
27. Porta, A.; Baselli, G.; Liberati, D.; Montano, N.; Cogliati, C.; Gnecci-Ruscone, T.; Malliani, A.; Cerutti, S. Measuring regularity by means of a corrected conditional entropy in sympathetic outflow. *Biol. Cybern.* **1998**, *78*, 71–78. [[CrossRef](#)] [[PubMed](#)]
28. Chen, W.; Wang, Z.; Xie, H.; Yu, W. Characterization of Surface EMG Signal Based on Fuzzy Entropy. *IEEE Trans. Neural Syst. Reh. Eng.* **2007**, *15*, 266–272. [[CrossRef](#)] [[PubMed](#)]
29. Yin, Y.; Peng, C.K.; Hou, F.; Gao, H.; Shang, P.; Li, Q.; Ma, Y. The application of multiscale joint permutation entropy on multichannel sleep electroencephalography. *AIP Adv.* **2019**, *9*, 125214. [[CrossRef](#)]
30. Zhang, Y.C. Complexity and $1/f$ noise. A phase space approach. *J. Phys. I France* **1991**, *1*, 971–977. [[CrossRef](#)]
31. Wu, H.T.; Lee, C.Y.; Liu, C.C.; Liu, A.B. Multiscale Cross-Approximate Entropy Analysis as a Measurement of Complexity between ECG R-R Interval and PPG Pulse Amplitude Series among the Normal and Diabetic Subjects. *Comput. Math. Methods Med.* **2013**, *2013*, 7. [[CrossRef](#)]
32. Humeau-Heurtier, A. The multiscale entropy algorithm and its variants: a review. *Entropy* **2015**, *17*. [[CrossRef](#)]
33. Yin, Y.; Shang, P. Asymmetric asynchrony of financial time series based on asymmetric multiscale cross-sample entropy. *Chaos* **2015**, *25*, 032101. [[CrossRef](#)]
34. Costa, M.; Goldberger, A.L.; Peng, C.K. Multiscale entropy to distinguish physiologic and synthetic RR time series. In Proceedings of the Computers in Cardiology, Memphis, TN, USA, 22–25 September 2002; pp. 137–140. [[CrossRef](#)]
35. Wu, S.D.; Wu, C.W.; Lin, S.G.; Wang, C.C.; Lee, K.Y. Time Series Analysis Using Composite Multiscale Entropy. *Entropy* **2013**, *15*, 1069–1084. [[CrossRef](#)]
36. Higuchi, T. Approach to an irregular time series on the basis of the fractal theory. *Phys. D Nonlinear Phenom.* **1988**, *31*, 277–283. [[CrossRef](#)]
37. Pham, T.D. Time-Shift Multiscale Entropy Analysis of Physiological Signals. *Entropy* **2017**, *19*, 257. [[CrossRef](#)]
38. Dong, Z.; Zheng, J.; Huang, S.; Pan, H.; Liu, Q. Time-Shift Multi-scale Weighted Permutation Entropy and GWO-SVM Based Fault Diagnosis Approach for Rolling Bearing. *Entropy* **2019**, *21*. [[CrossRef](#)]
39. Zhu, X.; Zheng, J.; Pan, H.; Bao, J.; Zhang, Y. Time-Shift Multiscale Fuzzy Entropy and Laplacian Support Vector Machine Based Rolling Bearing Fault Diagnosis. *Entropy* **2018**, *20*. [[CrossRef](#)]
40. Kesić, S.; Spasić, S.Z. Application of Higuchi's fractal dimension from basic to clinical neurophysiology: A review. *Comput. Methods Prog. Biomed.* **2016**, *133*, 55–70. [[CrossRef](#)]
41. Hu, M.; Liang, H. Uncovering perceptual awareness of visual stimulus with adaptive multiscale entropy. In Proceedings of the 2011 3rd International Conference on Awareness Science and Technology (iCAST), Dalian, China, 27–30 September 2011; pp. 485–490, ISSN 2325-5986, 2325-5994. [[CrossRef](#)]

42. Rehman, N.; Mandic, D.P. Multivariate empirical mode decomposition. *Proc. Royal Soc. A Math. Phys. Eng. Sci.* **2010**, *466*, 1291–1302. [[CrossRef](#)]
43. Wu, H.T.; Lo, M.T.; Chen, G.H.; Sun, C.K.; Chen, J.J. Novel Application of a Multiscale Entropy Index as a Sensitive Tool for Detecting Subtle Vascular Abnormalities in the Aged and Diabetic. *Comput. Math. Methods Med.* **2013**, *2013*, 8. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).