

Article

# Slope Entropy: A New Time Series Complexity Estimator Based on Both Symbolic Patterns and Amplitude Information

David Cuesta-Frau 

Technological Institute of Informatics, Universitat Politècnica de València, Alcoi Campus, 03801 Alcoi, Spain; dcuesta@disca.upv.es; Tel.: +34-966528505

Received: 6 November 2019; Accepted: 26 November 2019; Published: 28 November 2019



**Abstract:** The development of new measures and algorithms to quantify the entropy or related concepts of a data series is a continuous effort that has brought many innovations in this regard in recent years. The ultimate goal is usually to find new methods with a higher discriminating power, more efficient, more robust to noise and artifacts, less dependent on parameters or configurations, or any other possibly desirable feature. Among all these methods, Permutation Entropy (PE) is a complexity estimator for a time series that stands out due to its many strengths, with very few weaknesses. One of these weaknesses is the PE's disregarding of time series amplitude information. Some PE algorithm modifications have been proposed in order to introduce such information into the calculations. We propose in this paper a new method, Slope Entropy (SlopEn), that also addresses this flaw but in a different way, keeping the symbolic representation of subsequences using a novel encoding method based on the slope generated by two consecutive data samples. By means of a thorough and extensive set of comparative experiments with PE and Sample Entropy (SampEn), we demonstrate that SlopEn is a very promising method with clearly a better time series classification performance than those previous methods.

**Keywords:** permutation entropy; sample entropy; signal classification; symbolic dynamics; discriminating power

## 1. Introduction

The capability of entropy or complexity measures to distinguish among time series classes and to understand the underlying dynamics is very well known [1–3]. Many different formulas, statistics, algorithms, or methods have been proposed since the introduction of the first method that arguably became widespread and generally used across a varied and diverse set of scientific and technological frameworks, Approximate Entropy, ApEn [4].

Most of these methods are based on counting events found or derived from the input time series under entropy analysis, in order to estimate probabilities from relative frequencies of such events. These probabilities are finally mapped to a single value that supposedly accounts somehow for the dynamic behaviour of the time series. These mapping frequently takes place using entropy definitions, such as Shannon [5], Renyi [6], Tsallis [7], or Kolmogorov–Sinai [8] entropies, among others not so often used.

The development of new entropy quantification methods is an ongoing and fruitful process. Since the introduction of ApEn, other derived methods have been proposed, such as Sample Entropy (SampEn) [9], Fuzzy Entropy (FuzzyEn) [10], Quadratic Sample Entropy (QSE) [11], and many more based on counting pattern matches in terms of time series subsequences amplitude differences. In addition, characterization studies related to these measures have also been published in order

to avoid blind application of such methods or to maximise the performance achieved [12–16]. The concomitant analysis of time series at different temporal scales is also a frequent strategy in practically any method to gain a better insight into temporal dynamics [17,18].

Another successful line of research is also based on counting pattern matches but using a symbolic representation of a time series subsequence instead of its original sample amplitude form. A good representative of this approach is the Lempel–Ziv Complexity (LZC) [19], but due to its simplicity and robustness, Permutation Entropy (PE) [20] is probably becoming the most used entropy measure in this group, well above LZC. There are also additional studies devoted to the characterization of these symbolic methods [21–26] and to implement multiscale temporal analyses [27,28].

As stated above, PE is a very successful entropy statistic, but despite its strengths, it still has a few, however important, weaknesses. Since PE is based on relative frequencies of ordinal patterns resulting from sorting subsequences, when equal values are found in such subsequences, there is an ambiguity in the sorting process that has to be addressed consistently. These equal values or ties can lead to a misinterpretation of the time series nature [26], although they seem to play a minor role where classification tasks are concerned [24]. Regardless, several methods have been proposed to address this potentially detrimental weakness [29,30].

It has also been frequently claimed that not including amplitude information could have an adverse impact on PE performance, too. Thus, PE variations that consider both ordinal and amplitude information have been proposed, such as Weighted-PE [31], Amplitude-Aware-PE [29], or Fine Grained-PE [32]. In this case, the inclusion of amplitude information does seem to improve the discriminating power of PE in classification tasks [33].

Due to this significance, we tried to devise another method that also combined a symbolic representation of patterns and amplitude information. We based our idea on methods employed for syntactic pattern recognition and polygonal approximation of data, successfully used in the past to classify electrocardiogram (ECG) records [34,35]. The basic idea was to encode the magnitude of the amplitude differences between consecutive samples in the time series by an alphabet in which symbols accounted for a range of differences, spanning from 0 to  $\infty$ . The method should also be simple, efficient in terms of memory requirements and computational cost, and without a strong dependence on thresholds and parameters.

Based on the scaffolding provided by the standard PE algorithm, we propose in this paper a new entropy statistic termed Slope Entropy (SlopEn) that satisfies the requirements stated above. The method uses an alphabet of three symbols, 0, 1, and 2, with positive (+) and negative versions (–) of the last two. Each symbol covers a range of slopes for the segment joining two consecutive samples of the input data, and the relative frequency of each pattern found is mapped into a real value using a Shannon entropy approach [5].

In order to validate the approach proposed, a comprehensive experimental comparative study was conducted. The comparison took place using PE (ordinal patterns) and SampEn (amplitude patterns), two of the most representative entropy measures used in the scientific literature, and an experimental dataset based on publicly available records to ensure reproducibility. The results confirmed SlopEn as an entropy measure of great potential that outperformed both PE and SampEn under a great disparity of conditions and experimental settings.

## 2. Materials and Methods

This study uses two well known entropy methods, PE and SampEn, as the references against which the new SlopEn method can be validated. The input time series is referred to as the vector  $\mathbf{x} = \{x_0, x_1, \dots, x_{N-1}\}$ , where  $x_i$  is the  $i$ -th amplitude sample, and the number of samples is  $N$ . The embedded dimension for all measures is referred to as  $m$ .

The experimental dataset contains synthetic and real records. This dataset has been chosen with two purposes in mind: facilitate the reproducibility of the results by using publicly available data and analyse records difficult to classify by the standard PE method. It also contains not only biomedical

data but electricity consumption data, as well as other records successfully classified by PE, in order to offer a complete and unbiased picture of the SlopEn capabilities and improvements. The methods stated above and the dataset are described in the next sections.

### 2.1. Sample Entropy

SampEn [36] is based on computing the relative frequency of similar amplitude subsequences. A subsequence starting at sample  $j$ , of length  $m$ , defined as  $\mathbf{x}_j^m = \{x_j, x_{j+1}, \dots, x_{j+m-1}\}$ , is compared with all the other possible subsequences of length  $m$  starting at sample  $i$ , extracted from  $\mathbf{x}$ , except with itself, that is,  $0 \leq i, j < N - m + 1, i \neq j$ .

The distance between  $\mathbf{x}_j^m$  and  $\mathbf{x}_i^m$  is given by  $d_{ji} = \max(|x_{j+k} - x_{i+k}|), 0 \leq k \leq m - 1$ . In order to consider two subsequences similar, this distance should be below a predefined threshold, usually termed  $r$ . In this work,  $r$  was set to 0.25 in all the experiments.

The number of subsequences similar to  $\mathbf{x}_j^m$  is stored in a specific counter,  $B_j^m(r)$ . When all the possible subsequences in  $\mathbf{x}$  have been processed, a final statistic for the time series is computed as:

$$B^m(r) = \frac{1}{N - m} \sum_{j=0}^{N-m-1} B_j^m(r).$$

The length of the subsequences is then increased by 1, and the previous similarity calculations are repeated for this new length. In this case, the final statistic is termed  $B^{m+1}(r)$ . SampEn can then be computed as:

$$\text{SampEn}(\mathbf{x}, m, r, N) = -\log \left[ \frac{B^{m+1}(r)}{B^m(r)} \right]. \quad (1)$$

### 2.2. Permutation Entropy

PE [20] is based on computing the relative frequency of ordinal patterns associated to time series subsequences. As for SampEn, all possible subsequences  $\mathbf{x}_j^m$  are sequentially drawn from  $\mathbf{x}$ . Then, the samples in  $\mathbf{x}_j^m$  are sorted in ascending order. The original indices of these samples conform another vector featuring the final location of each sample once they were sorted. This vector is usually defined as  $\boldsymbol{\pi}_j^m = \{\pi_0, \pi_1, \dots, \pi_{m-1}\}$  such that  $x_{j+\pi_0} \leq x_{j+\pi_1} \leq x_{j+\pi_2} \dots \leq x_{j+\pi_{m-1}}$ . The number of different ordinal patterns that can emerge from an alphabet of  $m$  symbols, is  $m!$ . Thus, comparing the ordinal pattern found,  $\boldsymbol{\pi}_j^m$ , with a list of all the possible  $m!$  ordinal patterns, it is possible to compute the relative frequency of each one. Thus, if a certain ordinal pattern has been found  $c_k$  times, its relative frequency can be obtained as  $p_k = \frac{c_k}{N-m+1}$ . PE can then be computed as the Shannon entropy of the estimated probabilities:

$$\text{PE} = - \sum_{k=0}^{m!-1} p_k \log p_k, \forall p_k > 0. \quad (2)$$

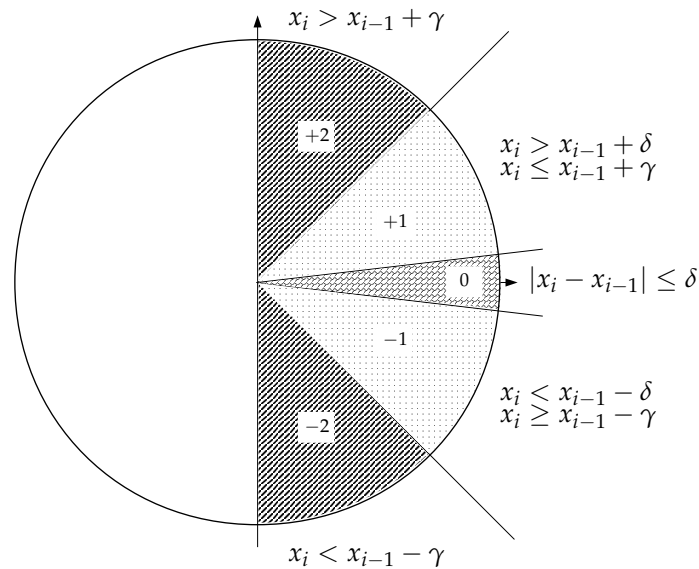
### 2.3. Slope Entropy

The purpose of SlopEn is to somehow include amplitude information in an otherwise symbolic representation of the input time series. Similar approaches have used a linear quantization scheme, with as many thresholds as levels desired, being Lempel–Ziv Complexity (LZC) [19] a good and generic representative of this approach, based usually on a single threshold and two symbols, 1 and 0. However, these methods are usually very dependent on the specific threshold chosen, and on the amplitude range of the time series under analysis.

Symbolic dynamics is a field of research that has already been explored in the context of signal classification. In addition to pattern recognition in ECG records [34,35], it has also been applied to RR records using a threshold to assign symbols to interbeat intervals [37]. Applied also to RR records, the method described in [38] represented heart rate accelerations by the symbol 1, and decelerations by

0. Even the number of forbidden words have been included as a classification feature [39]. However, these methods are too record specific, and their discriminating power was not very high [39].

We wanted to apply a similar scheme that had to be simple and less dependent on thresholds and specific amplitude values or records. In this regard, thresholds were based on the gradient between consecutive samples, instead of absolute values, and second, symbols should be assigned according to a certain range of differences or slopes, as depicted in Figure 1.



**Figure 1.** Graphical interpretation of the Slope Entropy (SlopEn) approach using three levels. Possible symbols are +2, +1, 0, -1, and -2, depending on the amplitude difference between consecutive samples.

Thus, each subsequence of length  $m$  drawn from  $x$ , can be transformed into another subsequence of length  $m - 1$  with the differences of each pair of consecutive samples,  $x_i - x_{i-1}$ . Then, a threshold or thresholds must be applied to these differences in order to find the corresponding symbolic representation. Once these symbols are obtained, a Shannon entropy approach can be applied in a similar manner as for PE, but instead of normalising by a constant value (the possible number of ordinal patterns in PE,  $m!$ ), the factor employed corresponds to the actual number of slope patterns found. This way, the possible additional information provided by forbidden patterns [40,41], can be also exploited, although the values obtained can not be considered a true probability value.

The specific SlopEn configuration proposed in the present study is very straightforward. It considers the horizontal increment between consecutive samples to always be 1, and the differences (vertical increment) are thresholded by a parameter  $\gamma$ , taken as 1 in the present study ( $45^\circ$  angle). The vicinity of the 0-difference region is managed by another threshold, termed  $\delta$ , whose chosen value in this case was  $1 \times 10^{-3}$ , to account for possible ties [26]. Being  $x_i$  and  $x_{i-1}$  two consecutive values of the input time series, the symbols can be assigned according to the following rules:

- If  $x_i > x_{i-1} + \gamma$ , the symbol is +2.
- If  $x_i > x_{i-1} + \delta$  and  $x_i \leq x_{i-1} + \gamma$ , below the  $45^\circ$  angle and above the 0 region when  $\gamma = 1$ , the symbol is +1.
- In the vicinity of the 0 difference, when  $|x_i - x_{i-1}| \leq \delta$ , the symbol assigned is 0.
- $x_i < x_{i-1} - \delta$  and  $x_i \geq x_{i-1} - \gamma$ , above the  $-45^\circ$  angle, and below the 0 region when  $\gamma = 1$ , the symbol is -1.
- If  $x_i < x_{i-1} - \gamma$ , the symbol is -2.

In any method, a resampling of the input sequence can be applied to study the possible information distribution across other temporal scales. The parameter that accounts for the magnitude

of this new scale, the embedded delay, is usually represented by  $\tau$ , with  $\tau \geq 1$ . In this work,  $\tau = 1$  for all the experiments, since it is still the most frequent case [29], except in Section 3.5.

For example, a subsequence  $\{4.4, 3.6, 5.3\}$  would result in a vector of differences  $\{-0.8, 1.7\}$ , whose SlopEn symbolic representation is  $\psi_{j=0}^3 = \{-1, +2\}$ , with vector components  $\psi_0 = -1$  and  $\psi_1 = +2$ . There is a complete and detailed SlopEn computation example in Appendix A, with source code in Appendix B. All the SlopEn computation steps are listed in Algorithm 1.

---

### Algorithm 1 Slope Entropy (SlopEn) Algorithm

---

**Input:** Time series  $\mathbf{x}$ , embedded dimension  $m > 2$ , length  $N > m + 1$ ,  $\delta, \gamma > \delta$

**Initialisation:** SlopEn  $\leftarrow 0$ , slope pattern counter vector  $\mathbf{c} \leftarrow \{\emptyset\}$ , slope patterns relative frequency vector  $\mathbf{p} \leftarrow \{\emptyset\}$ , list of slope patterns found  $\Psi^m \leftarrow \{\emptyset\}$

**for**  $j \leftarrow 0, \dots, N - m$  **do** ▷ For all the samples in the input time series

**for**  $i \leftarrow j + 1, \dots, j + m - 1$  **do** ▷ Examine samples in a subsequence of length  $m$

**if**  $(x_i - x_{i-1}) \in [-\delta, \delta]$  **then**

$\psi_{i-(j+1)} \leftarrow 0$  ▷ Add symbol 0 to pattern vector  $\psi_j^m$

**end if**

**if**  $(x_i - x_{i-1}) \in ]\delta, \gamma]$  **then**

$\psi_{i-(j+1)} \leftarrow +1$  ▷ Add symbol +1 to pattern vector  $\psi_j^m$

**end if**

**if**  $(x_i - x_{i-1}) \in ]\gamma, \infty[$  **then**

$\psi_{i-(j+1)} \leftarrow +2$  ▷ Add symbol +2 to pattern vector  $\psi_j^m$

**end if**

**if**  $(x_i - x_{i-1}) \in [-\gamma, -\delta[$  **then**

$\psi_{i-(j+1)} \leftarrow -1$  ▷ Add symbol -1 to pattern vector  $\psi_j^m$

**end if**

**if**  $(x_i - x_{i-1}) \in ]-\infty, \gamma[$  **then**

$\psi_{i-(j+1)} \leftarrow -2$  ▷ Add symbol -2 to pattern vector  $\psi_j^m$

**end if**

**end for**

bFound  $\leftarrow$  false

**for**  $i \leftarrow 0, \dots, \text{sizeof}(\Psi^m) - 1$  **do** ▷ Search on list of patterns already found for current pattern  $\psi_j^m$

**if**  $\psi_j^m = \Psi_i^m$  **then** ▷ A match for  $\psi_j^m$  found in  $\Psi_i^m$

$c_i \leftarrow c_i + 1$  ▷ Update frequency counter for  $\psi_j^m$

bFound  $\leftarrow$  true

**break** ▷ Stop search

**end if**

**end for**

**if not** bFound **then** ▷ Slope pattern  $\psi_j^m$  not found in  $\Psi_i^m$

$\Psi^m \leftarrow \psi_j^m$  ▷ Append pattern  $\psi_j^m$  to list  $\Psi_i^m$

$\mathbf{c} \leftarrow 1$  ▷ Append and initialise pattern count

**end if**

**for**  $i \leftarrow 0, \dots, \text{sizeof}(\Psi^m) - 1$  **do**

$\mathbf{p} \leftarrow p_i \leftarrow \frac{c_i}{\text{sizeof}(\Psi^m)}$  ▷ Append probability normalised by number of actual patterns found

SlopEn  $\leftarrow$  SlopEn +  $(-p_i \log p_i)$  ▷ Compute Shannon entropy

**end for**

**Output:** SlopEn( $\mathbf{x}, m, N, \gamma, \delta$ ) ▷ Return result

---

Algorithm 1 can easily be optimised. For example, consecutive slope patterns overlap in the last samples, and it would not be necessary to compute the entire pattern for each subsequence. However, the algorithm is proposed in a generic and basic form, and optimisations are left for future studies.

#### 2.4. Experimental Dataset

The experimental dataset was composed of synthetic and real records. These datasets are described next:

- Random records. There is a clear synthetic case where PE failed to find differences between two classes: random time series with Gaussian or uniform amplitude distributions. This is a

representative example of what happens when classes under analysis have the same temporal correlations but differ in amplitude: PE discriminating power gets lost [42]. A dataset of this case was included in the experiments in order to find out if SloPEn was capable of overcoming this known weakness of PE. Two classes were generated using Gaussian or uniform amplitude distributions, with 100 records each, with a length of 5000 samples. An example of records from each class is shown in Figure 2. This dataset will be referred to in the paper as the RANDOM dataset.

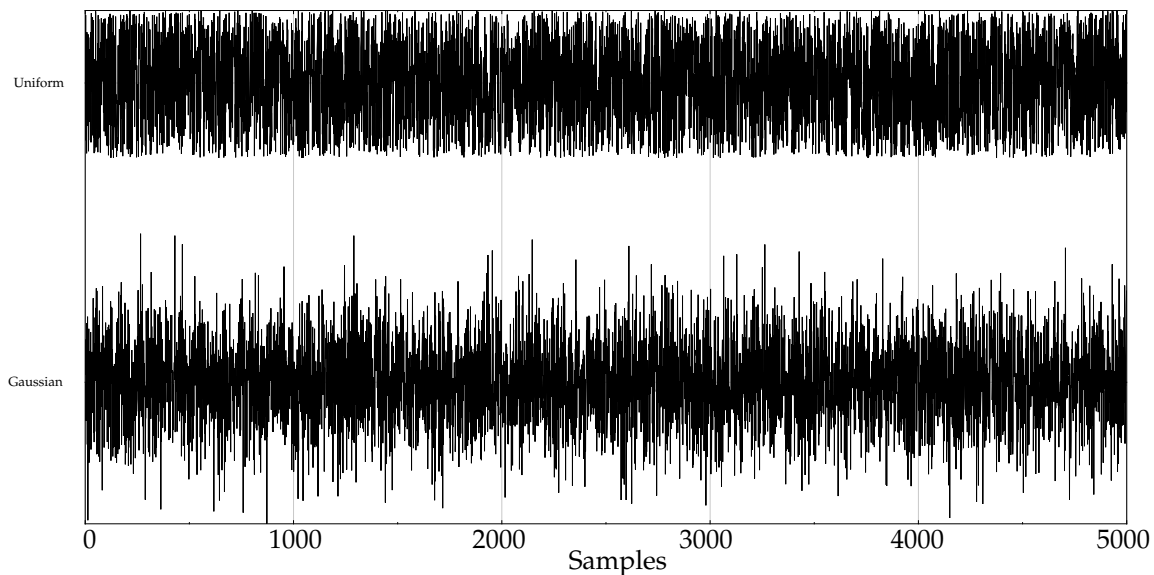


Figure 2. Example of generated synthetic records for the RANDOM database.

- Electroencephalographic records (EEGs) are the focus of many studies using entropy measures [43–45]. They have been used for a variety of purposes, such as to assess the mental status of a subject, driver’s fatigue, depth of anaesthesia, to detect a neurological disorder, or to predict the onset of epileptic seizures. There is also a great public availability of EEG records. For its good results using PE and SampEn in previous works, and due to the fact that it is probably the most widely known and analysed EEG database, we chose the University of Bonn EEG database [46]. There are five record classes in this database, but we only used the seizure-free and seizure-included records of classes D and E, respectively (100 records each one, uniform length of 4096 samples), easily separable, in principle. An example of class D record is plotted in Figure 3a, and in Figure 3b for class E.

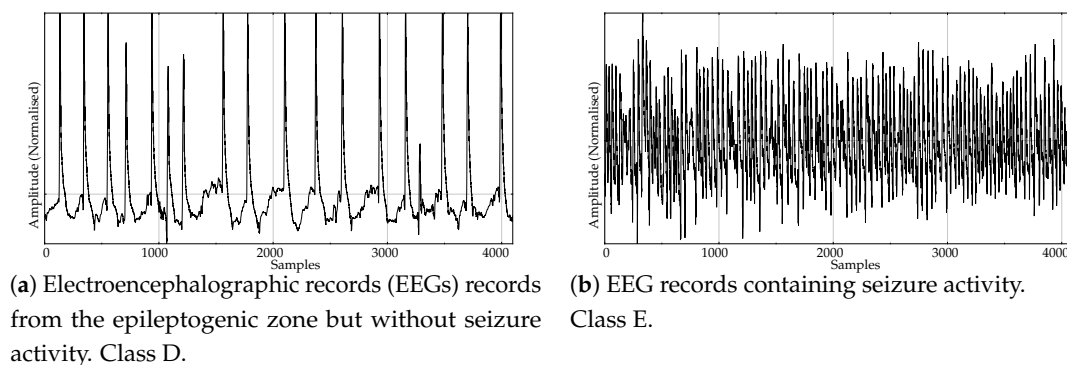
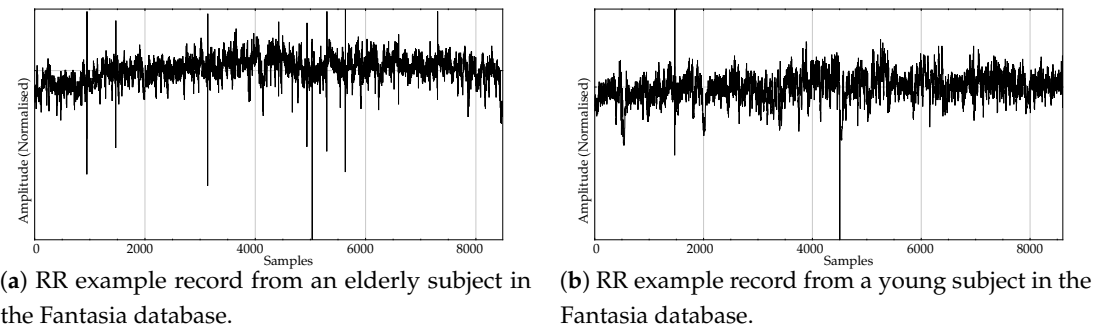


Figure 3. Examples of EEG records from the two classes used in the experiments.

- Another type of biomedical records extensively analysed using non-linear methods are series of time durations between consecutive R-waves in the electrocardiogram (ECG), or RR

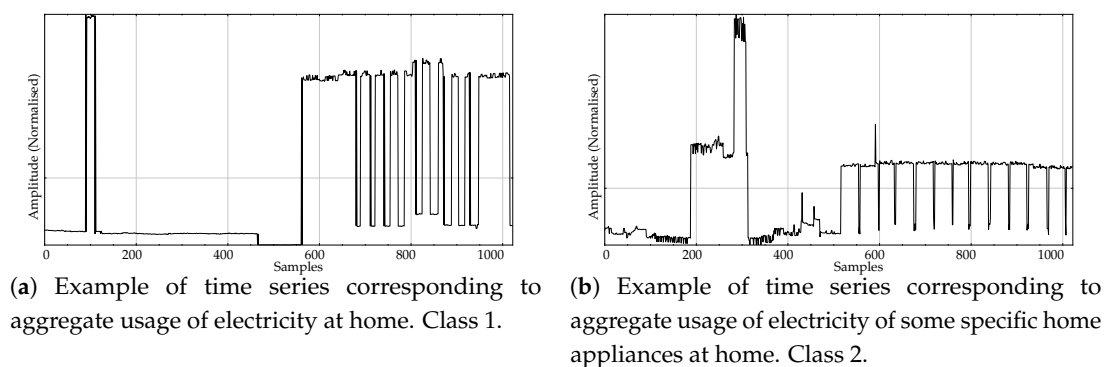


intervals [47–49]. We chose a publicly available RR database from the PhysioBank [50], the well known Fantasia database [51]. This database contains 20 young (21–34 years old) and 20 elderly (68–85 years old) healthy subjects data whose ECG signal was recorded during 120 min while in continuous supine resting. Examples of records from the elderly and young population are shown in Figure 4a,b, respectively.



**Figure 4.** Examples of signals from the two classes of the RR database.

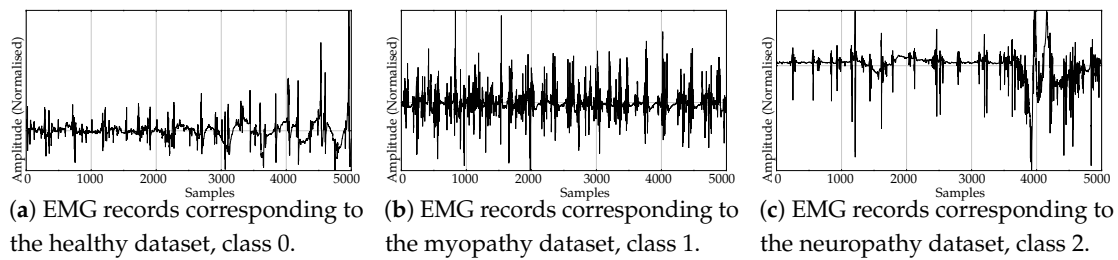
- Entropy measures are also very popular in other time series domains, beyond the very successful one of biomedical records. Along this line, we looked for other publicly available datasets featuring a complete different kind of time series, and we found the varied and diverse repository at [www.timeseriesclassification.com](http://www.timeseriesclassification.com) [52]. Within this repository, we chose two classes of data from the Personalised Retrofit Decision Support Tools for UK Homes Using Smart Home Technology (REFIT) project [53]. The first class contains data related to aggregate usage of electricity (Figure 5a), and the second one to aggregate usage of electricity of some specific home appliances (Figure 5b). This dataset contains 20 records from each class, with a uniform length of 1022 samples. We used this dataset in a previous study [33] where PE was unable to find significant differences between the two classes. Therefore, this should be considered a difficult dataset for entropy measures based only on ordinal patterns. We will refer to this dataset across the paper as the ENERGY dataset.



**Figure 5.** Example of time series from the two classes included in the ENERGY experimental dataset.

- The scientific and medical interest on Electromyograms (EMGs) and entropy measures is raising due to the recent availability of inexpensive continuous portable monitoring devices and the insight they provide into a number of important pathologies and motor disorders. They have been used to assess Parkinson’s disease [54], the neuromuscular impact of strokes [55], and muscular performance [56,57], to name just a few. The well-known site of Physionet [50] provides examples of EMGs, which we have used in previous classification studies, easily separable [22]. From three very long records of healthy, myopathy and neuropathy patients, we created three datasets by extracting non-overlapping epochs of 5000 samples. As a result, this dataset contains 10 healthy 5000 samples records (class 0), 22 myopathy 5000 samples records (class 1), and 29 neuropathy

5000 samples records (class 2). Examples of each class are shown in Figure 6a–c, respectively. This dataset will be referred to as the EMG dataset.



**Figure 6.** Examples of Electromyogram (EMG) records from the three classes used in the experiments.

All records were normalised (zero mean and unit variance) before computing SampEn and PE. They were not normalised for SlopEn in order to assess the possible influence of the amplitude, given that the thresholds were constant and the same for all records. For example, differences in the RR dataset ranged mainly between 0 and 100 ms, between 100 and 200  $\mu\text{V}$  for the EEG records, and between 0 and 20  $\mu\text{V}$  for the EMG time series, and SlopEn should be able to deal with these differences. A normalised amplitude is less challenging in terms of input parameter dependence or configuration, and that will be analysed in future studies along with more specific guidelines for parameter selection.

### 3. Experiments and Results

#### 3.1. Classification Accuracy Tests

This test was devised to find the classification accuracy achieved by each method using all datasets for  $m$  between 3 and 8. The average performance is shown in Table 1.

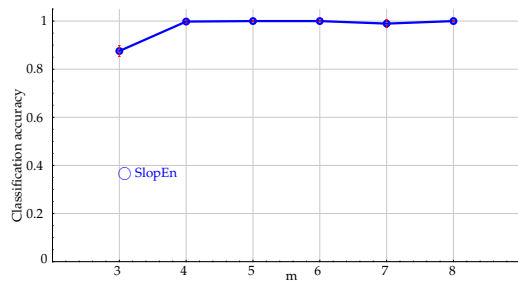
**Table 1.** Classification accuracy average results for all datasets. Statistically significant results are shown in bold. SampEn  $r = 0.25$ .

Dataset	Accuracy		
	PE	SampEn	SlopEn
RANDOM	0.62 $\pm$ 0.055	<b>0.62 <math>\pm</math> 0.063</b>	<b>0.97 <math>\pm</math> 0.047</b>
ENERGY	0.64 $\pm$ 0.013	<b>0.81 <math>\pm</math> 0.013</b>	<b>0.85 <math>\pm</math> 0.094</b>
EEG	<b>0.90 <math>\pm</math> 0.014</b>	<b>0.70 <math>\pm</math> 0.034</b>	<b>0.86 <math>\pm</math> 0.140</b>
RR	0.62 $\pm$ 0.031	<b>0.69 <math>\pm</math> 0.036</b>	<b>0.78 <math>\pm</math> 0.081</b>
EMG(0,1)	<b>1 <math>\pm</math> 0.000</b>	<b>0.74 <math>\pm</math> 0.063</b>	<b>0.91 <math>\pm</math> 0.164</b>
EMG(0,2)	<b>0.78 <math>\pm</math> 0.171</b>	<b>0.86 <math>\pm</math> 0.098</b>	<b>0.98 <math>\pm</math> 0.026</b>
EMG(1,2)	<b>1 <math>\pm</math> 0.000</b>	<b>0.89 <math>\pm</math> 0.048</b>	<b>0.94 <math>\pm</math> 0.112</b>

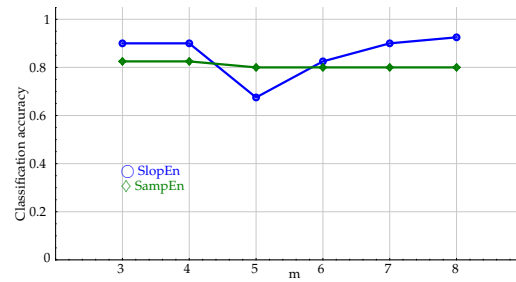
#### 3.2. Embedded Dimension Influence Tests

The influence of  $m$  in the performance of PE and SampEn is a well known issue [58]. Although some efforts have been devoted to minimise this influence [59], it still plays an important role, and its impact should be characterised, and compared. To this end, the classification experiments were repeated for  $m$  values ranging from 3 up to 8. These results are plotted in Figure 7.

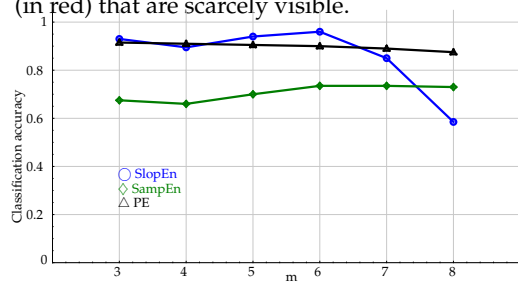




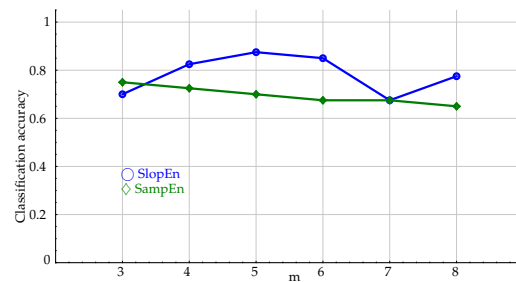
(a) Evolution of classification performance with  $m$  using the RANDOM dataset. Only Slope Entropy (SlopEn) results are shown since Sample Entropy (SampEn) and Permutation Entropy (PE) did not achieve statistically significant results in this case for any  $m$  value. Confidence intervals are so narrow (in red) that are scarcely visible.



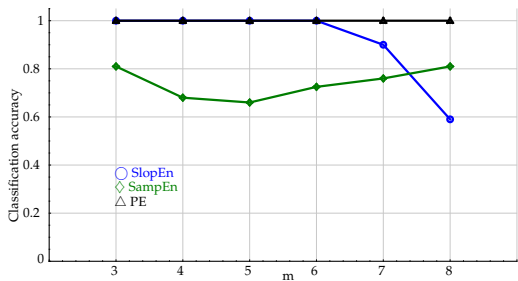
(b) Evolution of classification performance with  $m$  using the ENERGY dataset. Results using PE are not shown since it did not achieve statistically significant results in this case for any  $m$  value.



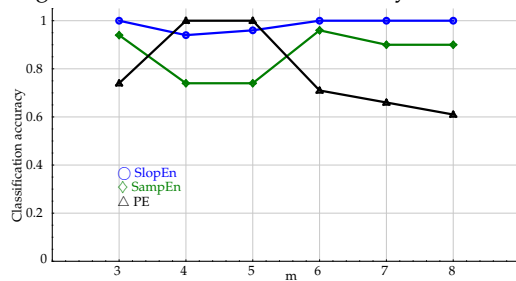
(c) Evolution of classification performance with  $m$  using the EEG dataset.



(d) Evolution of classification performance with  $m$  using the RR dataset. Results using PE are not shown since it did not achieve statistically significant results in this case for any  $m$  value.



(e) Evolution of classification performance with  $m$  using the EMG dataset. Classes under comparison 0 and 1.

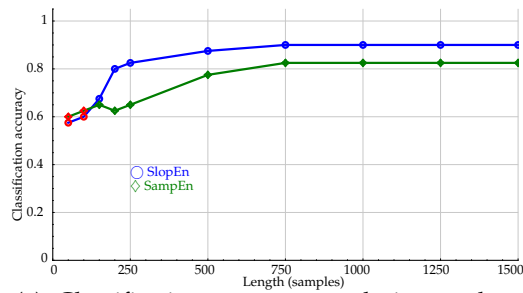


(f) Evolution of classification performance with  $m$  using the EMG dataset. Classes under comparison 0 and 2.

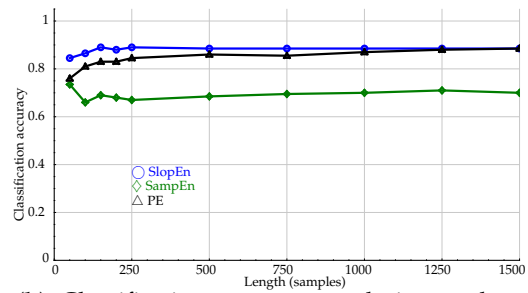
**Figure 7.** Evolution of classification performance with  $m$  for all the experimental datasets, and the three methods tested: PE, SampEn ( $r = 0.25$ ), and SlopEn.

### 3.3. Length Influence Tests

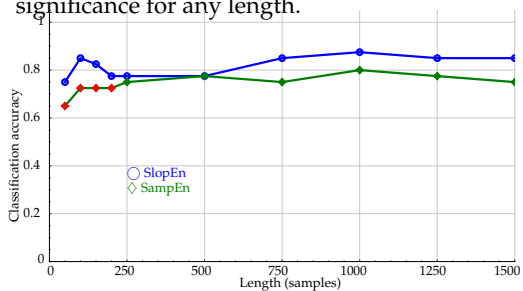
The length influence was assessed using the classification accuracy achieved at lengths  $n \in \{50, 100, 150, 200, 250, 500, 750, 1000, 1250, 1500\}$ . The first  $n$  samples of each record were used in these experiments instead of the entire records. The results are shown graphically in Figure 8.



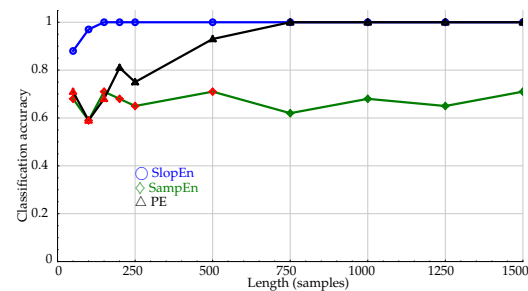
(a) Classification accuracy evolution as length increases for ENERGY records. Points in red correspond to lengths where significance was not achieved. PE is not included since it did not reach significance for any length.



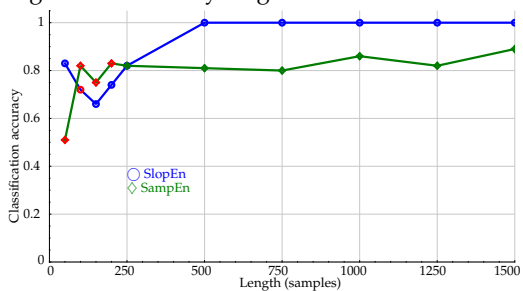
(b) Classification accuracy evolution as length increases for EEG records.



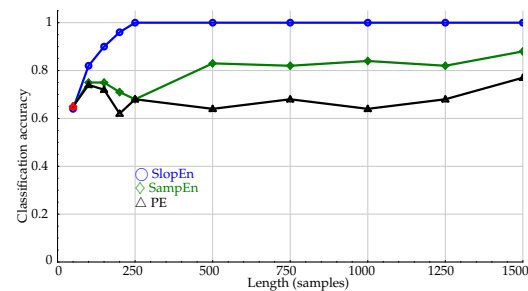
(c) Classification accuracy evolution as length increases for Fantasia RR records. Points in red correspond to lengths where significance was not achieved. PE is not included since it did not reach significance for any length.



(d) Classification accuracy evolution as length increases for EMG records. Points in red correspond to lengths where significance was not achieved. Classes under comparison 0 and 1.



(e) Classification accuracy evolution as length increases for EMG records. Points in red correspond to lengths where significance was not achieved. Classes under comparison 0 and 2. PE is not included since it did not reach significance for any length.



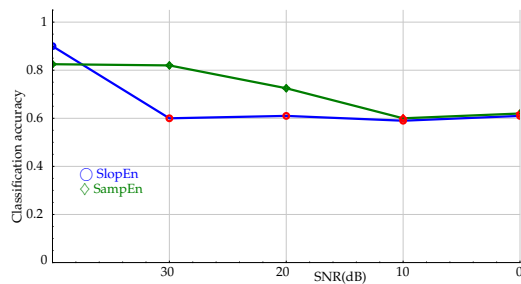
(f) Classification accuracy evolution as length increases for EMG records. Points in red correspond to lengths where significance was not achieved. Classes under comparison 1 and 2.

**Figure 8.** Classification accuracy evolution as length increases for the records in the dataset and the three entropy methods assessed, including the new method proposed, SlopEn ( $m = 3$ ). Red symbols represent statistically non-significant results. SampEn  $r = 0.25$ .

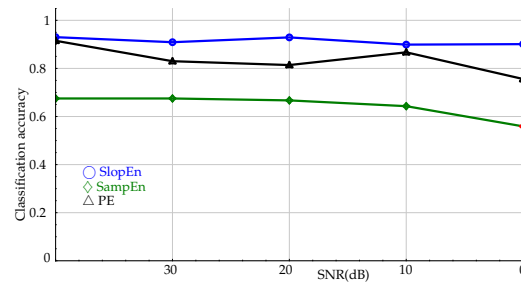
### 3.4. Noise Influence Tests

Robustness against noise has always been a very desirable property of non-linear measures. Some methods have failed to be widely used precisely because they were too sensitive to noise, no matter how high was their discriminating power when the records were clean. In order to avoid a similar fate for SlopEn, classification tests were repeated adding synthetic random uniform noise to the records in the experimental datasets, with signal-to-noise (SNR) ratios of 30dB, 20dB, 10dB, and 0dB, and  $m = 3$ . It was not possible to know the initial SNR, therefore the synthetic SNR was computed

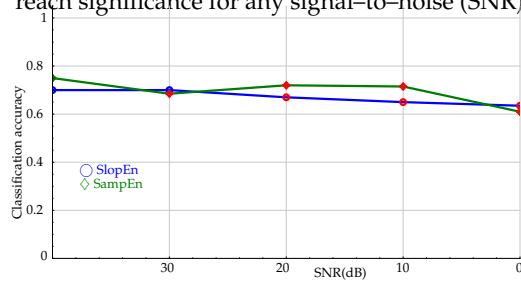
against the baseline records, in its original state in the database. The evolution of the classification accuracy for each case is plotted in Figure 9.



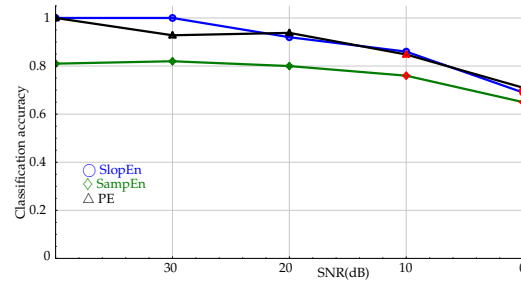
(a) Classification accuracy evolution as noise level increases for ENERGY records. Points in red correspond to noise levels where significance was not achieved. PE is not included since it did not reach significance for any signal-to-noise (SNR).



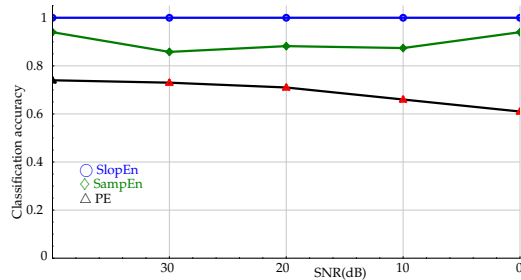
(b) Classification accuracy evolution as noise level increases for EEG records. Points in red correspond to noise levels where significance was not achieved.



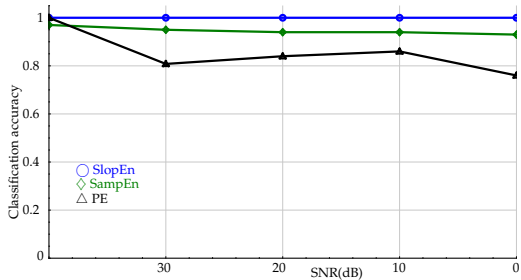
(c) Classification accuracy evolution as noise level increases for RR records. Points in red correspond to noise levels where significance was not achieved. PE is not included since it did not reach significance for any SNR.



(d) Classification accuracy evolution as noise level increases for EMG records. Points in red correspond to noise levels where significance was not achieved. Classes under comparison 0 and 1.



(e) Classification accuracy evolution as noise level increases for EMG records. Points in red correspond to noise levels where significance was not achieved. Classes under comparison 0 and 2.



(f) Classification accuracy evolution as noise level increases for EMG records. Points in red correspond to noise levels where significance was not achieved. Classes under comparison 1 and 2.

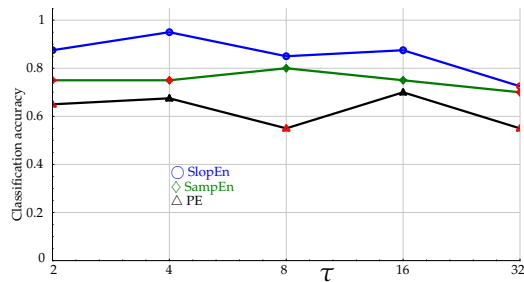
**Figure 9.** Classification accuracy evolution as noise level increases. The starting SNR, before the synthetic noise was added, was considered to be  $\infty$ . All the experiments used  $m = 3$ . Confidence intervals are not shown due to their small size, around 0.002–0.003.

### 3.5. Embedded Delay Influence Tests

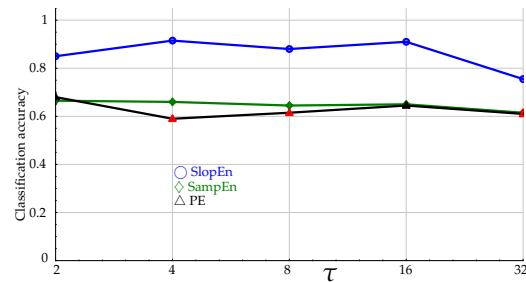
The embedded delay  $\tau$  in PE is frequently assumed to be 1 [29]. However, sometimes information about the time series dynamics is scattered across different time scales, and values  $\tau > 1$ ,  $\tau \in \mathbb{N}$ , have been proven to be very useful in those cases [60]. That is also the case for SampEn [61].

The experiments in this section were devised to assess the possible influence of  $\tau$  on SlopEn, as it does on PE and SampEn. In practical terms,  $\tau > 1$  corresponds to a downsampling process, where the output time series is obtained from the input one sampled at every  $\tau$  samples,  $x[nT] \rightarrow$

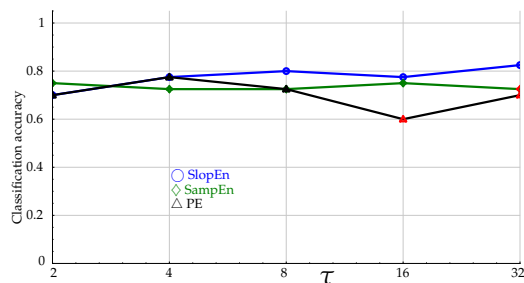
$x[n\tau T], n \in \mathbb{N}^+$ , but without applying a low-pass filter in order to avoid a possible aliasing [29]. The specific time scales employed in the experiments were  $\{2, 4, 8, 16, 32\}$ . For example, for  $\tau = 2$ , the initial data  $\{x_0, x_1, x_2, x_3, x_4, \dots\}$  will become  $\{x_0, x_2, x_4, \dots\}$ . The results are shown in Figure 10.



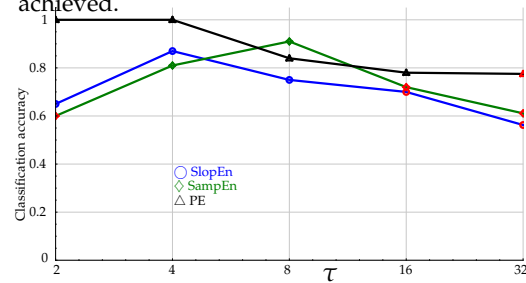
(a) Classification accuracy for different embedded delays  $\tau$  using ENERGY records. Points in red correspond to delays where significance was not achieved.



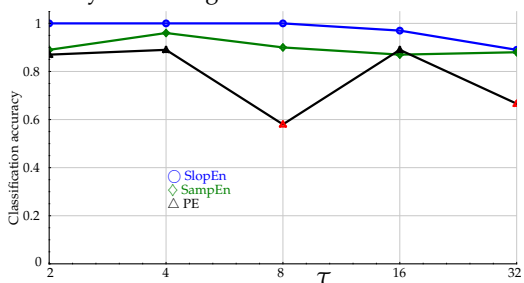
(b) Classification accuracy for different embedded delays  $\tau$  using EEG records. Points in red correspond to delays where significance was not achieved.



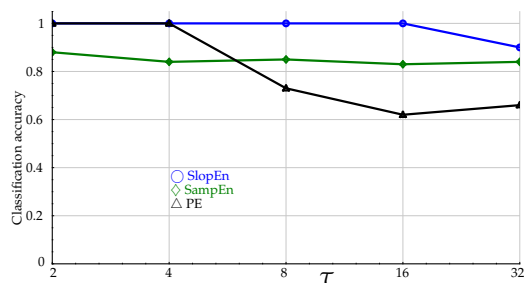
(c) Classification accuracy for different embedded delays  $\tau$  using RR records. Points in red correspond to delays where significance was not achieved.



(d) Classification accuracy for different embedded delays  $\tau$  using EMG records, classes 0 and 1. Points in red correspond to delays where significance was not achieved.



(e) Classification accuracy for different embedded delays  $\tau$  using EMG records, classes 0 and 2. Points in red correspond to delays where significance was not achieved.



(f) Classification accuracy for different embedded delays  $\tau$  using EMG records, classes 1 and 2.

**Figure 10.** Classification accuracy as a function of embedded delay  $\tau$ . All the experiments used  $m = 3$ .

### 3.6. SlopEn Parameters Influence Tests

The addition of more parameters to an entropy estimator method can be seen as a disadvantage, since, in principle, it increases the configuration effort for optimal performance. On the other hand, more parameters can also provide more flexibility for adaptation to the problem under analysis, provided the dependence on specific parameter values is low. This was the case, for example, for the evolution of SampEn to FuzzyEn [10]. The experiments in this section were devised to assess the robustness of SlopEn against changes in its specific parameters,  $\gamma$  and  $\delta$ . The results are shown in Table 2.

**Table 2.** SlopEn performance variation when parameters  $\gamma$  and  $\delta$  are modified ( $m = 3$ ). The first column of data corresponds to the baseline configuration, that used in all the previous experiments.

$\gamma$	1	3	6	5	5	1	2
$\delta$	$1 \times 10^{-3}$	$1 \times 10^{-3}$	$1 \times 10^{-3}$	1	$1 \times 10^{-3}$	$1 \times 10^{-9}$	$1 \times 10^{-9}$
RR	0.70	0.70	0.82	0.85	0.70	0.70	0.70
EEG	0.93	0.94	0.94	0.95	0.95	0.93	0.94
ENERGY	0.90	0.87	0.57	0.60	0.62	0.90	0.92
EMG	1,1,1	1,1,1	1,0.88,0.88	0.51,1,1	0.93,1,1	1,1,1	1,1,1

#### 4. Discussion

The classification accuracy achieved with SlopEn was the highest in all cases tested (Table 1). Since the datasets were chosen from previous works where PE exhibited some limitations due to its inability to include amplitude information [33,42], its results were the worst of the three metrics, as expected. PE only found significant differences for EEG and EMG records. On the contrary, SampEn and SlopEn classification accuracy was significant in all cases, with SampEn performing best for  $m = 3$ , as usually recommended for SampEn and ApEn [4,36]. The performance for SlopEn was perfect for the RANDOM and EMG datasets, and very high for the other three. As hypothesised, SlopEn seems to take advantage of symbolic and amplitude information simultaneously, since it improves the individual results of both PE and SampEn. Cases where significant classification was not achieved in any experiment configuration were omitted in the subsequent tests.

The variation of this classification with the embedded dimension  $m$  yields a disparity of performances (Figure 7). In principle, SlopEn seems to have a greater variability with  $m$  than PE and SampEn, for datasets ENERGY (Figure 7b), EEG (Figure 7c), RR (Figure 7d), and the first case of the EMG dataset (Figure 7e). However, SlopEn is very stable for the RANDOM (Figure 7a) and the EMG dataset, second configuration (Figure 7f). Anyway, this is a similar behaviour exhibited by amplitude-based PE derived methods [33], and the performance of PE and SampEn is usually well below that of SlopEn for most of the  $m$  values tested, being SlopEn the only one achieving statistical significance in all cases.

It is important to note that the results in this paper can be compared with other in previous studies, since some experimental datasets are the same. For example, in [33], the classification performance of PE, along with improved PE versions, such as Weighted-PE [31], Fine Grained-PE [32], and Amplitude Aware-PE [29], was assessed too for datasets ENERGY, RR, and EEG. The best classification accuracy for these datasets in [33] was 0.87, achieved using Fine Grained-PE, 0.87, achieved using Weighted-PE, and 0.91, using Fine-Grained PE. The other method analysed in that study, Amplitude Aware-PE, achieved an accuracy of 0.62, 0.75, and 0.85 for those datasets. Using SlopEn, the results obtained in the present paper were 0.92, 0.87, and 0.96, using the same measure in the three cases. Therefore, SlopEn was able to outperform Weighted-PE, Fine Grained-PE, and Amplitude Aware-PE methods, too without customising the  $\gamma$  and  $\delta$  parameters, or the number of thresholds.

The comparative results of the length analysis shown in Figure 8a–d indicate that SlopEn is reasonably robust against short datasets. The classification performance provided by PE has already been demonstrated to be robust in these terms [22], and SampEn has also been claimed to exhibit less dependence on length than other very successful methods like ApEn [58]. In this context of already robust methods, SlopEn was capable of outperforming them, with an even more stable behaviour, in addition to higher accuracy performances, discussed in other experiments. For Energy consumption records (Figure 8a), SlopEn stabilised at  $N = 500$ , approximately, with performances above 0.8 at  $N = 250$ . SampEn also needed some 500 samples, and PE was not able to achieve statistical significance for any length studied. The results in Figure 8b for EEG records show that the three methods become stable at  $N = 250$ , but SlopEn achieves the maximum accuracy exhibited at  $N = 1500$ , at that point already, whereas the other two still need more points. The stability for the classification of Fantasia RR records (Figure 8c) is quantitatively very similar among the three methods but not qualitatively since

the results for the shortest lengths were not significant for SampEn, requiring at least 250 samples, and PE was unable to find significant differences. The experiment using EMG records is probably where the differences were most prominent (Figure 8d). With only 150 samples, SlopEn achieved the maximum performance, whereas PE needed 750 samples, the same for SampEn to achieve statistical significance. The last two length analysis experiments showed an even superior performance of SlopEn (Figure 8e,f).

SlopEn is also reasonably robust against noise. Except for the energy records (Figure 9a), the results achieved by SlopEn kept significance at lower SNR than SampEn and PE. For the EEG records (Figure 9b), the SlopEn response was quite flat, like that of PE, but with higher classification performance. SampEn lost significance at 0dB. In the case of RR results in Figure 9c, SampEn and SlopEn trends were fairly similar but significance was, again, better kept by SlopEn. It is also important to note that the experiments used  $m = 3$ , at which SampEn achieved the maximum accuracy but not SlopEn (Figure 7d). With other  $m$  values, the SlopEn response was even better, but in order to maintain the homogeneity of the experiments,  $m$  was kept constant. EMG records exhibited the same behaviour, as can be seen in Figure 9d.

The embedded delay  $\tau$  analysis results point in the same direction of SlopEn being superior to SampEn and PE. In all the cases tested and reported in Figure 10, the discriminating power of SlopEn was above that of the other two methods, except in the case of EMG records, classes 0 and 1, plotted in Figure 10d. Unless the time series exhibits a specific behaviour at a certain temporal scale, which is mainly not the case in the experimental dataset used in this study, the discriminating power is expected to fall with greater  $\tau$  values since some signal information is lost when samples are removed. The response of SlopEn, along with that of SampEn, was reasonably flat in this regard, being PE response more oscillating. Moreover, there were several non-significant results for PE in all cases except Figure 10e,f, for SampEn except in Figure 10f. SlopEn only failed to find significant differences for  $\tau = 16, 32$  in Figure 10a,d. Although, as stated above, none of the records exhibited a clear temporal multiscale behaviour, it is important to note that methods that were unable to find differences for the baseline case,  $\tau = 1$ , were capable at other embedded delay values, such as PE with ENERGY (Figure 10a,  $\tau = 4, 16$ ) and RR records (Figure 10c,  $\tau = 2, 4, 8$ ).

Table 2 shows the classification accuracy achieved for  $m = 3$  but with changes in SlopEn parameters  $\gamma$  and  $\delta$ . Despite significant variations in these parameters, the performance was fairly stable, except for ENERGY records with great  $\gamma$  values. It seems that highest accuracy can be achieved provided  $\delta$  is close to 0, and  $\gamma$  is relatively low, in the vicinity of 1 or 2. The key issue for SlopEn is arguably to distinguish between steep and gentle slopes in a fuzzy way, the exact values do not matter much. For example, the classification accuracy for RR records was the same, 0.70, for  $\gamma = 1, 3, 5$ . For EEG records was almost constant (around 0.94) for all the different  $\gamma$  and  $\delta$  combinations tested, and results for EMG dataset were also fairly stable, except for  $\delta = 1$ . ENERGY was one of the most difficult to classify cases, and that is reflected by a higher sensitivity to parameter values. This specific dataset required  $\gamma$  values below 4. It is also important to note that SlopEn was applied to not normalised records, with a great disparity in amplitudes. For optimal performance, a grid search could be conducted [14], and after a normalisation process, it could also be found out which parameter values are optimal for each dataset. Thus, further studies will be required to fine tune the use of the parameters, the number of thresholds, and to define a more uniform scheme using signal normalisation.

In a few tests where classification accuracies were not significant, this significance seemed sometimes not to be clearly correlated, or follow a uniform relationship pattern with classification accuracy. For example, in Figure 8e, using SlopEn, results were significant for length 50, with an accuracy of 0.83, then became non-significant for length 100, accuracy 0.72, and then significant again for length 150 and onward, with an accuracy at that point of 0.66, below the previous non-significant one of 0.72. Although this might seem counter-intuitive, it is actually relatively frequent due to the following reasons:



1. Accuracy is a kind of average between sensitivity and specificity, and a higher accuracy does not ensure significance because it can be the result of an unbalanced average. In this case, with a length of 100 samples, sensitivity was 0.51, and specificity 0.80, the average 0.72 was not significant because despite its high value, it came from a very low sensitivity. The same average for another test was achieved with a sensitivity of 0.80, and specificity of 0.68, but in this case, it was statistically significant. For length 150, the sensitivity was 1 and the specificity 0.57, significant for an accuracy of 0.66 but borderline.
2. There are many methods for equal mean hypothesis testing, each one with its strengths and weaknesses [62]. We used the Bootstrap method, since no assumptions about the input data have to be made [63]. However, the size and distribution of the data may influence its results, mainly when significance is borderline. For example, in the previous 0.72 and 0.66 example, the test prioritised specificity over sensitivity due to the size differences of the input classes, 10 and 29, respectively.
3. Rejecting the equal hypothesis is not a demonstration that it is completely false, or the other way round. Again, this is specially true in borderline cases where a minor random change can completely reverse the results.
4. There are many factors than can influence the differences between time series. They are usually considered stationary, but in reality, they might exhibit some temporal changes. For example, border effects are quite common in biomedical records [14], and this impacts the results in a length influence analysis. Other well-known effects are the stochastic resonance [64,65], whereby more noise does not necessary imply less discriminating power, just the opposite. Regarding the temporal scale given by  $\tau$ , a regular trend should not be expected in all cases because the classification performance depends on the information content of the temporal scale analysed. These scales could be completely independent in terms of this information content.

## 5. Conclusions

We proposed in this paper a new entropy estimator termed Slope Entropy (SlopEn). It is based on the relative frequency of symbolic patterns, where each symbol is assigned according to the difference between consecutive samples of the input time series. The algorithm is very simple and easy to implement, with a lot of room for improvements and customisations in further studies.

Although SlopEn requires two new parameters,  $\gamma$  and  $\delta$ , the classification accuracy is very stable for a wide range of these parameters but best for  $\delta \rightarrow 0$  and  $\gamma \approx 1, 2$ . For normalised records,  $\gamma$  just needs to be rescaled in the range  $] \delta, 1 [$ , approximately. The goal is to somehow detect and account for abrupt differences between consecutive samples in the time series, combining the positional or ordinal information of these differences, and their magnitude: high ( $\pm 2$  symbol), low ( $\pm 1$  symbol), and ties (0 symbol).

A thorough and fair comparative study was conducted to assess the goodness of the new method proposed. Two of the most used entropy quantification methods were included in the experiments for comparative purposes: PE, as a good representative of ordinal-based approaches, and SampEn, based on amplitude differences. The experimental dataset included usual biomedical records in classification studies: EEG and RR records, records where PE achieved very good classification accuracy, EMG records, and synthetic and real records where PE has failed because amplitude information was a key distinguishing feature: Gaussian and uniform random noise [42], and energy consumption records [33]. This way, the study was also not constrained to just biomedical records.

In absolute terms, and using a default and stable parameter configuration for SlopEn, the classification accuracy achieved by this new measure was higher than that achieved by PE or SampEn. Even for the difficult cases where PE was unable to achieve statistically significant differences, SlopEn performance was between 87% and 100%. Not reported in this paper, but probably the focus of future studies, preliminary tests on other biomedical records, including RR, temperature, blood

pressure, or blood glucose records, have shown the same trend of superior performance by SlopEn without any fine tuning at all, just using the baseline configuration proposed in the present study.

The additional tests for parameter dependence and noise robustness have also demonstrated that SlopEn is a very promising method for a myriad of classification applications in the near future and in different contexts. Specifically, SlopEn seems to be more dependent on  $m$  than PE or SampEn but yields a higher accuracy in most of the cases. On the contrary, SlopEn is more robust against length  $N$  than the other two methods, with significant results with just 50 samples in almost all cases tested. The dependence on its specific parameters,  $\gamma$  and  $\delta$ , is low, and with a better customisation of these parameters, the performance of SlopEn in all tests would have been even higher. Anyway, normalisation should be considered a more robust approach, and in future works parameter customisation should be studied in the context of normalised records to ensure even better results. The embedded delay  $\tau$  had a negligible influence on SlopEn for the temporal scales and records employed. SlopEn was also robust against different levels of noise.

Further studies will be required in order to find out the possible relationship between normalised amplitudes and thresholds for optimal performance. Moreover, the number of thresholds could also be characterised for a specific time series. Other works should be devised to design classifiers based on SlopEn in order to confirm its superior accuracy for a particular application and data. Due to its simplicity and good performance, SlopEn could become a successful and widespread entropy quantifier method similar to SampEn or PE.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The author declares no conflict of interest.

## Appendix A. Example of SlopEn Computation

Let  $\mathbf{x}$  be an input time series series defined by  $\mathbf{x} = \{8.2, 8.1, 4.4, 3.6, 5.3, 5.4, 8.3, 1.9, 3.7, 8.6, 9.6, 9, 6, 8.7, 6.7, 3.3, 2, 2.5, 2.7, 4.6, 9.1, 1, 3.1, 1.7, 4.1, 3.8, 6.4, 1.3, 5.7, 3.4, 2.4, 2.1, 4.2\}$ . For  $m = 3$ ,  $N = 33$ , and the default configuration described in Section 2.3, the computation of SlopEn is as follows:

- Extract first subsequence from  $\mathbf{x}$ ,  $\mathbf{x}_{j=0}^3 = \{8.2, 8.1, 4.4\}$ . Compute the corresponding slope pattern,  $\psi_{j=0}^3 = \{-1, -2\}$ , since  $8.1 - 8.2 \in [-1, -0.001]$ , and  $4.4 - 8.1 < -1$ . Append pattern  $\{-1, -2\}$  to the list of patterns found  $\Psi^3$  and initialise its counter to 1:  $\Psi^3 = \left\{ \left\{ \{-1, -2\}, 1 \right\} \right\}$ .
- The next subsequence from  $\mathbf{x}$  is  $\mathbf{x}_{j=1}^3 = \{8.1, 4.4, 3.6\}$ . Pattern  $\psi_{j=1}^3 = \{-2, -1\}$ . This pattern is not in  $\Psi^3$ . Append it, and initialise its counter to 1:  $\Psi^3 = \left\{ \left\{ \{-1, -2\}, 1 \right\}, \left\{ \{-2, -1\}, 1 \right\} \right\}$ .
- Subsequence  $\mathbf{x}_{j=2}^3 = \{4.4, 3.6, 5.3\}$ . Pattern  $\psi_{j=2}^3 = \{-1, 2\}$ . This pattern is not in  $\Psi^3$ . Append it, and initialise its counter to 1:  $\Psi^3 = \left\{ \left\{ \{-1, -2\}, 1 \right\}, \left\{ \{-2, -1\}, 1 \right\}, \left\{ \{-1, 2\}, 1 \right\} \right\}$ .
- Subsequence  $\mathbf{x}_{j=3}^3 = \{3.6, 5.3, 5.4\}$ . Pattern  $\psi_{j=3}^3 = \{2, 1\}$ . This pattern is not in  $\Psi^3$ . Append it, and initialise its counter to 1:  $\Psi^3 = \left\{ \left\{ \{-1, -2\}, 1 \right\}, \dots, \left\{ \{-1, 2\}, 1 \right\}, \left\{ \{2, 1\}, 1 \right\} \right\}$ .
- Subsequence  $\mathbf{x}_{j=4}^3 = \{5.3, 5.4, 8.3\}$ . Pattern  $\psi_{j=4}^3 = \{1, 2\}$ . This pattern is not in  $\Psi^3$ . Append it, and initialise its counter to 1:  $\Psi^3 = \left\{ \left\{ \{-1, -2\}, 1 \right\}, \dots, \left\{ \{2, 1\}, 1 \right\}, \left\{ \{1, 2\}, 1 \right\} \right\}$ .
- Subsequence  $\mathbf{x}_{j=5}^3 = \{5.4, 8.3, 1.9\}$ . Pattern  $\psi_{j=5}^3 = \{2, -2\}$ . This pattern is not in  $\Psi^3$ . Append it, and initialise its counter to 1:  $\Psi^3 = \left\{ \left\{ \{-1, -2\}, 1 \right\}, \dots, \left\{ \{1, 2\}, 1 \right\}, \left\{ \{2, -2\}, 1 \right\} \right\}$ .
- Subsequence  $\mathbf{x}_{j=6}^3 = \{8.3, 1.9, 3.7\}$ . Pattern  $\psi_{j=6}^3 = \{-2, 2\}$ . This pattern is not in  $\Psi^3$ . Append it, and initialise its counter to 1:  $\Psi^3 = \left\{ \left\{ \{-1, -2\}, 1 \right\}, \dots, \left\{ \{2, -2\}, 1 \right\}, \left\{ \{-2, 2\}, 1 \right\} \right\}$ .

- Subsequence  $x_{j=7}^3 = \{1.9, 3.7, 8.6\}$ . Pattern  $\psi_{j=7}^3 = \{2, 2\}$ . This pattern is not in  $\Psi^3$ . Append it, and initialise its counter to 1:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 1 \}, \dots, \{ \{-2, 2\}, 1 \}, \{ \{2, 2\}, 1 \} \right\}$ .
- Subsequence  $x_{j=8}^3 = \{3.7, 8.6, 9.6\}$ . Pattern  $\psi_{j=8}^3 = \{2, 1\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 2:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 1 \}, \dots, \{ \{-1, 2\}, 1 \}, \{ \{2, 1\}, 2 \}, \dots \right\}$ .
- Subsequence  $x_{j=9}^3 = \{8.6, 9.6, 9\}$ . Pattern  $\psi_{j=9}^3 = \{1, -1\}$ . This pattern is not in  $\Psi^3$ . Append it, and initialise its counter to 1:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 1 \}, \dots, \{ \{2, 2\}, 1 \}, \{ \{1, -1\}, 1 \} \right\}$ .
- Subsequence  $x_{j=10}^3 = \{9.6, 9, 6\}$ . Pattern  $\psi_{j=10}^3 = \{-1, -2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 2:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots \right\}$ .
- Subsequence  $x_{j=11}^3 = \{9, 6, 8.7\}$ . Pattern  $\psi_{j=11}^3 = \{-2, 2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 2:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{-2, 2\}, 2 \}, \dots \right\}$ .
- Subsequence  $x_{j=12}^3 = \{6, 8.7, 6.7\}$ . Pattern  $\psi_{j=12}^3 = \{2, -2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 2:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{2, -2\}, 2 \}, \dots \right\}$ .
- Subsequence  $x_{j=13}^3 = \{8.7, 6.7, 3.3\}$ . Pattern  $\psi_{j=13}^3 = \{-2, -2\}$ . This pattern is not in  $\Psi^3$ . Append it, and initialise its counter to 1:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{1, -1\}, 1 \}, \{ \{-2, -2\}, 1 \} \right\}$ .
- Subsequence  $x_{j=14}^3 = \{6.7, 3.3, 2\}$ . Pattern  $\psi_{j=14}^3 = \{-2, -2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 2:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{1, -1\}, 1 \}, \{ \{-2, -2\}, 2 \} \right\}$ .
- Subsequence  $x_{j=15}^3 = \{3.3, 2, 2.5\}$ . Pattern  $\psi_{j=15}^3 = \{-2, 1\}$ . This pattern is not in  $\Psi^3$ . Append it, and initialise its counter to 1:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{-2, -2\}, 2 \}, \{ \{-2, 1\}, 1 \} \right\}$ .
- Subsequence  $x_{j=16}^3 = \{2, 2.5, 2.7\}$ . Pattern  $\psi_{j=16}^3 = \{1, 1\}$ . This pattern is not in  $\Psi^3$ . Append it, and initialise its counter to 1:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{-2, 1\}, 1 \}, \{ \{1, 1\}, 1 \} \right\}$ .
- Subsequence  $x_{j=17}^3 = \{2.5, 2.7, 4.6\}$ . Pattern  $\psi_{j=17}^3 = \{1, 2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 2:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{1, 2\}, 2 \}, \dots \right\}$ .
- Subsequence  $x_{j=18}^3 = \{2.7, 4.6, 9.1\}$ . Pattern  $\psi_{j=18}^3 = \{2, 2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 2:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{2, 2\}, 2 \}, \dots \right\}$ .
- Subsequence  $x_{j=19}^3 = \{4.6, 9.1, 1\}$ . Pattern  $\psi_{j=19}^3 = \{2, -2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 3:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{2, -2\}, 3 \}, \dots \right\}$ .
- Subsequence  $x_{j=20}^3 = \{9.1, 1, 3.1\}$ . Pattern  $\psi_{j=20}^3 = \{-2, 2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 3:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{-2, 2\}, 3 \}, \dots \right\}$ .
- Subsequence  $x_{j=21}^3 = \{1, 3.1, 1.7\}$ . Pattern  $\psi_{j=21}^3 = \{2, -2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 4:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{2, -2\}, 4 \}, \dots \right\}$ .
- Subsequence  $x_{j=22}^3 = \{3.1, 1.7, 4.1\}$ . Pattern  $\psi_{j=22}^3 = \{-2, 2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 4:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{-2, 2\}, 4 \}, \dots \right\}$ .

- Subsequence  $x_{j=23}^3 = \{1.7, 4.1, 3.8\}$ . Pattern  $\psi_{j=23}^3 = \{2, -1\}$ . This pattern is not in  $\Psi^3$ . Append it, and initialise its counter to 1:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{1, 1\}, 1 \}, \{ \{2, -1\}, 1 \} \right\}$ .
- Subsequence  $x_{j=24}^3 = \{4.1, 3.8, 6.4\}$ . Pattern  $\psi_{j=24}^3 = \{-1, 2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 2:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{-1, 2\}, 2 \}, \dots \right\}$ .
- Subsequence  $x_{j=25}^3 = \{3.8, 6.4, 1.3\}$ . Pattern  $\psi_{j=25}^3 = \{2, -2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 5:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{2, -2\}, 5 \}, \dots \right\}$ .
- Subsequence  $x_{j=26}^3 = \{6.4, 1.3, 5.7\}$ . Pattern  $\psi_{j=26}^3 = \{-2, 2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 5:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{-2, 2\}, 5 \}, \dots \right\}$ .
- Subsequence  $x_{j=27}^3 = \{1.3, 5.7, 3.4\}$ . Pattern  $\psi_{j=27}^3 = \{2, -2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 6:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{2, -2\}, 6 \}, \dots \right\}$ .
- Subsequence  $x_{j=28}^3 = \{5.7, 3.4, 2.4\}$ . Pattern  $\psi_{j=28}^3 = \{-2, -1\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 2:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 1 \}, \{ \{-2, -1\}, 2 \}, \dots \right\}$ .
- Subsequence  $x_{j=29}^3 = \{3.4, 2.4, 2.1\}$ . Pattern  $\psi_{j=29}^3 = \{-1, -1\}$ . This pattern is not in  $\Psi^3$ . Append it, and initialise its counter to 1:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{2, -1\}, 1 \}, \{ \{-1, -1\}, 1 \} \right\}$ .
- Subsequence  $x_{j=30}^3 = \{2.4, 2.1, 4.2\}$ . Pattern  $\psi_{j=30}^3 = \{-1, 2\}$ . This pattern is already in  $\Psi^3$ . Update its counter to 3:  $\Psi^3 = \left\{ \{ \{-1, -2\}, 2 \}, \dots, \{ \{-1, 2\}, 3 \}, \dots \right\}$ .

Once all the patterns have been processed, the resulting list of coincidences is  $\{2, 2, 3, 2, 2, 6, 5, 2, 1, 2, 1, 1, 1, 1\}$ . Normalising by the number of actual patterns found, 14, and applying Shannon entropy, the final SlopEn value for this time series is 5.29.

## Appendix B. SlopEn Source Code Implementation

This section includes a possible implementation of the SlopEn algorithm using C++ programming language (no optimisation, no error checking). It is implemented using a base-2 logarithm, but it can be replaced by a natural logarithm. This source is included below:

```
float fComputeSlopeEntropy(std::vector<float> vectorTimeSeries, int iEmbeddedDimension, float fGamma, float fDelta)
{
    struct structBin
    {
        int c;
        std::vector<int> vectorSymbols;
    };
    bool bFound;
    float fSlopEn, p, fSlope;
    std::vector<int> vectorSlopePattern;
    std::list<structBin> listPatternsFound;

    fSlopEn = 0.0;
    for(int j=0; j<vectorTimeSeries.size()-(iEmbeddedDimension-1); j++)
    {
        vectorSlopePattern.clear();
        for(int i=j+1; i<=iEmbeddedDimension; i++)
        {
            fSlope = vectorTimeSeries[i] - vectorTimeSeries[i-1];
            if(fabs(fSlope)<=fDelta) vectorSlopePattern.push_back(0); else
            if(fSlope>fDelta && fSlope<=fGamma) vectorSlopePattern.push_back(1); else
            if(fSlope>fGamma) vectorSlopePattern.push_back(2); else
            if(fSlope<-fDelta && fSlope>=-fGamma) vectorSlopePattern.push_back(-1); else
            vectorSlopePattern.push_back(-2);
        }
    }
}
```

```

bFound = false;
for(auto &patternInList : listPatternsFound)
{
    if (patternInList.vectorSymbols==vectorSlopePattern)
    {
        patternInList.c++;
        bFound = true;
        break;
    }
}
if(bFound==false) listPatternsFound.push_back({ 1,vectorSlopePattern });
}
for(auto &patternInList : listPatternsFound)
{
    p = (float)patternInList.c/listPatternsFound.size();
    fSlopEn += -p*log2(p);
}
return(fSlopEn);
}

```

## References

1. Kannathal, N.; Choo, M.L.; Acharya, U.R.; Sadasivan, P. Entropies for detection of epilepsy in (EEG). *Comput. Methods Programs Biomed.* **2005**, *80*, 187–194. [[CrossRef](#)] [[PubMed](#)]
2. Abásolo, D.; Hornero, R.; Espino, P.; Álvarez, D.; Poza, J. Entropy analysis of the EEG background activity in Alzheimer’s disease patients. *Physiol. Meas.* **2006**, *27*, 241. [[CrossRef](#)] [[PubMed](#)]
3. Pincus, S.; Gladstone, I.; Ehrenkranz, R. A regularity statistic for medical data analysis. *J. Clin. Monit. Comput.* **1991**, *7*, 335–345. [[CrossRef](#)] [[PubMed](#)]
4. Pincus, S.M. Approximate entropy as a measure of system complexity. *Proc. Natl. Acad. Sci. USA* **1991**, *88*, 2297–2301. [[CrossRef](#)]
5. Shannon, C.E.; Weaver, W. *The Mathematical Theory of Communication*; The University of Illinois Press: Urbana, IL, USA, 1949.
6. Rényi, A. *Probability Theory*; North-Holland Series in Applied Mathematics and Mechanics; Elsevier: Amsterdam, The Netherlands, 1970.
7. Tsallis, C. Possible generalization of Boltzmann-Gibbs statistics. *J. Stat. Phys.* **1988**, *52*, 479–487. [[CrossRef](#)]
8. Sinai, Y.G. About A. N. Kolmogorov’s work on the entropy of dynamical systems. *Ergod. Theory Dyn. Syst.* **1988**, *8*, 501–502. [[CrossRef](#)]
9. Richman, J.; Moorman, J.R. Physiological time-series analysis using Approximate Entropy and Sample Entropy. *Am. J. Physiol. Heart Circ. Physiol.* **2000**, *278*, H2039–H2049. [[CrossRef](#)]
10. Chen, W.; Zhuang, J.; Yu, W.; Wang, Z. Measuring complexity using FuzzyEn, ApEn, and SampEn. *Med. Eng. Phys.* **2009**, *31*, 61–68. [[CrossRef](#)]
11. Escudero, J.; Abásolo, D.; Simons, S. Classification of Alzheimer’s disease from quadratic sample entropy of electroencephalogram. *Healthc. Technol. Lett.* **2015**, *2*, 70–73.
12. Simons, S.; Espino, P.; Abásolo, D. Fuzzy Entropy Analysis of the Electroencephalogram in Patients with Alzheimer’s Disease: Is the Method Superior to Sample Entropy? *Entropy* **2018**, *20*, 21. [[CrossRef](#)]
13. Aboy, M.; Cuesta-Frau, D.; Austin, D.; Micó-Tormos, P. Characterization of Sample Entropy in the Context of Biomedical Signal Analysis. In Proceedings of the 2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Lyon, France, 22–26 August 2007; pp. 5942–5945.
14. Cuesta-Frau, D.; Novák, D.; Burda, V.; Molina-Picó, A.; Vargas, B.; Mraz, M.; Kavalkova, P.; Benes, M.; Haluzik, M. Characterization of Artifact Influence on the Classification of Glucose Time Series Using Sample Entropy Statistics. *Entropy* **2018**, *20*, 871. [[CrossRef](#)]
15. Alcaraz, R.; Abásolo, D.; Hornero, R.; Rieta, J. Study of Sample Entropy ideal computational parameters in the estimation of atrial fibrillation organization from the ECG. In Proceedings of the 2010 Computing in Cardiology, Belfast, UK, 26–29 September 2010; pp. 1027–1030.
16. Lu, S.; Chen, X.; Kanters, J.K.; Solomon, I.C.; Chon, K.H. Automatic Selection of the Threshold Value  $r$  for Approximate Entropy. *IEEE Trans. Biomed. Eng.* **2008**, *55*, 1966–1972. [[PubMed](#)]
17. Costa, M.; Goldberger, A.L.; Peng, C.K. Multiscale entropy analysis of biological signals. *Phys. Rev. E* **2005**, *71*, 021906. [[CrossRef](#)] [[PubMed](#)]

18. Wu, S.D.; Wu, C.W.; Lin, S.G.; Lee, K.Y.; Peng, C.K. Analysis of complex time series using refined composite multiscale entropy. *Phys. Lett. A* **2014**, *378*, 1369–1374. [[CrossRef](#)]
19. Lempel, A.; Ziv, J. On the Complexity of Finite Sequences. *IEEE Trans. Inf. Theory* **1976**, *22*, 75–81. [[CrossRef](#)]
20. Bandt, C.; Pompe, B. Permutation Entropy: A Natural Complexity Measure for Time Series. *Phys. Rev. Lett.* **2002**, *88*, 174102. [[CrossRef](#)]
21. Aboy, M.; Hornero, R.; Abasolo, D.; Alvarez, D. Interpretation of the Lempel-Ziv Complexity Measure in the Context of Biomedical Signal Analysis. *IEEE Trans. Biomed. Eng.* **2006**, *53*, 2282–2288. [[CrossRef](#)]
22. Cuesta-Frau, D.; Murillo-Escobar, J.P.; Orrego, D.A.; Delgado-Trejos, E. Embedded Dimension and Time Series Length. Practical Influence on Permutation Entropy and Its Applications. *Entropy* **2019**, *21*, 385. [[CrossRef](#)]
23. Li, D.; Liang, Z.; Wang, Y.; Hagihira, S.; Sleigh, J.W.; Li, X. Parameter selection in permutation entropy for an electroencephalographic measure of isoflurane anesthetic drug effect. *J. Clin. Monit. Comput.* **2013**, *27*, 113–123. [[CrossRef](#)]
24. Cuesta-Frau, D.; Varela-Entrecanales, M.; Molina-Picó, A.; Vargas, B. Patterns with Equal Values in Permutation Entropy: Do They Really Matter for Biosignal Classification? *Complexity* **2018**, *2018*, 1–15. [[CrossRef](#)]
25. Riedl, M.; Müller, A.; Wessel, N. Practical considerations of permutation entropy. *Eur. Phys. J. Spec. Top.* **2013**, *222*, 249–262. [[CrossRef](#)]
26. Zunino, L.; Olivares, F.; Scholkmann, F.; Rosso, O.A. Permutation entropy based time series analysis: Equalities in the input signal can lead to false conclusions. *Phys. Lett. A* **2017**, *381*, 1883–1892. [[CrossRef](#)]
27. Liu, T.; Yao, W.; Wu, M.; Shi, Z.; Wang, J.; Ning, X. Multiscale permutation entropy analysis of electrocardiogram. *Phys. A Stat. Mech. Its Appl.* **2017**, *471*, 492–498. [[CrossRef](#)]
28. Gao, Y.; Vilecco, F.; Li, M.; Song, W. Multi-Scale Permutation Entropy Based on Improved LMD and HMM for Rolling Bearing Diagnosis. *Entropy* **2017**, *19*, 176. [[CrossRef](#)]
29. Azami, H.; Escudero, J. Amplitude-aware permutation entropy: Illustration in spike detection and signal segmentation. *Comput. Methods Programs Biomed.* **2016**, *128*, 40–51. [[CrossRef](#)]
30. Traversaro, F.; Risk, M.; Rosso, O.; Redelico, F. An empirical evaluation of alternative methods of estimation for Permutation Entropy in time series with tied values. *arXiv* **2017**, arXiv:1707.01517.
31. Fadlallah, B.; Chen, B.; Keil, A.; Príncipe, J. Weighted-permutation entropy: A complexity measure for time series incorporating amplitude information. *Phys. Rev. E* **2013**, *87*, 022911. [[CrossRef](#)]
32. Liu, X.-F.; Wang, Y. Fine-grained permutation entropy as a measure of natural complexity for time series. *Chin. Phys. B* **2009**, *18*, 2690.
33. Cuesta-Frau, D. Permutation entropy: Influence of amplitude information on time series classification performance. *Math. Biosci. Eng.* **2019**, *16*, 6842. [[CrossRef](#)]
34. Koski, A.; Juhola, M.; Meriste, M. Syntactic recognition of ECG signals by attributed finite automata. *Pattern Recognit.* **1995**, *28*, 1927–1940. [[CrossRef](#)]
35. Koski, A. Primitive coding of structural ECG features. *Pattern Recognit. Lett.* **1996**, *17*, 1215–1222. [[CrossRef](#)]
36. Lake, D.E.; Richman, J.S.; Griffin, M.P.; Moorman, J.R. Sample entropy analysis of neonatal heart rate variability. *Am. J. Physiol.-Regul. Integr. Comp. Physiol.* **2002**, *283*, R789–R797. [[CrossRef](#)] [[PubMed](#)]
37. Wessel, N.; Ziehmman, C.; Kurths, J.; Meyerfeldt, U.; Schirdewan, A.; Voss, A. Short-term forecasting of life-threatening cardiac arrhythmias based on symbolic dynamics and finite-time growth rates. *Phys. Rev. E Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.* **2000**, *61*, 733–739. [[CrossRef](#)] [[PubMed](#)]
38. Cysarz, D.; Lange, S.; Matthiessen, P.; Van Leeuwen, P. Regular heartbeat dynamics are associated with cardiac health. *Am. J. Physiol. Regul. Integr. Comp. Physiol.* **2007**, *292*, R368–R372. [[CrossRef](#)]
39. Parlitz, U.; Berg, S.; Luther, S.; Schirdewan, A.; Kurths, J.; Wessel, N. Classifying cardiac biosignals using ordinal pattern statistics and symbolic dynamics. *Comput. Biol. Med.* **2012**, *42*, 319–327. [[CrossRef](#)]
40. Zunino, L.; Zanin, M.; Tabak, B.M.; Pérez, D.G.; Rosso, O.A. Forbidden patterns, permutation entropy and stock market inefficiency. *Phys. A Stat. Mech. Its Appl.* **2009**, *388*, 2854–2864. [[CrossRef](#)]
41. Amigó, J. *Permutation Complexity in Dynamical Systems*; Springer Series in Synergetics: Heidelberg, Germany, 2010.
42. Cuesta-Frau, D.; Molina-Picó, A.; Vargas, B.; González, P. Permutation Entropy: Enhancing Discriminating Power by Using Relative Frequencies Vector of Ordinal Patterns Instead of Their Shannon Entropy. *Entropy* **2019**, *21*, 1013. [[CrossRef](#)]



43. Deng, B.; Cai, L.; Li, S.; Wang, R.; Yu, H.; Chen, Y.; Wang, J. Multivariate multi-scale weighted permutation entropy analysis of EEG complexity for Alzheimer's disease. *Cogn. Neurodynam.* **2017**, *11*, 217–231. [[CrossRef](#)]
44. Keller, K.; Unakafov, A.M.; Unakafova, V.A. Ordinal Patterns, Entropy, and EEG. *Entropy* **2014**, *16*, 6212–6239. [[CrossRef](#)]
45. Li, P.; Karmakar, C.; Yan, C.; Palaniswami, M.; Liu, C. Classification of 5-S Epileptic EEG Recordings Using Distribution Entropy and Sample Entropy. *Front. Physiol.* **2016**, *7*, 136. [[CrossRef](#)]
46. Andrzejak, R.G.; Lehnertz, K.; Mormann, F.; Rieke, C.; David, P.; Elger, C.E. Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. *Phys. Rev. E* **2001**, *64*, 061907. [[CrossRef](#)] [[PubMed](#)]
47. Baumert, M.; Czipelova, B.; Ganesan, A.; Schmidt, M.; Zaunseder, S.; Javorka, M. Entropy Analysis of RR and QT Interval Variability during Orthostatic and Mental Stress in Healthy Subjects. *Entropy* **2014**, *16*, 6384–6393. [[CrossRef](#)]
48. Liu, C.; Li, K.; Zhao, L.; Liu, F.; Zheng, D.; Liu, C.; Liu, S. Analysis of Heart Rate Variability Using Fuzzy Measure Entropy. *Comput. Biol. Med.* **2013**, *43*, 100–108. [[CrossRef](#)] [[PubMed](#)]
49. Mayer, C.C.; Bachler, M.; Hörtenhuber, M.; Stocker, C.; Holzinger, A.; Wassertheurer, S. Selection of entropy-measure parameters for knowledge discovery in heart rate variability data. *BMC Bioinform.* **2014**, *15*, S2. [[CrossRef](#)] [[PubMed](#)]
50. Goldberger, A.L.; Amaral, L.A.N.; Glass, L.; Hausdorff, J.M.; Ivanov, P.C.; Mark, R.G.; Mietus, J.E.; Moody, G.B.; Peng, C.K.; Stanley, H.E. PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. *Circulation* **2000**, *101*, 215–220. [[CrossRef](#)] [[PubMed](#)]
51. Iyengar, N.; Peng, C.K.; Morin, R.; Goldberger, A.L.; Lipsitz, L.A. Age-related alterations in the fractal scaling of cardiac interbeat interval dynamics. *Am. J.-Physiol.-Regul. Integr. Comp. Physiol.* **1996**, *271*, R1078–R1084. [[CrossRef](#)] [[PubMed](#)]
52. Bagnall, A.; Lines, J.; Bostrom, A.; Large, J.; Keogh, E. The great time series classification bake off: A review and experimental evaluation of recent algorithmic advances. *Data Min. Knowl. Discov.* **2017**, *31*, 606–660. [[CrossRef](#)]
53. Lines, J.; Bagnall, A.; Caiger-Smith, P.; Anderson, S. Classification of Household Devices by Electricity Usage Profiles. In *Intelligent Data Engineering and Automated Learning—IDEAL 2011*; Yin, H., Wang, W., Rayward-Smith, V., Eds.; Springer: Berlin/Heidelberg, Germany, 2011; pp. 403–412.
54. Flood, M.; Jensen, B.; Malling, A.S.; Lowery, M. Increased EMG Intermuscular Coherence and Reduced Signal Complexity in Parkinson's Disease. *Clin. Neurophysiol.* **2019**, *130*, 259–269. [[CrossRef](#)]
55. Tang, X.; Zhang, X.; Gao, X.; Xiang, C.; Zhou, P. A Novel Interpretation of Sample Entropy in Surface Electromyographic Examination of Complex Neuromuscular Alternations in Subacute and Chronic Stroke. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2018**, *26*, 1878–1888. [[CrossRef](#)]
56. Zhu, X.; Zhang, X.; Tang, X.; Gao, X.; Xiang, C. Re-Evaluating Electromyogram—Force Relation in Healthy Biceps Brachii Muscles Using Complexity Measures. *Entropy* **2017**, *19*, 624. [[CrossRef](#)]
57. Bingham, A.; Arjunan, S.P.; Jelfs, B.; Kumar, D.K. Normalised Mutual Information of High-Density Surface Electromyography during Muscle Fatigue. *Entropy* **2017**, *19*, 697. [[CrossRef](#)]
58. Montesinos, L.; Castaldo, R.; Pecchia, L. On the use of approximate entropy and sample entropy with centre of pressure time-series. *J. Neuroeng. Rehabil.* **2018**, *15*, 116. [[CrossRef](#)]
59. Manis, G.; Aktaruzzaman, M.; Sassi, R. Bubble Entropy: An Entropy Almost Free of Parameters. *IEEE Trans. Biomed. Eng.* **2017**, *64*, 2711–2718. [[PubMed](#)]
60. Li, D.; Li, X.; Liang, Z.; Voss, L.J.; Sleight, J.W. Multiscale permutation entropy analysis of EEG recordings during sevoflurane anesthesia. *J. Neural Eng.* **2010**, *7*, 046010. [[CrossRef](#)] [[PubMed](#)]
61. Costa, M.; Goldberger, A.L.; Peng, C.K. Multiscale Entropy Analysis of Complex Physiologic Time Series. *Phys. Rev. Lett.* **2002**, *89*, 068102. [[CrossRef](#)] [[PubMed](#)]
62. Kalpić, D.; Hlupić, N.; Lovrić, M. Student's t-Tests. In *International Encyclopedia of Statistical Science*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 1559–1563.
63. Zoubir, A.M.; Iskander, D.R. *Bootstrap Techniques for Signal Processing*; Cambridge University Press: Cambridge, UK, 2004.

64. Hari, V.N.; Anand, G.V.; Premkumar, A.B.; Madhukumar, A.S. Design and Performance Analysis of a Signal Detector Based on Suprathreshold Stochastic Resonance. *Signal Process.* **2012**, *92*, 1745–1757. [[CrossRef](#)]
65. Greenwood, P.E.; Müller, U.U.; Ward, L.M.; Wefelmeyer, W. Statistical Analysis of Stochastic Resonance in a Thresholded Detector. *Austrian J. Stat.* **2016**, *32*, 49–70.



© 2019 by the author. 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/>).