A Statistically Significant Test to Evaluate the Order or Disorder for a Binary String of a Finite Length

Jonathan Blackledge
Technological University Dublin, jonathan.blackledge@tudublin.ie

N. Mosola
University of KwaZulu-Natal

Recommended Citation

J M Blackledge & N Mosola (2020) A Statistically Significant Test to Evaluate the Order or Disorder for a Binary String of a Finite Length, *ISSC2020, IEEE UK and Ireland Signal Processing Chapter and IEEE Computational Intelligence Society (UK & Ireland), Letterkenny Institute of Technology, 11-12, June 2020.*

A Statistically Significant Test to Evaluate the Order or Disorder of a Binary String

J. M. Blackledge
School of Electrical and Electronic Engineering,
Technological University Dublin;
Department of Computer Science,
University of Western Cape.
jonathan.blackledge@tudublin.ie

N. Mosola
School of Mathematics, Statistics and Computer Science,
University of KwaZulu-Natal;
Department of Computer Science,
National University of Lesotho.
nn.mosola@nul.ls

***Abstract*—This paper addresses a basic problem in regard to the analysis of a finite binary string or bit stream (of compact support), namely, how to tell whether the string is representative of non-random or intelligible information (involving some form of periodicity, for example), whether it is the product of an entirely random process or whether it is something in between the two. This problem has applications that include cryptanalysis, quantitative finance, machine learning, artificial intelligence and other forms of signal and image processing involving the general problem of how to distinguishing real noise from information embedded in noise, for example. After providing a short introduction to the problem, we focus on the application of information entropy for solving the problem given that this fundamental metric is an intrinsic measure on information in regard to some measurable system. A brief overview on the concept of entropy is given followed by examples of how algorithms can be design to compute the binary entropy of a finite binary string including important variations on a theme such as the BiEntropy. The problem with computing a single metric of this type is that it can be representative of similar binary strings and lacks robustness in terms of its statistically significance. For this reasons, the paper presents a solution to the problem that is based on the Kullback-Leibler Divergence (or Relative Entropy) which yields a measure of how one probability distribution is different from another reference probability distribution. By repeatedly computing this metric for different reference (simulated or otherwise) random finite binary strings, it is shown how the distribution of the resulting signal changes for intelligible and random binary strings of a finite extent. This allows a number of standard statistical metrics to be computed from which the foundations for a machine learning system can be developed. A limited number of results are present for different natural languages to illustrate the approach, a prototype MATLAB function being provide for interested readers to reproduce the results given as required, investigate different data sets and further develop the method considered.**

***Index Terms*—Binary Strings, Randomness, Entropy, Binary Information Entropy, Kullback-Leibler Divergence, Machine Learning.**

I. Introduction

The term ‘intelligibility’ usually applies to the clarity of speech and/or writing and whether they are clear enough to be understood. In this work, the term refers to whether or not a binary string is genuinely random or otherwise (or a mixture of both) where, in the former case, it is assumed that a binary string is a binary representation of natural noise, for example. One of the keys to doing this is to analyze binary strings in terms of their information entropy which is re-visited in the following section.

II. Information and Entropy

The first and arguably the most important relationship between information and entropy was first established by Leo Szilard as a result of his solution to the ‘Maxwell demon’ thought experiment, named after James Clerk Maxwell. Maxwell first proposed this thought experiment as a result of his work on the properties of ideal gases in the 1860’s. He considered a model where gas particles are free to move inside a container whose interactions occur through elastic collisions in which they exchange energy and momentum with each other that is consistent with their thermal environment. This model is compounded in the Maxwell-Boltzmann Probability Density Function $P(v)$ for the velocities v of identical gas particles with a mass m given by

$$p(v)=\left(\frac{m}{2\pi k_BT}\right)^{\frac{3}{2}}\exp\left(-\frac{mv^2}{2k_BT}\right)$$

where T is the thermodynamic equilibrium temperature of the gas in $^\circ$K and $k_B\simeq 1.4\times 10^{-23}\text{JK}^{-1}$ (Joules per Kelvin) is the Boltzmann constant which relates the average kinetic energy of particles in an ideal gas with the temperature of that gas. The mode of this distribution gives the most probable velocity of a particle, i.e. $v_p=\sqrt{2k_BT/m}$.

The thought experiment considers a ‘demon’ operating a frictionless shutter placed at the center of a container which partitions the container into two sections. The shutter can be opened to allow particles with a velocity $v<v_p$ to enter into one section of the container and particles with velocity $v\geq v_p$ to enter into the other section, where both the container and shutter are perfectly thermally isolated. In this way, high and low velocity gas particles are separated into the two sections of the container, preserving their velocities. Consequently, the equilibrium temperatures of the two sections become higher and lower than that of the original container, respectively.

For a classical thermodynamic process, work W can only take place when there is a temperature gradient, and, for

978-1-7281-9418-9/20/$31.00 © 2020 IEEE

an irreversible process, the Entropy S always increases, the change in entropy ΔS being given by $\Delta S = \Delta W/T$. This is the basis for the second law of thermodynamics, a law that appears to be broken according to the thought experiment considered above because the entropy of the two sections is now different, and, given that there is an increase in temperature in one of the sections, the entropy has been lowered without expending energy.

In Leo Szilard's 1922 doctoral dissertation and companion landmark paper [1], he showed how the paradox can be solved by taking into account the fact that in order for the demon to open and close the shutter to let particles of different velocities through, a decision must be made, a decision that is based on gathering information on the velocity of the particle before it is let through the shutter *a priori*. The information measured is taken to provide a 'balance' to the decrease in the physical entropy and is compounded in the 'Information Entropy'. In this context, Szilard's principal contribution was to consider that the demon must be an 'intelligent being' that can make a decision based on *a priori* information on the velocity of a particle, a critical issue, that Maxwell had failed to conceive of and include in his original thought experiment.

Szilard's original concept on information entropy has become the basis of information theory. He showed that there is an increase of $k_B \log_2 2$ units of entropy in any measurement. This concept was independently 'discovered' by Claude Shannon in 1949 [2] (to whom credit is usually given) and Andre Kolmogorov and Yakov Sinai, who developed a modified form in 1959 [3]. In developing a solution to a paradox in thermodynamics, Leo Szilard introduced an idea that is arguably the single most important icon of the information revolution of today. This is because information entropy provides the key for estimating the (average) minimum number of bits needed to encode a string of symbols, based on the frequency of those symbols.

In statistical mechanics, entropy is a measure of the number of ways in which a system may be arranged, often taken to be a measure of 'disorder' where the higher the entropy, the higher the disorder. Another way of interpreting this metric is in terms of it being a measure of the lack of information available on the exact state of a system. Shannon entropy is a measure of the information required to determine precisely a systems state from all possible states, and is expressed in binary digits, or 'Bits'.

More generally, information entropy is a measure of order, a universal measure applicable to any structure or any system. It quantifies the instructions that are needed to produce a certain organization. There are several ways in which one can quantify information but a specially convenient one is in terms of binary choices. We compute the information inherent in any given arrangement from the number of choices that we must make to arrive at that particular arrangement among all possible arrangements. Intuitively, the more arrangements that are possible, the more information that is required to achieve a particular arrangement.

A. Shannon Entropy

Consider a digital signal $s_m, \ m = 1, 2, ..., M$ composed of M (real) values. Let the (discrete) probability distribution function or histogram that a specific value s_n occurs within a bin in the signal be $p_n, n = 1, 2, ..., N$ where N is the number of bins. The information associated with an outcome s_m is $-\log p_n$ which is a measure of the information required to specify s_m in terms of it being a member or a subset or bin where p_n is the distribution of bins. The mean value μ say, of s_m is equal to the sum over every possible value n weighted by p_n of that value, i.e.

$$\mu = \sum_{n=1}^{N} n p_n$$

Similarly, the Shannon Information Entropy (usually denoted by S), is a measure of the mean (in this context, the 'expected value') of the information measure $-\log p_n$ weighted by p_n and is given by the dimensionless quantity

$$S = -\sum_{n=1}^{N} p_n \log p_n$$

The higher the entropy of a signal becomes the greater is ambiguity, and, in this context, information entropy is a measure of the unpredictability or randomness of any message contained in the signal. This is typically determined by the noise that distorts the information contained in a signal. In general, the information entropy associated with the transmission of information in a signal tends to increase with time. This is due to the increase in noise that distorts the signal as it propagates, the sources of this noise being multi-faceted and tending to Gaussian noise as a consequence of the Central Limit Theorem.

B. Boltzmann Entropy

The partner entity to the information entropy in physics has a dimension called 'Entropy' first introduced by Ludwig Boltzmann and J. Willard Gibbs as a measure of the dispersal of energy; in a sense, a measure of disorder, just as information is a measure of order. In fact, Boltzmann's entropy concept has the same mathematical roots as Shannon's information concept in terms of computing the probabilities of sorting objects into bins. In statistical mechanics, the Boltzmann Entropy is defined as

$$E = -k_B \sum_{n=1}^{N} p_n \ln p_n$$

Shannon's and Boltzmann's definitions of entropy are similar given that S and E differ only by their scaling factors.

In the definition of the Boltzmann entropy, the probabilities p_n refer to energy levels of a 'classical system' (e.g. a collection of classical Newtonian particles). With the information entropy, p_n is not assigned *a priori* such specific roles and the expression can be applied to any physical system to provide a measure of order. Thus, information becomes a concept equivalent to physical entropy and any system can be described in terms of one or the other. An increase in information entropy implies a decrease of information.

C. Renyi Entropy

As with many other fundamental definitions in mathematics and physics, so the information entropy has a number of 'variations on a theme'. A generalization of the Shannon entropy is the Renyi Entropy H_α (of order a) given by

$$H_\alpha = \frac{1}{1-\alpha}\log\sum_{n=1}^{N} p_n^\alpha,\ \alpha \geq 0,\ \alpha \neq 1$$

where

$$\lim_{\alpha\to 1} H_\alpha = \sum_{n=1}^{N} p_n \log p_n$$

which recovers the Shannon entropy. From this generalization, a number of complementary information entropy measures are obtained when $\alpha = 0$ ('maximum entropy'), $\alpha = 2$ ('collision entropy') and $H_\infty = -\log[\max p_n]$ ('minimum entropy'), for example.

D. Binary Information Entropy

In the case of a binary string f_n composed of L bits (i.e. the bit-stream $\{0,1\}^L$) the elements of the string can take on only two values, 0 and 1, which are mutually exclusive. In this case, the Binary Information Entropy (BIE) function denoted by H becomes

$$H = -\sum_{n=1}^{2} p_n \log_2 p_n \tag{1}$$

$$= -p\log_2 p - (1-p)\log_2(1-p) \text{ Bits}$$

where, if we let p denote the probability of 1 occurring in the binary string, then the probability of obtaining a 0 in the same string is $1-p$. Similarly, if p is taken to denote the probability of 0 occurring in the string, then the probability of obtaining 1 is $(1-p)$. In either case, $0\log_2 0 \equiv 0$ and $H(p) = H(1-p)$.

III. Evaluating Order and Disorder of a Binary String using Binary Information Entropy

Given a binary string, our problem is to evaluate whether the string is a binary representation of noise or whether it contains intelligible information in terms of it having some degree of determinism. This could include any natural language that has evolved through use, application and repetition without conscious planning but binary coded in a planned a premeditated way, e.g. the ASCII. The purpose is therefore to establish a method by which a finite binary string of arbitrary length can be compared against another string (of the same length) in terms of the relative order and/or disorder of all of its bits. Applying a basic binary entropy test is not sufficient. This is primarily because of its failure to differentiate between binary strings that include periodicity, for example, and are therefore not random. For perfectly ordered binary strings whose elements are all equal to 1 or all equal 0 and when $p = 0$ or $p = 1$, $H(p) = 0$. On the other hand, when $p = 0.5$, $H(p) = 1$ reflecting maximum irregularity. However, for a string such as 01010101 which has a repeating pattern of 01, $p = 0.5$ and $H(p) = 1$ so that in this case, the value of $H(p)$ appears to represent a random binary string when in reality, the sting is periodic and therefore not random at all. What is required is an entropy based metric that differentiates more fully than is possible with Equation (1) alone. This is required in the context of Table I which suggests how we might intuitively regard the order and disorder of some 8 bit binary strings.

TABLE I
An intuitive insight of order and disorder associated with some short binary strings (from [5]).

Binary String	Description	Reason
11111111	Perfectly ordered	All 1's
00000000	Perfectly ordered	All 0's
01010110	Mostly ordered	Mostly 01's
01010101	Regular, not disordered	Repeating 01's
11001100	Regular, not disordered	Repeating 1100's
01011010	Mostly ordered	0101 then 1010
01101011	Somewhat disordered	No apparent pattern
10110101	Somewhat disordered	No apparent pattern

There have been a number of algorithms designed to compute various entropy-based metrics for the determination or measurement of randomness, regularity, irregularity, order, disorder and entropy for binary and other strings, e.g. [6] and [7]. These include measures that aim to characterize randomness, disorder through the entropy of finite strings such Approximate Entropy [8], [9], [10], [11] Sample Entropy [12] and Fuzzy Entropy [13]. Such metrics are based on algorithms that can classified as follows: (i) moving window methods that examine sub-strings of the original string [14]; (ii) algorithms which generate a metric based on the entire string length [15], [16]. Applications include basic randomness tests [17], cryptanalysis [18], [19] and [20]. This includes the development of a *BiEntropy* (BiEn) measure which is based upon a weighted average of the Shannon entropy for all but the last binary derivatives of the string given by [5]

$$\text{BiEn} = \frac{1}{2^{N-1}-1}\sum_{n=0}^{N-1}[-p_n\log_2 p_n - (1-p_n)\log_2(1-p_n)]2^n$$

which is suitable for shorter binary strings where $N \leq 32$ (approximately). This result is based on the Shannon binary entropies of the string and the first n-2 binary derivatives using a simple power law. Another version of this BiEntropy metric which is based on logarithmic weighting and can be used for longer binary strings is given by [5]

$$\text{BiEn} = \frac{\sum\limits_{n=0}^{N-1}[-p_n\log_2 p_n - (1-p_n)\log_2(1-p_n)]w_n}{\sum\limits_{n=0}^{n-2} w_n}$$

where $w_n = \log_2(n+2)$ for $N > 32$ (approximately). The logarithmic weighting provides greater weight to the higher derivatives, and, depending upon the application, other weightings can be used. This is one a many studies that have been undertaken to develop suitable tests and measures of order, disorder, randomness, irregularity and entropy based on the computation of a single metric.

While desirable computationally, focusing on the use of a single metric for this purpose is restrictive and may be statistically insignificant because of its self-selecting data predication. For this reason, in the following section, we consider a complementary approach to the problem which is based on the application of the Kullback-Leibler Divergence for a stream of data that yields a statistically significant result as apposed to a single metric. This provides the foundations for an application of a machine learning approach as discussed later.

IV. Application of Kullback-Leibler Divergence

Since intelligibility is a relative concept, a relative metric should be considered which provides a measure of how a binary string compares in some way with a string that is known to be the product of a genuinely random process. Further, this comparison needs to be undertaken on a statistical basis, measuring how one probability distribution associated with the binary string compares to a reference probability distribution in terms of its information content.

We consider a solution to this problem using the Kullback-Leibler Divergence or Relative (Binary) Entropy function given by

$$R = -\sum_{n=1}^{2} p_n \log_2 \left(\frac{q_n}{p_n} \right)$$

where p_n is the binary histogram of binary string $f_n \equiv \{0,1\}^L$ and q_n is the binary histogram of some reference binary string $g_n \equiv \{0,1\}^L$, both strings being of finite length L. Suppose string f_n is a non-random 'intelligible string' (e.g. a binary string representation of some text from a natural language) and g_n is a genuine random string. We require the metric R to be significantly different in terms of its numerical value to the case when both f_n and g_n are random binary strings. Ideally, what is required is to establish a threshold for the value of R below which f_n can be classified as intelligible say, and above which, f_n can be classified as random. However, this assumes that a binary decision making process can be applied which may not be statistically significant and does not consider any transition from f_n being random to intelligible or vice versa.

Instead, we consider an analysis of the relative entropy based on an interpretation of the statistical difference between the case when f_n is intelligible and g_n is random and when both f_n and g_n are random strings. Thus, we compute the Relative Entropy Signal (RES) given by

$$R_m = -\sum_{n=1}^{2} p_{nm} \log_2 \left(\frac{q_{nm}}{p_{nm}} \right), \; m = 1, 2, ..., M \qquad (2)$$

where q_{nm} denotes the m^{th} binary histogram of the m^{th} random bit-stream. We then consider the following cases (which are referred to as such in regard to presenting the results that follow):

Case (i): p_{nm} is the m^{th} binary histogram of a non-random binary string;
Case (ii): p_{nm} is the m^{th} binary histogram of a random binary string.

In the results that follow, the non-random string is obtained by generating the binary representation of the English text associated with the Abstract of this paper, achieved using the ASCII text to binary converter available at [21] (with the delimiter string set to none). A sequence of random binary arrays are generated using the MATLAB uniform distributed random number generator function *rand* (which returns floating point numbers in the interval [0,1]) and applying a round transformation (to output an array consisting of 0's and 1's), each array having the same length L and being independent of any other array in terms of the pattern of elements that is generated. For each array of random bits, Equation (2) is computed where $p_{nm} = p_n$, $\forall m$.

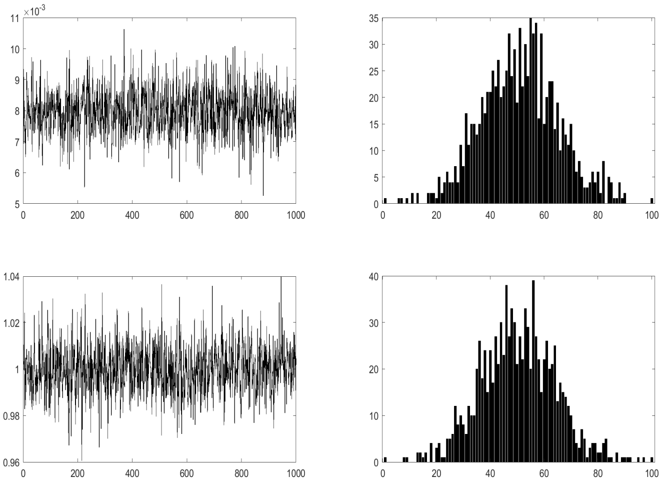

Fig. 1. Plots of the R_m (left) given by Equation (2) and the corresponding 100-bin histograms (right) for Case (i) - above - and Case (ii) - below, respectively, with $M = 1000$.

Figure 1 shows example signals of the RES given by Equation (2) for Case (i) and Case (ii) above with $M = 1000$ and the corresponding 100-bin histograms. It is immediately clear that:

- For Case (i), when one of the strings is intelligible, $R_m > 0 \forall m$ and has a non-zero mean Gaussian-type distribution with a defined mean value $M_{(\text{i})}$, say.
- For Case (ii), when all strings are random, $R_m > 0$ has a non-zero mean Gaussian-type distribution with a defined mean $M_{(\text{ii})} > M_{(\text{i})}$. However, in both cases, application of the Jargue-Bera (JB) test for normality shows that the JB statistic is significantly larger than χ^2 and the null hypothesis must therefore rejected, i.e. the series R_m does not conform to a normal distribution.

From Figure 1, a principal observation is that the statistical characteristics of R_m for Case (i) and Case (ii) above are different. For example, the difference in the mean of the RES for the two cases is at least one order of magnitude and can therefore be used to differentiate between an intelligible and a random binary string. We thus consider additional statistical measures to the mean value alone.

Figure 2, shows logarithmic plots of the mean, standard deviation (std), the median (med) and the mode for Case (i) and Case (ii) using four natural languages. These are translations of the Abstract for this paper, obtained with the online translator [22] and converting the translations to binary strings using the text to binary converter available at [21] (with the delimiter string set to none). This result shows that an intelligible binary string can be clearly differentiated from a random string by computing the RES in the way described above. The results given in Figure 2 illustrate that the statistical characteristics of R_m change significantly for at least four different natural languages. This is quantified in terms of the numerical values of the metrics considered and presented on a logarithmic scale in Figure (2) .

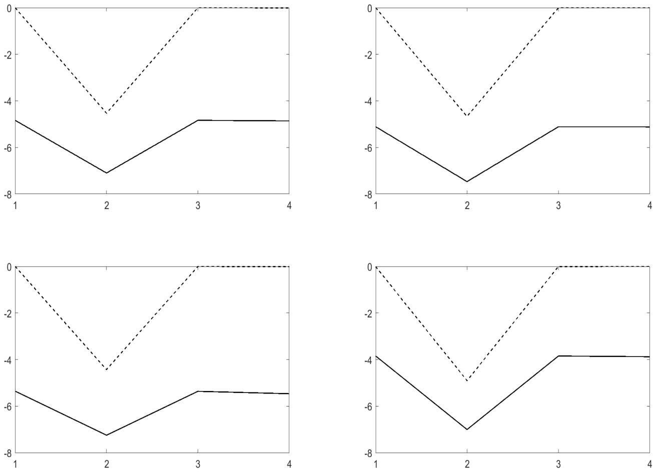

Fig. 2. Log-linear signatures of the mean, standard deviation, the median and the mode (from left to right in each line plot - points 1, 2, 3 and 4, respectively) for Case (i) - solid line - and Case (ii) - dashed line. The top-left plot shows the result for English, the top-right plot is the result for Arabic, the lower-left plot for Chinese (traditional) and the lower-left plot for Greek.

In each case, the four metrics considered adhere to the following conditions:

$$\text{mean}[R_m]_{\text{(i)}} < \text{mean}[R_m]_{\text{(ii)}},\ \text{std}[R_m]_{\text{(i)}} < \text{std}[R_m]_{\text{(ii)}},$$

$$\text{med}[R_m]_{\text{(i)}} < \text{med}[R_m]_{\text{(ii)}},\ \text{mode}[R_m]_{\text{(i)}} < \text{mode}[R_m]_{\text{(ii)}}$$

where the subscripts (i) and (ii) denote the two cases considered. The difference in the standard deviations between the two cases is less significant than the other parameters. Nevertheless, the mean, standard deviation, median and mode do provide differentiators between the two cases that are clearly statistically significant.

The natural languages used for this exercise have been chosen for their structural and semantic differences, a more comprehensive study in this regard using a broader spectrum of natural languages as well as other non-random and semi-random digital signals lying beyond the scope of this work. In this respect, the MATLAB code given in Appendix A - function RBET (Relative Binary Entropy Test) - used to generate the four metrics considered in Figure 2 is provided for readers to reproduce the results given and to investigate the focus of this approach for any other class of binary strings derived from a range of different digital signals and other sources.

In order to achieve a statistically significant results of this type, it is necessary to use relatively long binary strings $L >> 1$ and values for the length of the RES $M >> 1$. It is important to note, that this test on the intelligibility of a binary string is predicated on the term 'intelligibility' being associated with a natural language only. This is a limited definition of the term and has only been considered in regard to developing the tests studied in this work. In general, the term 'intelligibility' should be applied to a binary string that can be considered to be the result of a process that is other than an entirely random processes.

V. Machine Learning

Given that the demarcation between an intelligible and a random binary string can be determined by applying the relative entropy test as discussed Section IV, the potential exists to compute further statistical metrics and other parameters based on an analysis of the signatures given in Figure 2. These may include the statistical moments and spectral properties of R_m, for example, designed to develop a feature vector whose purpose is to provide a multi-class classification used to input into an Artificial Neural Network (ANN). Four components of such a feature vector could be the mean, standard deviation, median and mode of the RES as considered in the results presented in Section IV. The value of such an approach in regard to the recent growth in Deep Learning using deep ANNs operating on the binary strings themselves remains to be quantified.

VI. Discussion and Conclusions

Since all data can be expressed as a binary string, irrespective of the code used to do so (i.e. ASCII or otherwise), the approach considered in this paper provides a relatively generic method of differentiating between random and non-random data. Unlike other approaches, as discussed in Section III, the method presented in Section IV is based on generating a signal computed using Equation (2) and characterizing the distribution of this signal. It has been shown that for a limited number of natural languages there is a significant difference in the distribution of R_m given by Equation (2) and random data. This illustrates that the method can differentiate between random and intelligible binary streams, at least for a natural language. Some common statistical metrics have been used to classify this effect and it has been briefly explained in Section V how these result can form the basis for a machine learning approach using complementary statistical parameters for the signal, coupled with metrics associated with the spectrum of the signal and other transformations. The purpose of this is to provide an analysis of a binary stream that can specify whether it is truly random, intelligible (i.e. the product of some communicable information, for example) or partially random in some way.

Appendix A
MATLAB Function to compute Basic Statistics of the Relative Entropy Test

The MATLAB function RBET (Relative Binary Entropy Test) given in this Appendix has not been exhaustively tested and has no data error checks. It is provided to give the reader a guide to the basic programming required to implement the computational procedures discussed in Section IV. The code given has been commented but is highly condensed in order to comply with the prescribed page limit for this publication.

```
function [Mean,Std,Median,Mode]=RBET(M);
%INPUTS: (int) M - length of the
%Relative Entropy Signal (RES).
%OUTPUTS: Mean - Mean of the RES.
%Std - Standard Deviation.
%Median - Median of the RES.
%Mode - Mode of the RES.
%Shuffle random number generator.
rng(’shuffle’);
%Read binary string from default file.
fid1=fopen(’binary_string.txt’,’r’);
bstring=fread(fid1); fclose(fid1);
%Compute length of string
L=size(bstring,1);
%and convert binary string to a bit
%array B with elements equal to 0 and 1.
for n=1:L
temp=bstring(n); if temp==48, B(n)=0;
else B(n)=1; end
end %Compute binary histogram
h=hist(B,2);%and evaluate
%probabilities of bits
p(1)=h(1)/L; p(2)=h(2)/L;
%Compute relative entropy signal.
for m=1:M
%Return random bits using function rand,
RB=round(rand(1,L));
%compute binary histogram
h=hist(RB,2);%and evaluate
%probabilities of bits.
q(2)=h(2)/L; q(1)=h(1)/L;
%Compute relative entropy.
RES(m)=-sum(p.*log2(q/p)); end
%Plot the RES in figure 1.
figure(1), plot(RES);
%Compute the 100-bin histogram
h=hist(RES,100);%and plot a bar
%graph of the result in figure 2.
figure(2), bar(h);%Compute the
%Mean, Standard Deviation, Median
%and Mode of the RES.
Mean=mean(RES); Std=std(RES);
Median=median(RES); Mode=mode(RES);
```

Acknowledgment

The authors would like to acknowledge the support of the Science Foundation Ireland, Technological University Dublin, Ireland, University of Western Cape, South Africa, University of KwaZulu-Natal, South Africa and Wrexham Glyndwr University, UK.

References

[1] L. Szilard, “On the Decrease of Entropy in a Thermodynamics System by the Intervention of Intelligent Beings”, *Zeitschrift für Physik*, vol. 53, pp. 840–856, 1922. http://www.sns.ias.edu/~tlusty/courses/InfoInBio/Papers/Szilard1929.pdf

[2] C. E. Shannon, “A Mathematical Theory of Communication”, *The Bell System Technical Journal*, vol. 27, pp. 379–423, 1948. http://math.harvard.edu/~ctm/home/text/others/shannon/entropy/entropy.pdf

[3] , A. N. Kolmogorov, “Entropy per Unit Time as a Metric Invariant of Automorphism”, *Russian Academy of Sciences*, vol. 124, pp. 754-755, 1959.

[4] Y. G. Sinai, “On the Notion of Entropy of a Dynamical System”, *Russian Academy of Sciences*, vol. 124, pp. 768-771, 1959.

[5] G. J. Croll, “BiEntropy – The Approximate Entropy of a Finite Binary String”, 2013. https://pdfs.semanticscholar.org/d8d6/bf07050f9dcd026bfafe62c079b4e309dd7a.pdf

[6] G. Marsaglia, “Random Numbers Fall Mainly in the Planes”, *Proc. Natl. Acad. Sci.* vol. 61, no. 1, pp. 25–28, 1968.

[7] Y. Gao, I. Kontoyiannis, E. Bienenstock, “Estimating the Entropy of Binary Time Series: Methodology, Some Theory and a Simulation Study”, Entropy, vol. 10, pp. 71-99, 2008.

[8] S. Pincus, “Approximate Entropy as a Measure of System Complexity”, *Proc. Natn. Acad. Sci.*, USA, vol 88, pp. 2297-2301, 1991.

[9] S. Pincus and B. H. Singer, “Randomness and Degrees of Irregularity”, *Proc. Natl. Acad. Sci.*, USA, vol. 93, no. 5, pp. 2083–2088, 1996.

[10] A. L. Rukhin, “Approximate Entropy for Testing Randomness”, *J. Appl. Prob.*, vol. 37, no. 1, pp. 88-100, 2000.

[11] A. L. Rukhin, “Testing Randomness: A Suite of Statistical Procedures”, *Theory of Prob. and its Appl.*, vol. 45, no. 1, pp. 111–132, 2000.

[12] J. S. Richman and J. R. Moorman, “Physiological Time-series Analysis using Approximate Entropy and Sample Entropy”, *American J. of Physiology-Heart and Circulatory Physiology*, vol. 278, no. 6, pp. H2039-H2049, 2000.

[13] W. Chen, J. Zhuang, W. Yu and Z. Wang, “Measuring Complexity using FuzzyEn, ApEn, and SampEn”, *Med. Eng. and Phys.*, vol. 31, no. 1, pp. 61-68. 2009.

[14] G. Marsaglia and A. Zaman, “Monkey Tests for Random Number Generators”, *Comp. and Math. with Appl.*, vol. 26, no. 9, pp. 1-10, 1993.

[15] A. N. Kolmogorov, “Three Approaches to the Quantitative Definition of Information”, *Prob. of Info. Trans.*, vol. 1, no. 1, pp. 1-7, 1965.

[16] G. J. Chaitin, “On the Length of Programs for Computing Finite Binary Sequences”, *J. of the ACM*, vol. 13, no. 4, pp. 547-569, 1966.

[17] J. McNair, *The Binary Derivative: A New Method of Testing the Appearance of Randomness in a Sequence of Bits*, MSc Thesis, University of Western Ontario, 1989.

[18] J. M. Carroll, “The Binary Derivative Test: Noise Filter, Crypto Aid and Random Number Seed Selector”, *Simulation*, vol. 53, pp. 129-135, 1989.

[19] J. M. Carroll and Y. Sun, *The Binary Derivative Test for the Appearance of Randomness and its use as a Noise Filter*, Technical Report No. 221, Department of Computer Science, University of Western Ontario, 1998.

[20] C. S. Bruwer, *Correlation Attacks on Stream Ciphers using Convolutional Codes*, Masters Thesis, University of Pretoria, 2005.

[21] Text to Binary translator, *RapidTables*, 2019. https://www.rapidtables.com/convert/number/ascii-to-binary.html

[22] Google Translate, 2019. https://translate.google.co.uk/