On Empirical Entropy

Paul M.B. Vitányi

Abstract

We propose a compression-based version of the empirical entropy of a finite string over a finite alphabet. Whereas previously one considers the naked entropy of (possibly higher order) Markov processes, we consider the sum of the description of the random variable involved plus the entropy it induces. We assume only that the distribution involved is computable. To test the new notion we compare the Normalized Information Distance (the similarity metric) with a related measure based on Mutual Information in Shannon's framework. This way the similarities and differences of the last two concepts are exposed.

Index Terms— Empirical entropy, Kolmogorov complexity, normalized information distance, similarity metric, mutual information distance

I. Introduction

In the basic set-up of Shannon [20] a message is a finite string over a finite alphabet. One is interested in the expected number of bits to transmit a message from a sender to a receiver, when both the sender and the receiver consider the same ensemble of messages (the set of possible messages provided with a probability for each message). The expected number of bits is known as the entropy of the ensemble of messages. This ensemble is also known as the source.

The empirical entropy of a single message is taken to be the entropy of a source that produced it as a typical element. (The notion of "typicality" is defined differently by different authors and we take here the intuitive meaning.) Traditionally, this source is a (possibly higher order) Markov process. This leads to the definition in Example 2.4. Here we want to liberate the notion so that it encompasses all computable random variables with finitely many outcomes consisting of finite strings over a finite alphabet. Moreover,

Paul Vitányi is a CWI Fellow with the National Research Center for Mathematics and Computer Science in the Netherlands (CWI) and Emeritus Professor of Computer Science at the University of Amsterdam. He was supported in part by the the ESF QiT Programmme, the EU NoE PASCAL II. Address: CWI, Science Park 123, 1098 XG Amsterdam, The Netherlands. Email: Paul.Vitanyi@cwi.nl.

since we are given only a single message, but not the ensemble from which it is an element, the new empirical entropy should provide both this ensemble and the entropy it induces. If we are given just the entropy but not the ensemble involved, then a receiver cannot in general reconstruct the message. Moreover, we are given a single message which has a particular length, say n. Therefore, given the family of random variables we draw upon, we can select one of them and compute the probability of every message of length n. For fixed n, this results in a Bernoulli variable that has $|\Sigma|^n$ outcomes.

We are thus led to a notion of empirical entropy that consists of a description of the Bernoulli variable involved plus the related entropy of the message induced. Since we assume the original probability mass function to be computable, the Bernoulli variable is computable and its effective description length can be expressed by its Kolmogorov complexity.

Normalized Information Distance (explained below) between two finite objects is often confused with a similar distance between two random variables. The last distance is expressed in terms of probabilistic mutual information. We use our new notion to explain the differences between the former distance between two individual objects and the latter distance between two random variables. This difference parallels that between the Kolmogorov complexity of a single finite object and the entropy of a random variable. The former quantifies the information in a finite object, while the latter gives us the expected number of bits to communicate any outcome of a random variable known to both the sender and the receiver. Computability notions are reviewed in Appendix A, and Kolmogorov complexity in Appendix B.

A. preliminaries

We write *string* to mean a finite string over a finite alphabet Σ. Other finite objects can be encoded into strings in natural ways. The set of strings is denoted by Σ^*. We usually take $\Sigma = \{0, 1\}$. The *length* of a string x is the number of letters in Σ in it denoted as $|x|$. The *empty* string ϵ has length $|\epsilon| = 0$. Identify the natural numbers $\mathcal{N}$ (including 0) and $\{0, 1\}^*$ according to the correspondence

$$(0, \epsilon), (1, 0), (2, 1), (3, 00), (4, 01), \ldots . \tag{I.1}$$

Then, $|010| = 3$. The emphasis here is on binary sequences only for convenience; observations in every finite alphabet can be so encoded in a way that is 'theory neutral.' For example, if a finite alphabet Σ has cardinality 2^k, then every element $i \in \Sigma$ can be encoded by $\sigma(i)$ which is a block of bits of length k. With this encoding every $x \in \Sigma^*$ satisfies that the Kolmogorov complexity $K(x) = K(\sigma(x))$ (see Appendix B for basic definitions and results on Kolmogorov complexity) up to an additive constant that

is independent of x.

II. The New Empirical Entropy

Let X be a random variable with outcomes in a finite alphabet $\mathbf{X}$. Shannon's entropy [20] is

$$H(X) = \sum_{x \in \mathcal{X}} P(X = x) \log 1/P(X = x).$$

There are three items involved in the new empirical entropy of data x:

- A class of random variables like the set of Bernoulli processes, or the set of higher order Markov processes; from each element of this class we construct a Bernoulli variable X with $|\Sigma|^n$ outcomes of length n;
- a selection of a random variable from this Bernoulli class such that x is a typical outcome, and
- a description of this random variable plus its entropy.

This is reminiscent of universal coding essentially due to Kolmogorov [11], and of two-part MDL due to Rissanen [19]. In its simplest form the former, assuming a Bernoulli process, codes a string x of length n over a finite alphabet Σ as follows: A string containing a description of $n, |\Sigma|$ and n/n_i ($1 \leq i \leq |\Sigma|$), and the index of x in the set constrained by these items. The coding should be such that the individual substrings can be parsed, except the description of the index which we put last. This takes additive terms that are logarithmic in the length of the items except the last one. The universal code takes $O(|\Sigma| \log n) + \binom{n}{n/n_1 \cdots n/n_{|\Sigma|}}$ bits. The two-part MDL complexity of a string [19], is the minimum of the self-information of that string with respect to a source and the number of bits needed to represent that source. The source is not required to be Markovian and the two-part MDL takes into account its complexity. However, the methods of encoding are arbitrary.

An n-length outcome $x = x_1, x_2, \ldots, x_n$ over Σ is the outcome of a stochastic process $X_1, X_2, \ldots, X_n$ characterized by a joint probability mass function $\Pr(\{X_1, X_2, \ldots, X_n) = (x_1, x_2, \ldots, x_n)\}$. For technical reasons we replace the list $X_1, X_2, \ldots, X_n$ by a single Bernoulli random variable X with outcomes in $\mathbf{X} = \Sigma^n$. Here, the random variables X_i may be independent copies of a single random variable as is the case wen the source stochastic process is a Bernoulli variable. But the source stochastic process may be a higher order Markov chain making some or all X_is dependent (this depends on whether the order of the Markov chain is greater then n). For certain stochastic processes all X_is are dependent for every n: the stochastic process assigns a probability to every outcome in Σ^*.

Definition 2.1: Let n be an integer, Σ a finite alphabet, $x \in \Sigma^n$ be a string, $\mathcal{X}$ a family of computable processes, each process $\Xi \in \mathcal{X}$ producing (possibly by repetition) a sequence of (possibly dependent) random variables $X = X_1, X_2, \ldots, X_n$, with $\Pr(X = x)$ is computable and $H(X) < \infty$. The *empirical entropy* of x with respect to $\mathcal{X}$ is given by

$$H(x|\mathcal{X}) = \min_{\Xi \in \mathcal{X}} \{K(X) + H(X) : |H(X) - \log 1/\Pr(X = x)| \text{ is minimal}\}.$$

This means that the expected binary length of encoding an outcome of X is as close as possible to $\log 1/\Pr(X = x)$. In the two-part description the complexity part describes X, and the entropy part is the ignorance about the data x in the set Σ^n given X.

Remark 2.2: By assumption n is fixed. By Theorem 3 in [20], i.e. the asymptotic equidistribution property, for ergodic Markov sources the following is the case. Let H be the per symbol entropy of the source. For example, if the source Ξ is Bernoulli with $\Pr(\Xi = s_i) = p(s_i)$ ($s_i \in \Sigma$ for $1 \leq i \leq |\Sigma|$), then $H = \sum_{i=1}^{|\Sigma|} p(s_1) \log 1/p(s_i)$. Let X be the induced Bernoulli variable with $|\Sigma|^n$ outcomes consisting of sequences of length n over Σ. Then, for every $\epsilon, \delta > 0$ there is an n_0 such that the sequences of length $n \geq n_0$ are divided into two classes: one set with total probability less than ϵ and one set such that for every y in this set holds $|H - \frac{1}{n} \log 1/\Pr(X = y)| < \delta$. Note that $H(X) = nH$. Thus, for large enough n we are almost certain to have $|H(X) - \log 1/\Pr(X = x)| = o(n)$.

Set $\epsilon = \delta$ for convenience. We call the set of y's such that $|H(X) - \log 1/\Pr(X = y)| = \epsilon n$, with $\epsilon > 0$ and some n_0 depending on ϵ and $n \geq n_0$, the *ϵ-typical* outcomes of X. The cardinality of the set $S \subseteq \Sigma^n$ of such y's satisfies

$$(1 - \epsilon)|\Sigma|^{H(X) - \epsilon n} \leq |S| \leq |\Sigma|^{H(X) + \epsilon n}.$$

See [7] Theorem 3.1.2. ♢

Lemma 2.3: Assume Definition 2.1. Then, $K(X) \leq K(x, \mathcal{X}) + O(1)$.

Proof: The family $\mathcal{X}$ consists of computable random variables, that is, in essence of computable probability mass functions. The family of all lower semicomputable semiprobability mass functions can be effectively enumerated, possibly with repetitions, Theorem 4.3.1 in [17]. The latter family contains all computable probability mass functions, hence it contains X. Thus, if we know $x, \mathcal{X}$ we can compute the $X \in \mathcal{X}$ of Definition 2.1 by going through this list. ■

Example 2.4: Assume Definition 2.1. Let n_i be the number of occurrences of the ith character of Σ in x. If w is a string then x_w is the string obtained by concatenating the characters immediately following

occurrences of w in x. The cardinality $|x_w|$ is the number of occurrences of w in x unless w occurs as a suffix of x in which case it is 1 less. In [12], [18], [8] the kth order empirical entropy of x is defined by

$$H_k(x) = \begin{cases} \frac{1}{n}\sum_{i=1}^{|\Sigma|} n_i \log \frac{n}{n_i} & \text{for } k = 0, \\ \frac{1}{n}\sum_{|w|=k} |x_w| H_0(x_w) & \text{for } k > 0. \end{cases} \tag{II.1}$$

The kth order empirical entropy of x can be reconstructed from x once we know k. The kth order empirical entropy of x results from the probability induced by a kth order Markov source $\Xi \in \mathcal{X}$. (A Bernoulli process is a 0th order Markov source.)

Let $\mathcal{X}$ to be the family of kth order Markov sources (a specific $k \geq 0$), provided the transition probabilities are computable. Such a family is subsumed under Definition 2.1. Let x be a string over Σ which is typically produced by such a Markov source of order k. The empirical entropy $H(x|\mathcal{X})$ of x is $K(X) + nH_k(x)$. Here X is the random variable associated with the kth order empirical entropy computed from x. Note that the empirical entropy $H_k(x)$ stops being a reasonable complexity metric for almost all strings roughly when $|\Sigma|^k$ surpasses n, [8]. $\diamondsuit$

Example 2.5: Let $x = (10)^{n/2}$ for even n (that is, $n/2$ copies of the pattern "10"). Let $\mathcal{X}_1$ be the family of binary Bernoulli processes. The empirical entropy $H(x|\mathcal{X}_1)$ is reached for i.i.d. sequence $X = X_1, X_2, \ldots, X_n \in \mathcal{X}_1$, each X_i being a copy of the same random variable Y with outcomes in $\{0, 1\}$ with $P(Y = 1) = \frac{1}{2}$. Then, $H(x|\mathcal{X}_1) = K(X) + nH(Y)$. Then X can be computed from the information concerning n in $O(\log n)$ bits, the particular $\Xi \in \mathcal{X}$ used in $O(1)$ bits, and a program of $O(1)$ bits to compute X from this information. In this way $K(X) = O(\log n)$. Moreover, $H(Y) = 1$, so that $H(x|\mathcal{X}_1) = n + O(\log n)$.

Let $\mathcal{X}_2$ be the family of first order Markov processes with 2 transitions each and with output alphabet $\{0, 1\}$ for each state. The empirical entropy $H(x|\mathcal{X}_2)$ is reached for the n-bit output of a deterministic "parity" Markov process. That is, $X = X_1, X_2, \ldots, X_n$ and every X_i gives the output at time i of the Markov process with 2 states s_0 and s_1 defined as follows. The transit probabilities are $p(s_0 \to s_1) = 1$ and $p(s_1 \to s_0) = 1$, while the output in state s_0 is 0 and in state s_1 is 1. The start state is s_0. In this way, $P(X = (10)^{n/2}) = 1$ while $H(X) = 0$. Then, $H(x|\mathcal{X}_2) = K(X) + H(X)$. Here $K(X) = O(\log n)$, since we require a description of n, the 2-state Merkov process involved, and a program to compute X from this information. Since the outcome is deterministic, $H(X) = 0$, so that $H(x|\mathcal{X}_2) = O(\log n)$. $\diamondsuit$

Example 2.6: Consider the first n bits of $\pi = 3.1415\ldots$. Let $\mathcal{X}_1$ be the family of Bernoulli processes. Empirically, it has been established that the frequency of 1's in the binary expansion of π is $n/2 \pm O(\sqrt{n})$,

that is, the binary expaqnsion of π is a typical pseudorandom sequence. Hence, $H(x|\mathcal{X}_1) = K(X) + nH(X)$ where $X = X_1, X_2, \ldots, X_n \in \mathcal{X}_1$ and the X_i's are n i.i.d. distributed copies of Y. Here Y is a Bernoulli process with $P(Y = 1) = \frac{1}{2}$. Then $K(X) = O(\log n)$ and $H(Y) = 1$, so that $H(x|\mathcal{X}_1) = n + O(\log n)$.

Let $\mathcal{X}_2$ be the family of computable random variables with as outcomes binary strings of length n. We know that there is a small program, say of about $10,000$ bits, incorporating an approximation algorithm that generates the successive bits of π forever. Telling it to stop after n bits, we can generate the computable Bernoulli variable $X \in \mathcal{X}_2$ assigning probability 1 to x and probability 0 to any other binary string of length n. Assume $n = 1,000,000,000$. Then, we have $K(X) \leq \log 1,000,000,000 + c \approx 30 + c$ where the c additive term is the number of bits of the program to compute π and a program required to turn the logarithmic description of $1,000,000,000$ and the program to compute π into the random variable X. Finally, $H(X) = 0$. Therefore, $H(x|\mathcal{X}_2) \leq 10,030 + c$. $\diamondsuit$

Example 2.7: Consider printed English, say just lower case and space signs, ignoring the other signs. The entropy of representative examples of printed English has been estimated experimentally by Shannon [21] based on human subjects guesses of successive characters in a text. His estimate is between 0.6 and 1.3 bits per character (bpc), and [22] obtained an estimate of 1.46 bpc for PPM based models, which we will use in this example. PPM (prediction by partial matching) is an adaptive statistical data compression technique. It is based on context modeling and prediction and uses a set of previous symbols in the uncompressed symbol stream to predict the next symbol in the stream, rather like a mechanical version of Shannon's method. Consider a text of n characters over the alphabet used by [22], and let $\mathcal{X}$ be the class of PPM based models with n output characters over the used alphabet. Since the PPM machine can be described in $O(1)$ bits (its program is finite) and the length n in $O(\log n)$ bits, we have $K(X) = O(\log n)$. Hence, $H(x|\mathcal{X}) \leq K(X) + 1.46n = 1.46n + O(\log n)$. $\diamondsuit$

In these examples we see that the empirical entropy is higher when the family of random variables considered is simpler. For simple random variables the knowledge in the Kolmogorov complexity part is neglible. The empirical entropy with respect to a complex family of random variables can be lower than that with respect to a family of simple random variables by transforming the ignorance in the entropy part into knowledge in the Kolmogorov complexity part. We use this observation to consider the widest family of computable probability mass functions.

Lemma 2.8: Let $\mathcal{X}$ be the family of computable random variables X with $H(X) < \infty$, and $x \in \Sigma^*$ with $|\Sigma| < \infty$. Then, $H(x|\mathcal{X}) = K(x) + O(1)$.

Proof: First, let p_x be a shortest prefix program which computes x. Hence $|p_x| = K(x)$. By adding $O(1)$ bits to it we have a program p_p which computes a probability mass function p with $p(x) = 1$ and $p(y) = 0$ for $y \neq x$ ($x, y \in \Sigma^*$). Hence $|p_p| \leq K(x) + O(1)$.

Second, let q_p be a shortest prefix program which computes a probability mass function p with $p(x) = 1$ and $p(y) = 0$ for $y \neq x$ ($x, y \in \Sigma^*$). Thus, $|q_p| \leq |p_p|$. Adding $O(1)$ bits to q_p we have a program q_x which computes x. Then, $K(x) \leq |q_p| + O(1)$.

Altogether, $|q_p| = K(x) + O(1)$. ■

For the sequel of this paper, we need to extend the notion of empirical entropy to joint probability mass functions.

Definition 2.9: Let n be an integer, Σ a finite alphabet, $x, y \in \Sigma^n$ be strings, $\mathcal{Z}$ be the family of computable joint probability mass functions, $Z \in \mathcal{Z}$ and (x, y) an outcome of Z. Let the probability mass function $p(x, y) = P(Z = (x, y))$ have a finite joint entropy $H(Z) < \infty$. The *empirical entropy* of (x, y) with respect to $\mathcal{Z}$ is

$$H(x, y|\mathcal{Z}) = \min_{Z \in \mathcal{Z}} \{K(Z) + H(Z) : |H(Z) - \log 1/p(x, y)| \ \text{ is } \ \text{minimal}\}.$$

Lemma 2.10: Let $\mathcal{Z}$ be the family of computable joint probability mass functions Z with $H(Z) < \infty$, and $x, y \in \Sigma^*$ with $|\Sigma| < \infty$. Then, $H(x, y|\mathcal{Z}) = K(x, y) + O(1)$.

Proof: Similar to that of Lemma 2.8. ■

III. Normalized Information Distance

The classical notion of Kolmogorov complexity [11] is an objective measure for the information in a *single* object, and information distance measures the information between a *pair* of objects [2]. This last notion has spawned research in the theoretical direction, see the many Google Scholar citations to the above reference. Research in the practical direction has focused on the normalized information distance (NID), also called "the similarity metric," which arises by normalizing the information distance in a proper manner. (The NID is defined by (III.2) below.)

If we approximate the Kolmogorov complexity through real-world compressors [16], [6], [4], then we obtain the normalized compression distance (NCD) from the NID. This is a parameter-free, feature-free, and alignment-free similarity measure that has had great impact in applications. (Only the compressor used can be viewed as a parameter or feature.) The NCD was preceded by a related nonoptimal distance [15]. In [10] another variant of the NCD has been tested on all major time-sequence databases used in

all major data-mining conferences against all other major methods used. The compression method turned out to be competitive in general and superior in heterogeneous data clustering and anomaly detection.

There have been many applications in pattern recognition, phylogeny, clustering, and classification, ranging from hurricane forecasting and music to to genomics and analysis of network traffic, see the many papers referencing [16], [6], [4] in Google Scholar. In [16] it is shown that the NID, and in [4] that the NCD subject to mild conditions on the used compressor, are metrics up to negligible discrepancies in the metric (in)equalities and that they are always between 0 and 1. The computability status of the NID has been resolved in [23]. The NCD is computable by definition.

The *information distance* $D(x,y)$ between strings x and y is defined as

$$D(x,y) = \min_p\{|p| : U(p,x) = y \wedge U(p,y) = x\},$$

where U is the reference universal Turing machine above. Like the Kolmogorov complexity K, the distance function D is upper semicomputable. Define

$$E(x,y) = \max\{K(x|y), K(y|x)\}.$$

In [2] it is shown that the function E is upper semicomputable, $D(x,y) = E(x,y) + O(\log E(x,y))$, the function E is a metric (more precisely, that it satisfies the metric (in)equalities up to a constant), and that E is minimal (up to a constant) among all upper semicomputable distance functions D' satisfying the mild normalization conditions $\sum_{y:y\neq x} 2^{-D'(x,y)} \leq 1$ and $\sum_{x:x\neq y} 2^{-D'(x,y)} \leq 1$. (Here and elsewhere in this paper "log" denotes the binary logarithm.) The *normalized information distance* (NID) e is defined by

$$e(x,y) = \frac{E(x,y)}{\max\{K(x), K(y)\}}. \tag{III.1}$$

It is straightforward that $0 \leq e(x,y) \leq 1$ up to some minor discrepancies for all $x, y \in \{0,1\}^*$. Rewriting e using (A.1) yields

$$e(x,y) = \frac{K(x,y) - \min\{K(x), K(y)\}}{\max\{K(x), K(y)\}}, \tag{III.2}$$

up to some lower order terms that we ignore.

Lemma 3.1: Let x be a string, $\mathcal{X}$, $\mathcal{Z}$ be the families of random variables with computable probability mass functions and computable joint probability mass functions, respectively. Moreover, for $X \in \mathcal{X}$ and $Z \in \mathcal{Z}$ we have $H(X), H(Z) < \infty$. Then, we can substitute the Kolmogorov complexities in (III.2) by the corresponding empirical entropies as in (III.3).

Proof: By Lemma's 2.8 and 2.10 we know the following. For $\mathcal{X}$ is the family of computable probability mass functions, $H(x|\mathcal{X}) = K(x)$, $H(y|\mathcal{X}) = K(y)$. For $\mathcal{Z}$ is the family of computable joint probability mass functions, $H(x, y|\mathcal{Z}) = K(x, y)$. Hence,

$$e(x, y) = \frac{H(x, y|\mathcal{Z}) - \min\{H(x|\mathcal{X}), H(y|\mathcal{X})\}}{\max\{H(x|\mathcal{X}), H(y|\mathcal{X})\}}, \tag{III.3}$$

ignoring lower order terms. ■

Remark 3.2: In Lemma 3.1 we can replace the computable random variables by the restriction to computable random variables that have a singleton support, that is, probability mass functions p with $p(x) = 1$ for some x and $p(y) = 0$ for all $y \neq x$. Alternatively, we can replace it by the family of computable Markov processes. To see this, for every x of length n there is a computable Markov process M of order $n - 1$ that outputs x deterministically and $K(x) = K(M) + O(1)$.

Clearly, if we replace the family of computable probability mass functions in the empirical entropies in Lemma 3.1 by weaker subfamilies like the families based on computable Bernoulli functions, computable Gaussians, or computable first order Markov processes, then Lemma 3.1 will not hold in general. ♢

Remark 3.3: The NCD is defined by

$$NCD_Z(x, y) = \frac{|Z(xy)| - \min\{|Z(x)|, |Z(y)|\}}{\max\{|Z(x)|, |Z(y)|\}}, \tag{III.4}$$

where $Z(x)$ is the compressed version of x when it is compressed by a lossless compressor Z. We have substituted xy for the pair (x, y) both for convenience and with ignorable consequences. Consider a simple compressor that uses only Bernoulli variables, for example a Huffman code compressor. The compressed version of a string is preceded by a header containing information identifying the compressor and the charcteristics used (the relative frequencies in this case) to compress the source string. In general this is the case with every compressor. (In [3] the NCD based on compressors computing the static Huffman code of a Bernoulli variable is shown to be the total Kullback-Leibler divergence to the mean. We refrain from explaining these terms since are extraneous to our treatment.)

Thus, $Z(x)$ is comprised of the header generated by Z for x. This header makes it possible to use the uncompress feature, denoted here by Z^{-1} so that $Z^{-1}Z(x) = x$. The header describes a random variable Ξ based on the compressor Z. The family of random variables induced by the compressor Z can be denoted by $\mathcal{X}_Z$.

In this way, we can define the Bernoulli variable X used to compress x. The empirical entropy $H(x|\mathcal{X}_Z) = K(X) + H(X)$. Here $K(X)$ is uncomputable. We approximate it by the length of the header,

say $|\alpha(X)|$. The Bernoulli variable X has entropy $H(X)$ and $|Z(x)| = |\alpha(X)| + H(X)$. Similarly for y and (x, y). Therefore,

$$NCD_Z(x,y) = \frac{|\alpha(XY)| + H(X,Y) - \min\{|\alpha(X)| + H(X), |\alpha(Y)| + H(Y)\}}{\max\{|\alpha(X)| + H(X), |\alpha(Y)| + H(Y)\}}, \tag{III.5}$$

ignoring lower order terms, where $|\alpha(X)| \geq K(X)$, $|\alpha(Y)| \geq K(Y)$, and $|\alpha(XY)| \geq K(XY)$.

♢

IV. Mutual Information

In [25], [1], [13], [9], [26], [14] the entropy and joint entropy of a pair of sequences is determined, and this is directly equated with the Kolmogorov complexity of those sequences. The Shannon type probabilistic version of (III.2) is

$$\begin{aligned} e_H(X,Y) &= \frac{H(X,Y) - \min\{H(X), H(Y)\}}{\max\{H(X), H(Y)\}} \\ &= 1 - \frac{\max\{H(X), H(Y)\} - H(X,Y) + \min\{H(X), H(Y)\}}{\max\{H(X), H(Y)\}} \\ &= 1 - \frac{I(X;Y)}{\max\{H(X), H(Y)\}}, \end{aligned}$$

since the *mutual information* $I(X;Y)$ between random variables X and Y is

$$I(X;Y) = H(X) + H(Y) - H(X,Y),$$

and

$$\max\{H(X), H(Y)\} + \min\{H(X), H(Y)\} = H(X) + H(Y).$$

In this way, $e_H(X,Y)$ is 1 minus the mutual information between random variables X and Y per bit of the maximal entropy. How do the cited references connect this distance between two random variables to (III.2), the distance between two individual outcomes x and y?

Ostensibly one has to replace the entropy of random variables X and Y by the empirical entropy according to Definition 2.1 deduced from strings x and y. To obtain the required result (III.2) one has to use families $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}$ of computable random variables such that $K(x) = H(x|\mathcal{X})$, $K(y) = H(y|\mathcal{Y})$, and $K(x,y) = H(x,y|\mathcal{Z})$. In our framework this is possible only if $\mathcal{X}, \mathcal{Y}$ are appropriate families of computable random variables, and $\mathcal{Z}$ is an appropriate family of computable joint random variables. Outside our framework the widest notion of empirical entropy is (II.1) and there it is not possible at all.

To obtain computable approximations using a real-world compressor Z for x and y as in (III.4) we can take the empirical entropy based on compressor Z as in (III.4) and (III.5).

APPENDIX

A. Computability

In 1936 A.M. Turing [24] defined the hypothetical 'Turing machine' whose computations are intended to give an operational and formal definition of the intuitive notion of computability in the discrete domain. These Turing machines compute integer functions, the *computable* functions. By using pairs of integers for the arguments and values we can extend computable functions to functions with rational arguments and/or values. The notion of computability can be further extended, see for example [17]: A function f with rational arguments and real values is *upper semicomputable* if there is a computable function $\phi(x,k)$ with x an rational number and k a nonnegative integer such that $\phi(x,k+1) \leq \phi(x,k)$ for every k and $\lim_{k\to\infty} \phi(x,k) = f(x)$. This means that f can be computably approximated from above. A function f is A function f is *lower semicomputable* if $-f$ is upper semicomputable. A function is called *semicomputable* if it is either upper semicomputable or lower semicomputable or both. If a function f is both upper semicomputable and lower semicomputable, then f is *computable*. A countable set S is *computably (or recursively) enumerable* if there is a Turing machine T that outputs all and only the elements of S in some order and does not halt. A countable set S is *decidable (or recursive)* if there is a Turing machine T that decides for every candidate a whether $a \in S$ and halts.

Example A.1: An example of a computable function is $f(n)$ defined as the nth prime number; an example of a function that is upper semicomputable but not computable is the Kolmogorov complexity function K in Appendix B. An example of a recursive set is the set of prime numbers; an example of a recursively enumerable set that is not recursive is $\{x \in \mathcal{N} : K(x) < |x|\}$. $\diamondsuit$

B. Kolmogorov Complexity

Informally, the Kolmogorov complexity of a string is the length of the shortest string from which the original string can be losslessly reconstructed by an effective general-purpose computer such as a particular universal Turing machine U, [11] or the text [17]. Hence it constitutes a lower bound on how far a lossless compression program can compress. In this paper we require that the set of programs of U is prefix free (no program is a proper prefix of another program), that is, we deal with the *prefix Kolmogorov complexity*. (But for the results in this paper it does not matter whether we use the plain

Kolmogorov complexity or the prefix Kolmogorov complexity.) We call U the *reference universal Turing machine*. Formally, the *conditional prefix Kolmogorov complexity* $K(x|y)$ is the length of the shortest input z such that the reference universal Turing machine U on input z with auxiliary information y outputs x. The *unconditional prefix Kolmogorov complexity* $K(x)$ is defined by $K(x|\epsilon)$. The functions $K(\cdot)$ and $K(\cdot \mid \cdot)$, though defined in terms of a particular machine model, are machine-independent up to an additive constant and acquire an asymptotically universal and absolute character through Church's thesis, see for example [17], and from the ability of universal machines to simulate one another and execute any effective process. The Kolmogorov complexity of an individual finite object was introduced by Kolmogorov [11] as an absolute and objective quantification of the amount of information in it. The information theory of Shannon [20], on the other hand, deals with *average* information *to communicate* objects produced by a *random source*. Since the former theory is much more precise, it is surprising that analogs of theorems in information theory hold for Kolmogorov complexity, be it in somewhat weaker form. For example, let X and Y be random variables with a joint distribution. Then, $H(X,Y) \leq H(X) + H(Y)$, where $H(X)$ is the entropy of the marginal distribution of X. Similarly, let $K(x,y)$ denote $K(\langle x,y\rangle)$ where $\langle\cdot,\cdot\rangle$ is a standard pairing function and x, y are binary strings. An example is $\langle x,y\rangle$ defined by $y + (x+y+1)(x+y)/2$ where x and y are viewed as natural numbers as in (I.1). Then we have $K(x,y) \leq K(x) + K(y) + O(1)$. Indeed, there is a Turing machine T_i that provided with $\langle p,q\rangle$ as an input computes $\langle U(p), U(q)\rangle$ (where U is the reference Turing machine). By construction of T_i, we have $K_i(x,y) \leq K(x) + K(y)$, hence $K(x,y) \leq K(x) + K(y) + O(1)$.

Another interesting similarity is the following: $I(X;Y) = H(Y) - H(Y \mid X)$ is the (probabilistic) *information in random variable X about random variable Y*. Here $H(Y \mid X)$ is the conditional entropy of Y given X. Since $I(X;Y) = I(Y;X)$ we call this symmetric quantity the *mutual (probabilistic) information*.

Definition A.2: The *(algorithmic) information in x about y* is $I(x:y) = K(y) - K(y \mid x)$, where x, y are finite objects like finite strings or finite sets of finite strings.

It is remarkable that also the algorithmic information in one finite object about another one is symmetric: $I(x:y) = I(y:x)$ up to an additive term logarithmic in $K(x) + K(y)$. This follows immediately from the *symmetry of information* property due to A.N. Kolmogorov and L.A. Levin (they proved it for plain

Kolmogorov complexity but in this form it holds equally for prefix Kolmogorov complexity):

$$K(x,y) = K(x) + K(y \mid x) + O(\log(K(x) + K(y))) \tag{A.1}$$
$$= K(y) + K(x \mid y) + O(\log(K(x) + K(y))).$$

REFERENCES

[1] D. Benedetto E. Caglioti and V. Loreto, "Language Trees and Zipping," ¡i¿Physical Rev. Letters,¡/i¿ vol. 88, 2002, p. 048702.

[2] C.H. Bennett, P. Gács, M. Li, P.M.B. Vitányi, and W. Zurek. Information Distance, *IEEE Transactions on Information Theory*, 44:4(1998), 1407–1423.

[3] R.L. Cilibrasi, *Statistical Inference Through Data Compression*, Ph.D. Thesis, University of Amsterdam, Amsterdam, The Netherlands, 2007.

[4] R.L. Cilibrasi, P.M.B. Vitányi, Clustering by compression, *IEEE Trans. Information Theory*, 51:4(2005), 1523- 1545.

[5] R.L. Cilibrasi, P.M.B. Vitányi, The Google Similarity Distance, *IEEE Trans. Knowledge and Data Engineering*, 19:3(2007), 370-383.

[6] R. Cilibrasi, P.M.B. Vitányi, R. de Wolf, Algorithmic clustering of music based on string compression, *Computer Music J.*, 28:4(2004), 49-67.

[7] T.M. Cover and J.A. Thomas, *Elements of Information Theory*, Wiley, New York, 1991.

[8] T. Gagie, Large alphabets and incompressibility, *Information Processing Letters*, 99(2006) 246251.

[9] Z. Dawy, J. Hagenauer, P. Hanus, J.C. Mueller, Mutual information based distance measures for classification and content recognition with applications to genetics, *Proc. IEEE Int. Conf. Communications*, Vol. 2, 2005, 820–824.

[10] E. Keogh, S. Lonardi, C.A. Ratanamahatana, L. Wei, S.-H. Lee, J. H., Compression-based data mining of sequential data, *Data Mining and Knowledge Discovery*, 14:1(2007), 99–129

[11] A.N. Kolmogorov, Three approaches to the quantitative definition of information, *Problems Inform. Transmission* 1:1 (1965) 1–7.

[12] S.R. Kosaraju and G. Manzini. Compression of low entropy strings with Lempel-Ziv algorithms. *SIAM J. Comput.*, 29(1999), 893911.

[13] A. Kraskov, H. Stogbauer, R.G. Andrzejak, P. Grassberger, Hierarchical clustering using mutual information, *Europhysics Letters*, 70:2(2005), 278–284.

[14] A. Kraskov, P. Grassberger, pp. 101–124 in: MIC: Mutual information based hierarchical clustering, *Information Theory and Statistical Learning*, F. Emmert-Streib, M. Dehmer, Eds., Springer, New York, 2009.

[15] M. Li, J.H. Badger, X. Chen, S. Kwong, P. Kearney, and H. Zhang. An information-based sequence distance and its application to whole mitochondrial genome phylogeny, *Bioinformatics*, 17:2(2001), 149–154.

[16] M. Li, X. Chen, X. Li, B. Ma, P.M.B. Vitányi. The similarity metric, *IEEE Trans. Inform. Th.*, 50:12(2004), 3250- 3264.

[17] M. Li and P.M.B. Vitányi. *An Introduction to Kolmogorov Complexity and its Applications*, Springer-Verlag, New York, 3rd Edition, 2008.

[18] G. Manzini, An analysis of the BurrowsWheeler Transform, *J. ACM*, 48(2001), 407430.

[19] J.J. Rissanen. *Stochastical Complexity and Statistical Inquiry*, World Scientific, 1989.

[20] C.E. Shannon, A Mathematical Theory of Communication, *Bell Syst. Tech. J.*, 27(1948), 379–423, 623–656.

[21] C.E. Shannon, Prediction and entropy of printed English, *Bell Syst. Tech. J.*, 30(1951), 50-64.

[22] W.J. Teahan, J.G. Cleary, The Entropy of English Using PPM-based Models, in: Proc. Data Compression Conf., 1996, 53-62.

[23] S.A. Terwijn, L. Torenvliet, P.M.B. Vitányi, Nonapproximability of the Normalized Information Distance, *J. Comput. System Sciences*, To appear.

[24] A.M. Turing, On computable numbers, with an application to the Entscheidungsproblem, *Proc. London Mathematical Society*, 42:2(1936), 230-265, ”Correction”, 43i(1937), 544-546.

[25] Z.G. Yu, P. Jiang, Distance, correlation and mutual information among portraits of organisms based on complete genomes, *Phys. Lett. A*, 286(2001), 34-46.

[26] Z.G. Yu, Z. Mao, L.-Q. Zhou, Vo.V. Anh, A mutual information based sequence distance for vertebrate phylogeny using complete mitochondrial genomes, Proc. IEEE 3rd Int. Conf. Natural Computation, 2007, 253–257.