May 1994

A Mathematical Formalism of Infinite Coding for the Compression of Stochastic Process

Kevin Atteson
University of Pennsylvania

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-94-27.

A Mathematical Formalism of Infinite Coding for the Compression of Stochastic Process

Abstract

As mentioned in [5, page 6], there are two basic models for sources of data in information theory: finite length sources, that is, sources which produce finite length strings, and infinite length sources, which produce infinite length strings. Finite length sources provide a better model for files, for instance, since files consist of finite length strings of symbols. Infinite length sources provide a better model for communication lines which provide a string of symbols which, if not infinite, typically have no readily apparent end. In fact, even in some cases in which the data is finite, it is convenient to use the infinite length source model. For instance, the widely used adaptive coding techniques (see, for instance [5]) typically use arithmetic coding which implicitly assumes an infinite length source (although practical implementations make modifications so that it may be used with finite length strings). In this paper, we formalize the notion of encoding an infinite length source. While such infinite codes are used intuitively throughout the literature, their mathematical formalization reveals certain subtleties which might otherwise be overlooked. For instance, it turns out that the pure arithmetic code for certain sources has not only unbounded but infinite delay (that is, it is necessary to see a complete infinite source string before being able to determine even one bit of the encoded string in certain cases). Fortunately, such cases occur with zero probability. The formalization presented here leads to a better understanding of infinite coding and a methodology for designing better infinite codes for adaptive data compression (see [1]).

Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-94-27.

A Mathematical Formalism of Infinite Coding for the Compression of Stochastic Processes

MS-CIS-94-27
GRASP LAB 374

Kevin Atteson

University of Pennsylvania
School of Engineering and Applied Science
Computer and Information Science Department
Philadelphia, PA 19104-6389

May 1994

A Mathematical Formalism of Infinite Coding for the Compression of Stochastic Processes

Kevin Atteson

May 25, 1994

As mentioned in [5, page 6], there are two basic models for sources of data in information theory: finite length sources, that is, sources which produce finite length strings, and infinite length sources, which produce infinite length strings. Finite length sources provide a better model for files, for instance, since files consist of finite length strings of symbols. Infinite length sources provide a better model for communication lines which provide a string of symbols which, if not infinite, typically have no readily apparent end. In fact, even in some cases in which the data is finite, it is convenient to use the infinite length source model. For instance, the widely used adaptive coding techniques (see, for instance [5]) typically use arithmetic coding which implicitly assumes an infinite length source (although practical implementations make modifications so that it may be used with finite length strings). In this paper, we formalize the notion of encoding an infinite length source. While such infinite codes are used intuitively throughout the literature, their mathematical formalization reveals certain subtleties which might otherwise be overlooked. For instance, it turns out that the pure arithmetic code for certain sources has not only unbounded but infinite delay (that is, it is necessary to see a complete infinite source string before being able to determine even one bit of the encoded string in certain cases). Fortunately, such cases occur with zero probability. The formalization presented here leads to a better understanding of infinite coding and a methodology for designing better infinite codes for adaptive data compression (see [1]).

First, we introduce some notation. Let $X = \{0, \ldots, b-1\}$ for some natural number b (this represents the coding alphabet). Let X^m denote the m-fold Cartesian product of X. Let X^* denote the set of all finite strings on X, that is, $X^* = \cup_{m=0}^{\infty} X^m$. Let X^∞ denote the sets of all infinite length strings on X (that is, functions from the natural numbers to X). For $x \in X^*$ and $x' \in X^* \cup X^\infty$, the notation xx' denotes the concatenation of x and x'. We make no distinction between a finite string $x \in X^*$, and the set of all infinite extensions of x, that is, we equate x and $\{xx' : x' \in X^\infty\}$. We let $\sigma(X^*)$ denote the σ-algebra on X^∞ which is the infinite product of the discrete σ-algebra on X. We write $x(1:m)$ for the first m symbols of an infinite string $x \in X^\infty$. Let R be the function which converts an infinite string into the real number for which x is a b-ary representation, that is:

$$R(x) = \sum_{i=1}^{\infty} x(i) b^{-i}$$

for all $x \in X^\infty$.

An infinite length code, f, is a partial function from infinite length strings to infinite length strings, that is, $f : X^\infty \to X^\infty$. In order for a code to be decodable, it must be injective. Also, we assume that f is measurable and has a measurable inverse. In order for such a code to be practically realizable, it must have finite delay, that is, one must be able to determine any finite portion of the encoded string given a sufficiently long portion of the source string. We now define the delay of an infinite code.

Definition 1 Let $f : X^\infty \to X^\infty$ be an infinite code and $x \in X^\infty$ such that $f(x) = x'$. Fix m and let $N' = \{n : f(x(1:n)) \subseteq x'(1:m)\}$. The *m digit delay* of an infinite code f at x is:

$$d_f(m, x) = \min_{n \in N'} n$$

if there is such an n and ∞ otherwise. In other words, it is the the number of digits of x needed to determine that $x'(1:m)$ is the first m digits of the encoding of x under f. We say that f has *finite delay* at x if for all m, the m digit delay of f at x is finite.

In fact, this property of finite delay corresponds with continuity in the natural topologies on infinite strings. The topologies on infinite strings are the lexicographic order topology and the product discrete topology which are equivalent; see [1] for details. Note that in order for the decoder to be practically realizable, it must also have finite delay which means that a finite delay infinite code must have a continuous inverse. In short, a finite delay infinite code is a homeomorphism between subsets of X^∞.

Now suppose we are to design infinite codes for data compression. We need some measure to compare infinite codes in terms of their ability to compress data. The length of any infinite string is infinite and so the notion of coding length is not as immediately apparent as in the case of finite strings. We consider the encoded length of an infinite string to be the infinite sequence of the encoding lengths of all finite initial segments of that string (note that coding length then becomes only partially ordered and not totally ordered as in the case of finite coding length but this turns out to be of little significance). Hence, we need to define the length of encoding of finite strings under infinite codes. However, this definition is also not immediately apparent because an infinite code does not necessarily encode a finite string with another finite string. The encoding of a finite string is the set of all infinite strings which are encodings of infinite extensions of the finite string (that is, the set of all strings which could be generated by the coder given that finite portion of the source string is known). In other words, the encoding of $x \in X^*$ under $f : X^\infty \to X^\infty$ is $f(x)$ which in general is a member of $\sigma(X^*)$. In order to define the notion of coding length for sets of $\sigma(X^*)$, we make an appeal to intuition (indeed, any suggestion that a mathematical concept corresponds with an object that it is purported to model is an appeal to intuition but some are more apparent than others). We define the length of such sets axiomatically, that is, by providing certain axioms which are natural for a measure of length and then showing the unique measure of length satisfying these axioms. This definition of length naturally extends the length of finite strings (a finite string here is the set of its infinite extensions). Let $l : \sigma(X^*) \to [0, \infty]$ be a length function. For a finite string $x \in X^*$, let $|x|$ denote its length. The first axiom is simply that l corresponds with $|\cdot|$ on X^*.

Axiom 1 *For any $x \in X^*$:*

$$l(x) = |x|$$

The second axiom is that l is monotonic. Let $x, x' \in \sigma(X^*)$ be such that $x \subseteq x'$. It is natural that we should have $l(x) \geq l(x')$. For instance, if $x, x' \in X^*$ then $x \subseteq x'$ implies that x' is a prefix of x and so $l(x) \geq l(x')$.

Axiom 2 *If $x, x' \in \sigma(X^*)$ with $x \subseteq x'$ then:*

$$l(x) \geq l(x')$$

For the next axiom, we need the following definition.

Definition 2 An infinite function t is a *translation* if there is a permutation, $f : X^m \longrightarrow X^m$ such that $t_f(xx') = f(x)x'$ for each $x \in X^m$ and $x' \in X^\infty$.

A translation corresponds with an infinite code which just "moves" around some finite strings. Examples of translations are functions which complement the first bit of an infinite string or which complement any finite set of bits. It seems reasonable that such functions do not affect the length of strings. Indeed, they do not alter the length of finite length strings (elements of X^*) which remain finite length strings under them. This is the third axiom.

Axiom 3 *For any translation t_f and $x \in \sigma(X^*)$:*

$$l(t_f(x)) = l(x)$$

For the next axiom, we make another definition.

Definition 3 Let x and x' be open sets. Hence[1], $x = \cup_i x_i$ and $x' = \cup_i x'_i$ for $x_i, x'_i \in X^*$. The *concatenation* of x and x' is defined as:

$$xx' = \cup_{i,j} x_i x'_j$$

The concatenation of open sets is a natural extension of concatenation of finite strings. Just as for $| \cdot |$ on X^*, the length function should be additive for concatenations. This is the fourth axiom.

Axiom 4 *If x and x' are open sets then:*

$$l(xx') = l(x) + l(x')$$

[1] Because the basis X^* of the topology is countable.

In fact, we only need this axiom to hold for finite unions of finite strings rather than arbitrary open sets (which are countable unions of finite strings). The fifth axiom is continuity. This one is perhaps somewhat difficult to justify because there is really no equivalent notion for finite strings. Let $x \in X^\infty$. We have that $\lim x(1:n) = \{x\}$ and it would seem natural to choose $l(x) = \infty$. Also, $\lim l(x(1:n)) = \infty$ and so $\lim l(x(1:n)) = l(\lim x(1:n))$. The final axiom asserts the extension of this to arbitrary measurable sets.

Axiom 5 *If $x, x_1, x_2, \ldots \in \sigma(X^*)$ and $\lim_i x_i = x$[2] then:*

$$\lim_i l(x_i) = l(\lim_i x_i) = l(x)$$

Now we define the code length for measurable sets to be a set function obeying the axioms described above, which we prove defines a unique set function.

Theorem 1 *There is a unique set function, $l : \sigma(X^*) \to [0, \infty]$, satisfying the Axioms 1, 2, 3, 4 and 5. Let μ be the Lebesgue measure on the real line. The unique set function at a measurable set x is given by:*

$$l(x) = -\log_c \mu(R(x))$$

Proof. See [1] for a proof. □

An infinite source is a distribution on a set of infinite strings, i.e. a stochastic process, on the measurable space $(X^\infty, \sigma(X^*))$. Note we do not immediately assume any of the special properties typically attributed to sources in information theory such as independence and identical distribution, ergodicity, stationarity, etc. We will assume that the infinite sources have no atoms, that is, there are no infinite strings of positive probability (for example, if the symbols are chosen independently then the only distributions having an atom are those in which some symbol has probability one and so our assumption is not terribly restrictive).

A measurable partition of X^∞ is a countable partition $x_1, x_2, \ldots$ of X^∞ and such that $x_i \in \sigma(X^*)$ for all i. Now let P be an infinite source, f an infinite code and $x_1, x_2, \ldots$ a measurable partition. The P-average coding length of $x_1, x_2, \ldots$ under f is $\sum_i P(x_i) l(f(x_i))$. Given an infinite source P, there is a method, known as arithmetic coding, of deriving an infinite code which achieves the minimal P-average coding length[3] for all measurable partitions. The arithmetic code basically corresponds with the cumulative distribution function[4] for P. However, as mentioned previously, the arithmetic code does not have finite delay for certain distributions. For instance, the distribution which chooses an infinite string of 0's and 1's independently and with non-dyadic probabilities has infinite delay at certain infinite strings (indeed, it is not even well-defined at certain strings). However, these problems occur only on a countable set (which has zero probability since we have assumed that the source has no atoms). In fact, the arithmetic code can be defined on a set of P probability 1 on which it has finite delay:

[2] Using the usual notion of limits of sets.

[3] The minimal P-average coding length of a measurable partition is the entropy of the partition under P.

[4] The concept of a cumulative distribution function makes sense for any measurable space which uses the Borel σ-algebra based on an order topology which turns out to be the case here.

Theorem 2 *Let P be a source distribution without atoms and let F be its cumulative distribution function, that is, $F(x) = P(\{x' \in X^\infty : x' \leq x\})$. There is an infinite code, $f : X^\infty \longrightarrow X^\infty$ such that f has finite delay and $P(f^{-1}(X^\infty)) = 1$ and such that:*

$$R(f(x)) = F(x)$$

for all $x \in f^{-1}(X^\infty)$. In fact, f achieves the minimal P-average coding length for all measurable partitions of X^∞.

Proof. See [1] for a proof. □

Also, given an infinite code which is "admissible" in a certain sense, there is an infinite source P for which it achieves the minimal P-average coding length for all measurable partitions. In [2, page 12], they define static and adaptive modeling for coding. Since ultimately these modeling techniques produce an infinite code, and since, as mentioned above, this code has some distribution for which it has minimal average coding length, every code can be considered as a static code (in fact, this is the case for finite codes as well which can be shown using the Kraft inequality and some other facts). The advantage of adaptive codes is that, while they are optimal for some distribution, they are nearly optimal for some wide class of distributions. Thus, adaptive codes have the advantage of being statistically "robust" for important classes of distributions. Hence, we can design "optimal" adaptive codes using methodologies from robust statistics such as Bayesian and minimax decision rules over the classes of distributions of interest. This is explored further in [1]. In particular, we extend some of the work of [4] and [3].

References

[1] Kevin Atteson. *Some Mathematical Analyses of Adaptive Text Data Compression.* PhD thesis, University of Pennsylvania, To appear.

[2] Timothy C. Bell, John G. Cleary, and Ian H. Witten. *Text Compression.* Prentice Hall, 1990.

[3] Bertrand Salem Clarke. *Asymptotic Cumulative Risk and Bayes Risk under Entropy Loss, with Applications.* PhD thesis, University of Illinois at Urbanna-Champaign, 1989.

[4] Toshiyasu Matsushima, Hiroshige Inazumi, and Shigeichi Hirasawa. A class of distortionless codes designed by bayes decision theory. *IEEE Transactions on Information Theory*, 37(5):1288–1293, September 1991.

[5] Ross N. Williams. *Adaptive Data Compression.* Kluwer Academic Publishers, 1991.