Lecture 6 Source Coding Theorem

September 16th, 2022

Outline

1. Review of execrises
2. Data Compression
3. Information content defined in terms of lossy compression
4. Typical set

Question

Let $\pi(n)$ denote the number of primes no greater than n. Note that every positive integer n has a unique prime factorization of the form

$$n = \Pi_{i=1}^{\pi(n)} p_i^{X_i},$$

where $p_1, p_2, \ldots$ are primes, and $X_i = X_i(n)$ is the non-negative integer representing the multiplicity of p_i in the prime factorization of n. Let N be uniformly distributed on $\{1, 2, 3, \ldots, n\}$.

(1) Show that $X_i(N)$ is an integer-valued random variable satisfying

$$0 \leq X_i(N) \leq \log n.$$

(2) Show that

$$\log n = H(N) \leq \pi(n) \log(\log n + 1).$$

Thus not only is $\pi(n) \to \infty$ but in fact $\pi(n) \geq \frac{\log n}{\log(\log n+1)}$.

Proof.

(1) $0 \leq X_i(N)$ is trivial. Note also that $2^{X_i} \leq p_i^{X_i} \leq N \leq n$. Thus, combining both results, $0 \leq X_i(N) \leq \log n$, as we wanted to show.

(2)

$$
\begin{aligned}
\log n &= H(N) \\
&= H(X_1, X_2, \ldots, X_{\pi(n)}) \\
&= \sum_{i=1}^{\pi(n)} H(X_i | X_1, \ldots, X_{i-1}) \\
&\leq H(X_1) + H(X_2) + \ldots + H(X_{\pi(n)}) \\
&= \pi(n) \log(\log n + 1),
\end{aligned}
$$

where the first step follows because there is a one-to-one mapping between N and $X_1, X_2, \ldots, X_{\pi(n)}$. The second step is by the chain rule for entropy. The next step is because conditioning reduced entropy, and the last one is because the distribution that maximizes entropy is the uniform one, there are $\pi(n)$ entropy terms, and X_i's can take at most $\log n + 1$ different values. $\square$

Outline

1. Review of execrises
2. Data Compression
3. Information content defined in terms of lossy compression
4. Typical set

A file is composed of a sequence of types. A byte is composed of 8 bits and can have a decimal value between 0 and 255. A typical text file is composed of the ASCII character set (decimal values 0 to 127). This character set uses only seven of the eight bits in a byte.

A file is composed of a sequence of types. A byte is composed of 8 bits and can have a decimal value between 0 and 255. A typical text file is composed of the ASCII character set (decimal values 0 to 127). This character set uses only seven of the eight bits in a byte.

Question

By how much could the size of a file be reduce given that it is an ASCII file? How would you achieve this reduction?

One way of measuring the information content of a random variable is simply to count the number of possible outcomes, $|\mathcal{A}_X|$. If we gave a binary name to each outcome, the length of each name would be $\log_2 |\mathcal{A}_X|$ bits, if $|\mathcal{A}_X|$ happened to be a power of 2.

One way of measuring the information content of a random variable is simply to count the number of possible outcomes, $|\mathcal{A}_X|$. If we gave a binary name to each outcome, the length of each name would be $\log_2 |\mathcal{A}_X|$ bits, if $|\mathcal{A}_X|$ happened to be a power of 2.

Definition

The raw bit content of X is

$$H_0(X) = \log_2 |\mathcal{A}_X|.$$

Question

Could there be a compressor that maps an outcome x to a binary code $c(x)$, and a decompressor that maps c back to x, such that every possible outcome is compressed into a binary code of length shorter than $H_0(X)$ bits?

Outline

Example

Let

$$\mathcal{X} = \{a, b, c, d, e, f, g, h\}$$

and

$$P_X = \{\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{3}{16}, \frac{1}{64}, \frac{1}{64}, \frac{1}{64}, \frac{1}{64}\}.$$

The raw bit content of this ensemble is 3 *bits, corresponding to* 8 *binary names. But notice that* $P(x \in \{a, b, c, d\}) = 15/16$*. So if we are willing to run a risk of* $\delta = 1/16$ *of not having a name for* x*, then we can get by four names - half as many names as are needed if every* $x \in \mathcal{X}$ *has a name.*

Definition

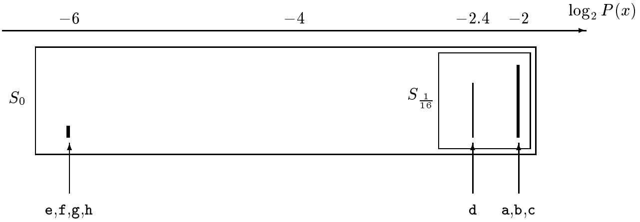
The smallest δ-sufficient subset S_δ is the smallest subset of $\mathcal{A}_X$ satisfying

$$P(x \in S_\delta) \geq 1 - \delta.$$

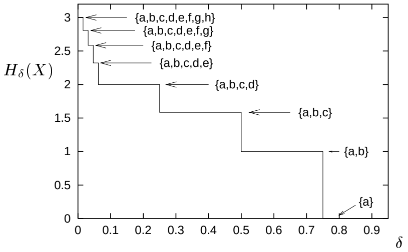
Definition

The essential bit content of X is

$$H_\delta(X) = \log_2 |S_\delta|.$$


-6
-4
-2.4
-2
$\log_2 P(x)$
S_0
$S_{\frac{1}{16}}$
e,f,g,h
d
a,b,c

(a)


$H_\delta(X)$
{a,b,c,d,e,f,g,h}
{a,b,c,d,e,f,g}
{a,b,c,d,e,f}
{a,b,c,d,e}
{a,b,c,d}
{a,b,c}
{a,b}
{a}
0
0.5
1
1.5
2
2.5
3
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
δ

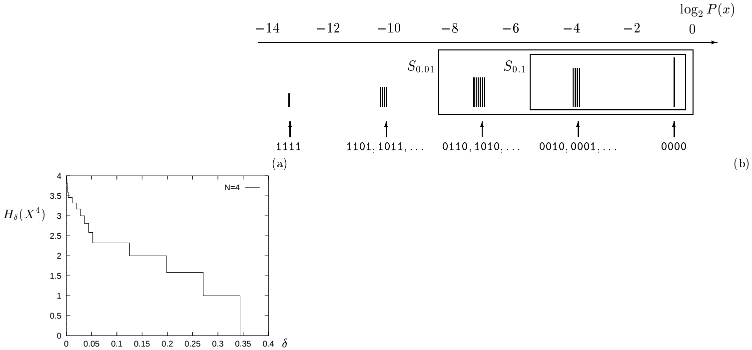
(b)

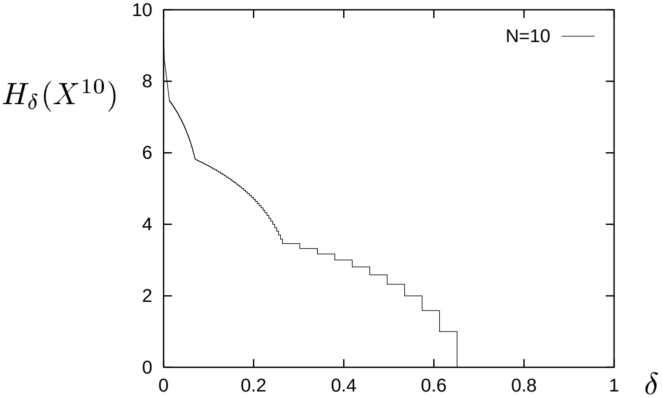
We now turn to examples where the outcome $\mathbf{x} = (x_1, x_2, \ldots, x_N)$ is a string of N independent identically distributed random variables from a single random variable X. We will denote by X^N the random vector $(X_1, X_2, \ldots, X_n)$. Remember that entropy is additive for independent variables, so $H(X^N) = NH(X)$.

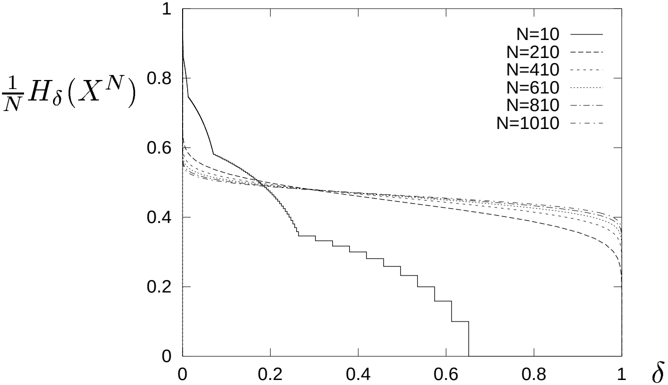
Example

Consider a string of N flips of a bent coin, $\boldsymbol{x} = (x_1, x_2, \ldots, x_N)$, where $x_n \in \{0, 1\}$, with probabilities $p_0 = 0.9$, $p_1 = 0.1$. If $r(\boldsymbol{x})$ is the number of 1s in $\boldsymbol{x}$ then

$$P(\boldsymbol{x}) = p_0^{N-r(\boldsymbol{x})} p_1^{r(\boldsymbol{x})}.$$


$\log_2 P(x)$
−14
−12
−10
−8
−6
−4
−2
0
$S_{0.01}$
$S_{0.1}$
1111
1101, 1011, . . .
0110, 1010, . . .
0010, 0001, . . .
0000
(a)
(b)
$H_\delta(X^4)$
N=4
δ


$H_\delta(X^{10})$
10
8
6
4
2
0
0
0.2
0.4
0.6
0.8
1
δ
N=10


$\frac{1}{N} H_\delta(X^N)$
1
0.8
0.6
0.4
0.2
0
0
0.2
0.4
0.6
0.8
1
δ
N=10
N=210
N=410
N=610
N=810
N=1010

Theorem (Shannon's source coding theorem)

Let X be an random variable with entropy $H(X) = H$ bits. Given $\epsilon > 0$ and $0 < \delta < 1$, there exists a positive integer N_0 such that for $N > N_0$,

$$|\frac{1}{N} H_\delta(X^N) - H| < \epsilon.$$