Advanced Techniques for Combinatorial Algorithms: Introduction to Information Theory

Gianluca Della Vedova

Univ. Milano–Bicocca
http://gianluca.dellavedova.org

2016-01-29T18:16:39+0000 71e9e9f

Entropy

Entropy

Let A be a discrete random variable over universe set U. Then its entropy is

$$H(A) = \sum_{x \in U} p_x \log_2 \frac{1}{p_x} = -\sum_{x \in U} p_x \log_2 p_x$$

where p_x is the probability of the event x.

Entropy measures information content.

Entropy 2

Entropy of a text T

Let $f(\sigma)$ be the relative frequency of symbol σ in T. Then its entropy is

$$H(T) = -\sum_{\sigma \in \Sigma} f(\sigma) \log_2 f(\sigma)$$

Jensen Inequality

Jensen Inequality

Let A be a random variable, and let ψ be a convex function. Then

$$\psi(E(A)) \leq E(\psi(A))$$

where $E()$ is the expectation.

Prefix code

Code

$$\phi : A \subset \mathbb{N} \mapsto \{0, 1\}^*$$

Variable length codewords

Example

$0 \mapsto 1010$
$1 \mapsto 0010$
$2 \mapsto 00$

Not prefix code

Prefix code

Prefix Code

No codeword is the prefix of another codeword.

Examples

$0 \mapsto 1$

$1 \mapsto 01$

$2 \mapsto 001$

Lemma

All prefix codes are uniquely decodable.

Elias gamma encoding

Encoding $n \geq 1$

1. $N \leftarrow \lfloor \log_2 n \rfloor$
2. N zeroes followed by n in binary

Requires $2\lfloor \log_2 n \rfloor + 1$ bits

Examples

$9 = 1001_2 \mapsto 0001001$

$15 = 1111_2 \mapsto 0001111$

Elias delta encoding

Encoding $n \geq 1$

1. $N \leftarrow \lfloor \log_2 n \rfloor$
2. $L \leftarrow \lfloor \log_2 N + 1 \rfloor$
3. Elias γ encoding of $N + 1$, followed by $n - 2^N$ as N-bits

Requires $2\lfloor \log_2 (\lfloor \log_2 n \rfloor + 1) \rfloor + \lfloor \log_2 n \rfloor + 1$ bits

Examples

$9 = 1001_2 \mapsto 011001$

$15 = 1111_2 \mapsto 011111$

Huffman encoding

All elements of U are weighted.

Algorithm 1: Huffman encoding

```
a, b ← the two lightest elements of U;
c ← new element w(c) ← w(a) + w(b);
if |U| > 1 then
    φ_1 ← Huffman(U \ {a, b} ∪ {c});
    φ ← φ_1;
    φ(a) ← φ_1(c)0; φ(b) ← φ_1(c)1;
    Remove c, φ(c);
else
    φ(u) ← empty codeword;
Return(φ);
```