Chapter 5

Data Compression

Peng-Hua Wang

Graduate Inst. of Comm. Engineering

National Taipei University

Chapter Outline

5.1 Example of Codes

Source code

Definition (Source code) A source code C for a random variable X is a mapping from $\mathcal{X}$, the range of X, to $\mathcal{D}^*$, the set of finite-length strings of symbols from a D-ary alphabet. Let $C(x)$ denote the codeword corresponding to x and let $l(x)$ denote the length of $C(x)$.

- For example, $C(\text{red}) = 00, C(\text{blue}) = 11$ is a source code with mapping from $\mathcal{X} = \{\text{red}, \text{blue}\}$ to $\mathcal{D}^2$ with alphabet $\mathcal{D} = \{0, 1\}$.

Source code

Definition (Expected length) The expected length $L(C)$ of a source code $C(x)$ for a random variable X with probability mass function $p(x)$ is given by

$$L(C) = \sum_{x \in \mathcal{X}} p(x) l(x).$$

where $l(X)$ is the length of the codeword associated with X.

Example

Example 5.1.1 Let X be a random variable with the following distribution and codeword assignment

$$\Pr\{X=1\}=\frac{1}{2}, \text{ codeword } C(1)=0$$
$$\Pr\{X=2\}=\frac{1}{4}, \text{ codeword } C(2)=10$$
$$\Pr\{X=3\}=\frac{1}{8}, \text{ codeword } C(3)=110$$
$$\Pr\{X=4\}=\frac{1}{8}, \text{ codeword } C(4)=111$$

- $H(X)=1.75$ bits.
- $El(x)=1.75$ bits.
- uniquely decodable

Example

Example 5.1.2 Consider following example.

$$\Pr\{X=1\}=\frac{1}{3}, \text{ codeword } C(1)=0$$
$$\Pr\{X=2\}=\frac{1}{3}, \text{ codeword } C(2)=10$$
$$\Pr\{X=3\}=\frac{1}{3}, \text{ codeword } C(3)=11$$

- $H(X)=1.58$ bits.
- $El(x)=1.66$ bits.
- uniquely decodable

Source code

Definition (non-singular) A code is said to be nonsingular if every element of the range of X maps into a different string in $\mathcal{D}^*$; that is,

$$x \neq x' \Rightarrow C(x) \neq C(x')$$

Definition (extension code) The extension C^* of a code C is the mapping from finite length-strings of $\mathcal{X}$ to finite-length strings of $\mathcal{D}$, defined by

$$C(x_1 x_2 \cdots x_n) = C(x_1)C(x_2)\cdots C(x_n)$$

where $C(x_1)C(x_2)\cdots C(x_n)$ indicates concatenation of the corresponding codewords.

Example 5.1.4 If $C(x_1) = 00$ and $C(x_2) = 11$, then $C(x_1 x_2) = 0011$.

Source code

Definition (uniquely decodable) A code is called uniquely decodable if its extension is nonsingular.

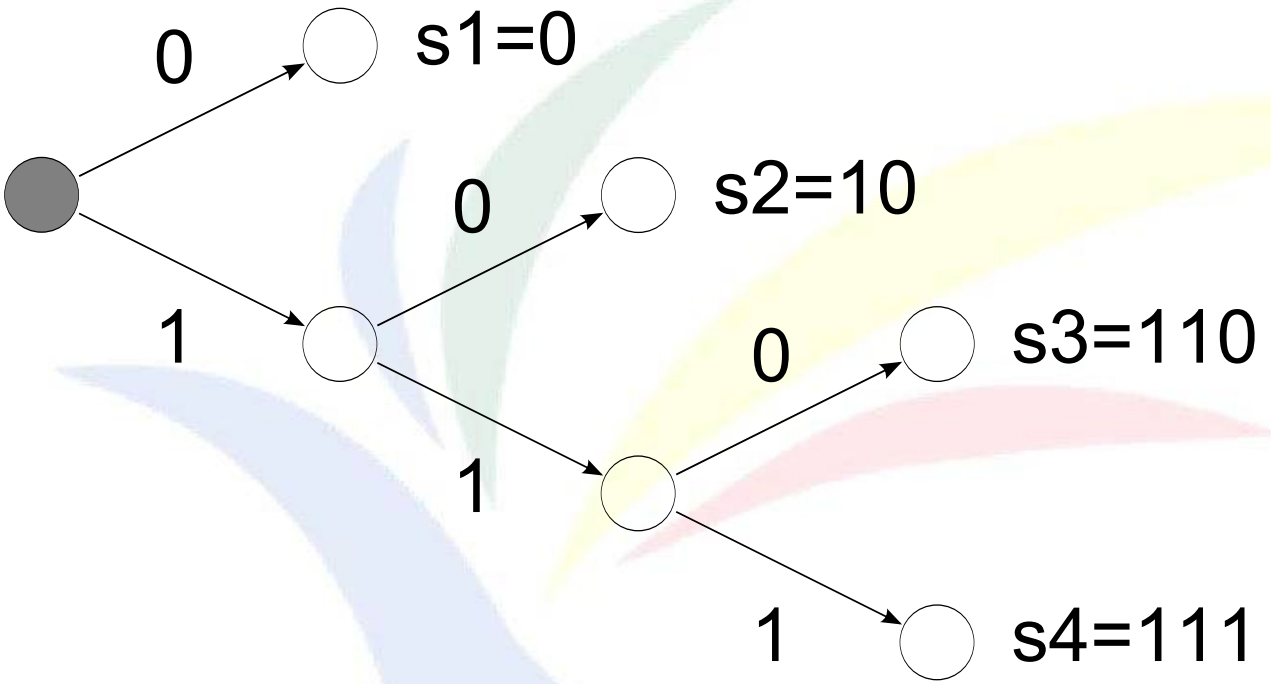
Definition (prefix code) A code is called a prefix code or an instantaneous code if no codeword is a prefix of any other codeword.

- For an instantaneous code, the symbol x_i can be decoded as soon as we come to the end of the codeword corresponding to it.
- For example, the binary string 01011111010 produced by the code of Example 5.1.1 is parsed as $0, 10, 111, 110, 10$.

Source code

X	Singular	Nonsingular, but not UD	UD, But Not Inst.	Inst.
1	0	0	10	0
2	0	010	00	10
3	0	01	11	110
4	0	10	110	111

Decoding Tree



5.2 Kraft Inequality

Kraft Inequality

Theorem 5.2.1 (Kraft Inequality) For any instantaneous code (prefix code) over an alphabet of size D, the codeword lengths $l_1, l_2, \ldots, l_m$ must satisfy the inequality

$$\sum_i D^{-l_i} \leq 1.$$

Conversely, given a set of codeword lengths that satisfy this inequality, there exists an instantaneous code with these word lengths.

Extended Kraft Inequality

Theorem 5.2.2 (Extended Kraft Inequality) For any countably infinite set of codewords that form a prefix code, the codeword lengths satisfy the extended Kraft inequality,

$$\sum_{i=1}^{\infty} D^{-l_i} \leq 1.$$

Conversely, given any $l_1, l_2, \ldots$ satisfying the extended Kraft inequality, we can construct a prefix code with these codeword lengths.

5.3 Optimal Codes

Minimize expected length

Problem Given the source pmf $pP_1, p_2, \ldots p_m$, find the code length $l_1, l_2, \ldots, l_m$ such that the expected code length is minimized

$$L = \sum p_i l_i$$

with constraint

$$\sum D^{-l_i} \leq 1.$$

- $l_1, l_2, \ldots, l_m$ are integers.
- We first relax the original integer programming problem. The restriction of integer length is relaxed to real number.
- Solve by Lagrange multipliers.

Solve the relaxed problem

$$J = \sum p_i l_i + \lambda \left(\sum D^{-l_i} \right), \quad \frac{\partial J}{\partial l_i} = p_i - \lambda D^{-l_i} \ln D$$

$$\frac{\partial J}{\partial l_i} = 0 \Rightarrow D^{-l_i} = \frac{p_i}{\lambda \ln D}$$

$$\sum D^{-l_i} \leq 1 \Rightarrow \lambda = \frac{1}{\ln D} \Rightarrow p_i = D^{-l_i}$$

$\Rightarrow$ optimal code length $l_i^* = -\log_D p_i$ $\square$

The expected code length is

$$L^* = \sum p_i l_i^* = -\sum p_i \log_D p_i = H_D(X)$$

Expected code length

Theorem 5.3.1 The expected length L of any instantaneous D-ary code for a random variable X is greater than or equal to the entropy $H_D(X)$; that is,

$$L \geq H_D(X)$$

with equality if and only if $D^{-l_i} = p_i$.

Proof.

$$\begin{aligned} L - H_D(X) &= \sum p_i l_i + \sum p_i \log_D p_i \\ &= -\sum p_i \log_D D^{-l_i} + \sum p_i \log_D p_i \\ &= D(p||q) \geq 0 \end{aligned}$$

where $q_i = D^{-l_i}$.

WRONG!! Because $\sum D^{-l_i} \leq 1$ may not be a valid distribution.

Expected code length

Proof.

$$\begin{aligned} L - H_D(X) &= \sum p_i l_i + \sum p_i \log_D p_i \\ &= -\sum p_i \log_D D^{-l_i} + \sum p_i \log_D p_i \end{aligned}$$

Let $c = \sum_j D^{-l_j}, r_i = D^{-l_i}/c$,

$$\begin{aligned} L - H_D(X) &= \sum p_i \log_D \frac{p_i}{r_i} - \log_D c \\ &= D(p||r) + \log_D \frac{1}{c} \geq 0 \end{aligned}$$

since $D(p||r) \geq 0$ and $c \leq 1$ by Kraft inequality. $L \leq H_D(X)$ with equality iff $p_i = D^{-l_i}$. That is, iff $-\log_D p_i$ is an integer for all i. $\square$

D-adic

Definition (D-adic) A probability distribution is called D-adic if each of the probabilities is equal to D^{-n} for some n.

- $L = H_D(X)$ if and only if the distribution of X is D-adic.
- How to find the optimal code? $\Rightarrow$ Find the D-adic distribution that is closest (in the relative entropy sense) to the distribution of X.
- What is the upper bound of the optimal code ?

5.4 Bound on Optimal Code Length

Optimal code length

Theorem 5.4.1 Let $l_1^*, l_2^*, \ldots, l_m^*$ be optimal codeword lengths for a source distribution $\mathbf{p}$ and sa D-ary alphabet, and let L^* be the associated expected length of an optimal code ($L^|ast = \sum p_i l_i^*$). Then

$$H_D(X) \leq L^* < H_D(X) + 1.$$

Proof. Let $l_i = \lceil \log_D \frac{1}{p_i} \rceil$ where $\lceil x \rceil$ is the smallest integer $\geq x$. These lengths satisfy the Kraft inequality since

$$\sum D^{-\lceil \log_D \frac{1}{p_i} \rceil} \leq D^{-\log_D \frac{1}{p_i}} = \sum p_i = 1.$$

These choice of codeword lengths satisfies

$$\log_D \frac{1}{p_i} \leq l_i \leq \log_D \frac{1}{p_i} + 1.$$

Optimal code length

Multiplying by p_i and summing over i, we obtain

$$H_D(X) \leq L < H_D(X) + 1.$$

Since L^* is the expected length of the optimal code,

$$L^* \leq L < H_D(X) + 1.$$

On another hand, from Theorem 5.3.1,

$$L^* \geq H_D(X).$$

Therefore,

$$H_D(X) \leq L^* < H_D(X) + 1. \quad \square$$

Optimal code length

Consider a system in which we send a sequence of n symbols from X.

Define L_n to be the expected codeword length per input symbol,

$$\begin{aligned} L_n &= \frac{1}{n}\sum p(x_1, x_2, \ldots, x_n) l(x_1, x_2, \ldots, x_n) \\ &= \frac{1}{n} E[l(X_1, X_2, \ldots, X_n)] \end{aligned}$$

We have

$$\begin{aligned} H(X_1, X_2, \ldots, X_n) &\leq E[l(X_1, X_2, \ldots, X_n)] \\ &< H(X_1, X_2, \ldots, X_n) + 1 \end{aligned}$$

If $X_1, X_2, \ldots, X_n$ are i.i.d., we have

$$H(X) \leq L_n < H(X) + \frac{1}{n}$$

Optimal code length

Consider a system in which we send a sequence of n symbols from X.

Define L_n to be the expected codeword length per input symbol,

$$\begin{aligned} L_n &= \frac{1}{n}\sum p(x_1, x_2, \ldots, x_n) l(x_1, x_2, \ldots, x_n) \\ &= \frac{1}{n} E[l(X_1, X_2, \ldots, X_n)] \end{aligned}$$

We have

$$\begin{aligned} H(X_1, X_2, \ldots, X_n) &\leq E[l(X_1, X_2, \ldots, X_n)] \\ &< H(X_1, X_2, \ldots, X_n) + 1 \end{aligned}$$

Optimal code length

If $X_1, X_2, \ldots, X_n$ are independent but not identically distributed, we have

$$\frac{1}{n}H(X_1, X_2, \ldots, X_n) \leq L_n < \frac{1}{n}H(X_1, X_2, \ldots, X_n) + 1$$

If the random process is stationary

$$\frac{1}{n}H(X_1, X_2, \ldots, X_n) \to H(\mathcal{X})$$

Optimal code length

Theorem 5.4.2 The minimum expected codeword length per symbol satisfies

$$\frac{1}{n}H(X_1, X_2, \ldots, X_n) \leq L_n^* < \frac{1}{n}H(X_1, X_2, \ldots, X_n) + 1$$

If $X_1, X_2, \ldots, X_n$ is a stationary random process,

$$L_n^* \to H(\mathcal{X})$$

where $H(\mathcal{X})$ is the entropy rate of the random process.

Wrong Code

Theorem 5.4.3 (Wrong Code) The expected length under $p(x)$ of the code assignment $l(x) = \lceil \log \frac{1}{q(x)} \rceil$ satisfies

$$H(p) + D(p||q) \leq E_p[l(X)] < H(p) + D(p||q) + 1.$$

Proof. The expected codelength is

$$\begin{aligned} E[l(X)] &= \sum_x p(x) \left\lceil \log \frac{1}{q(x)} \right\rceil < \sum_x p(x) \left(\log \frac{1}{q(x)} + 1 \right) \\ &= \sum_x p(x) \log \frac{1}{q(x)} + 1 \\ &= \sum_x p(x) \left(\log \frac{p(x)}{q(x)} - \log p(x) \right) + 1 \\ &= H(p) + D(p||q) + 1 \end{aligned}$$

The lower bound can be derived similarly. □

5.5 Kraft Inequality for Uniquely Decodable Codes

Uniquely Decodable Codes

Theorem 5.5.1 (Wrong Code) The codeword lengths of any uniquely decodable D-ary code must satisfy the Kraft inequality

$$\sum D^{-l_i} \leq 1.$$

Conversely, given a set of codeword lengths that satisfy this inequality, it is possible to construct a uniquely decodable code with these codeword lengths.

Proof. Consider

$$\left(\sum_x D^{-l(x)}\right)^k = \sum_{x_1}\sum_{x_2}\cdots\sum_{x_k} D^{-l(x_1)}D^{-l(x_2)}\cdots D^{-l(x_k)}$$

$$= \sum_{x_1,\ldots,x_k\in\mathcal{X}^k} D^{-l(x_1)}D^{-l(x_2)}\cdots D^{-l(x_k)} = \sum_{x^k\in\mathcal{X}^k} D^{-l(x^k)}$$

Uniquely Decodable Codes

We now gather the terms by word lengths to obtain

$$\sum_{x^k \in \mathcal{X}^k} D^{-l(x^k)} = \sum_{m=1}^{kl_{\max}} a(m) D^{-m}$$

where $l_{\max}$ is the maximum codeword length and $a(m)$ is the number of source sequences mapping into codewords of length m.

Since the code is uniquely decodable, so there is at most one sequence mapping into each code m-sequence and there are at most D^m code m-sequences. Thus,

$$a(m) \leq D^m.$$

Uniquely Decodable Codes

Therefore,

$$\left(\sum_x D^{-l(x)}\right)^k = \sum_{m=1}^{kl_{\max}} a(m) D^{-m} \leq \sum_{m=1}^{kl_{\max}} D^m D^{-m} = kl_{\max}$$

or

$$\sum_j D^{-l_j} \leq (kl_{\max})^{1/k}$$

Since this inequality is true for all k, it is true in the limit as $k \to \infty$.

Since $(kl_{\max})^{1/k} \to 1$, we have

$$\sum_j D^{-l_j} \leq 1. \quad \square$$

5.6 Huffman Codes

Example 5.6.1

$\mathcal{X} = \{1, 2, 3, 4, 5\}, p = \{0.25, 0.25, 0.2, 0.15, 0.15\}$, binary.

Example 5.6.2

$\mathcal{X} = \{1, 2, 3, 4, 5\}, p = \{0.25, 0.25, 0.2, 0.15, 0.15\}$, ternary.

Example 5.6.3

$\mathcal{X} = \{1, 2, 3, 4, 5, 6\}, p = \{0.25, 0.25, 0.2, 0.1, 0.1, 0.1\}$, ternary.

- At kth stage, the total number of symbols is $1 + k(D - 1)$.

Shannon Code

- Using codeword lengths of $\lceil \log \frac{1}{p_i} \rceil$
- May be much worse than the optimal code. For example, $p_1 = 1 - 1/1024$ and $p_2 = 1/1024$. $\lceil \log \frac{1}{p_1} \rceil = 1$ and $\lceil \log \frac{1}{p_2} \rceil = 10$. However, we can use exactly 1 bit.

5.8 Optimality of Huffman Codes

Properties of Optimal Codes

Lemma 5.8.1 For any distribution, there exists an optimal instantaneous code that satisfies the following properties:

1. If $p_j > p_k$, then $l_j \leq l_k$.
2. The two longest codewords have the same length.
3. Two of the longest codewords differ only in the last bits.

Optimality of Huffman codes

Lemma 5.8.2 Let C be the codes with distribution $p_1 \leq p_2 \leq \cdots \leq p_{K-1} \leq p_K$. C' is the codes with distribution $p_1, p_2, \ldots, p_{K-1} + p_K$. If C' is optimal with code assignment

$$p_1 \to w_1, p_2 \to w_2, \ldots, p_{K-1} + p_K \to w_{K-1},$$

then C is also optimal with code assignment

$$p_1 \to w_1, p_2 \to w_2, \ldots, p_{K-1} \to w_{K-1}0, p_K \to w_{K-1}1.$$

Optimality of Huffman codes

Proof. The average length for C' is

$$L(C') = p_1 l_1 + p_2 l_2 + \cdots + (p_{K-1} + p_K) l_{K-1}.$$

The average length for C is

$$L(C) = p_1 l_1 + p_2 l_2 + \cdots + p_{K-1}(l_{K-1} + 1) + p_K (l_{K-1} + 1).$$

We have

$$L(C) = L(C') + p_{K-1} + p_K.$$

That is, we can minimize $L(C)$ by minimizing $L(C')$ since $p_{K-1} + p_K$ is a constant. $\square$