信息论与编码

马啸
maxiao@mail.sysu.edu.cn

计算机学院
中山大学

2021 年春季学期

1. Huffman Codes
   - Optimality of Huffman Codes

引例

设 X 的分布律是

X	0	1
p(X)	3/4	1/4

据此构造Shannon码。设 X 为 0 和 1 时的码字分别为 c_0, c_1。计算相应的码长 ℓ_0, ℓ_1：

$$\ell_0 = \lceil \log_2 \frac{1}{p_0} \rceil = 1$$

$$\ell_1 = \lceil \log_2 \frac{1}{p_1} \rceil = 2$$

计算 $F_0 = 0.00\cdots$，$F_1 = 0.1100\cdots$，根据码长对它们取截断，得到码字

$$c_0 = 0$$

$$c_1 = 11$$

平均码长 $\bar{L} = p_0\ell_0 + p_1\ell_1 = \frac{5}{4}$。

从平均码长的角度考虑，Shannon码明显不是最优的，因为 $0 \to 0$，$1 \to 1$ 是最显然的编码，且平均码长为1。

问： 使平均码长最短的编码方法是什么？

Brief Review of Tree

Definition 1 (full tree)

A $D-$ary tree in which each node has exactly zero or D children.

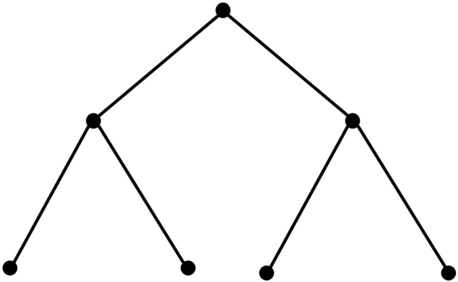


Figure: full tree example

Brief Review of Tree

Definition 2 (complete tree)

A tree in which every level, except possibly the deepest, is entirely filled. At depth n, the height of the tree, all nodes are as far left as possible.

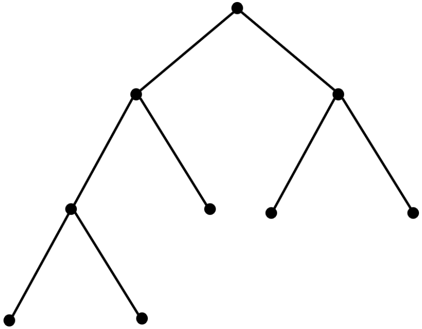


Figure: complete tree example

Brief Review of Tree

Definition 3 (balanced tree)

A balanced binary tree is a binary tree structure in which the left and right subtrees of every node differ in height by no more than 1.

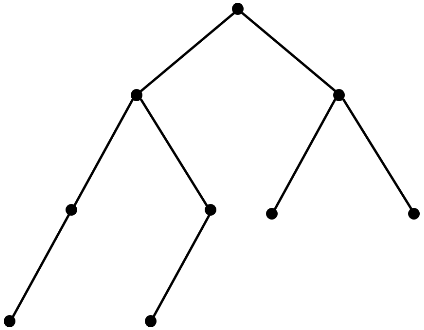


Figure: balanced tree example

例子：Huffman编码

Procedure *Huffman*(C: symbols a_i with frequencies w_i,$i = 1, \cdots, n$)

$F :=$ forest of n rooted trees, each consisting of the single vertex a_i and assigned weight w_i

while F *is not a tree* **do**

> Replace the rooted trees T and $T^{'}$ of least weights from F with $w(T) \geq w(T^{'})$ with a tree having a new root that has T as its left subtree and $T^{'}$ as its right subtree. Label the new edge to T with 0 and the new edge to $T^{'}$ with 1.
>
> Assign $w(T) + w(T^{'})$ as the weight of the new tree.

end

{the Huffman coding for the symbol a_i is the concatenation of the labels of the edges in the unique path from the root to the vertex a_i}

Algorithm 1: Huffman Coding (Kenneth H. Rosen, 2011)

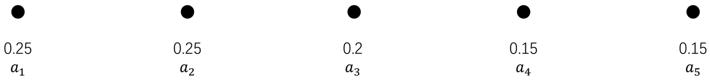
例子：Huffman编码

给定分布下的最优前缀码可以通过Huffman编码构造。

Example 4

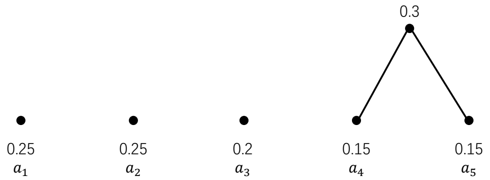
信源 $\mathcal{X} = \{1, 2, 3, 4, 5\}$ 有概率分布 $0.25, 0.25, 0.2, 0.15, 0.15$。在二进制下，Huffman编码的流程如下：

0. 建立森林



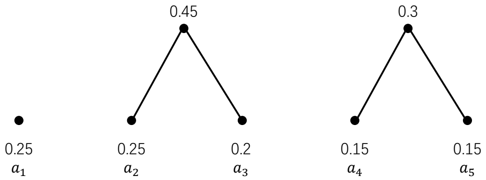
例子：Huffman编码

1. 找森林中两棵权重最小的树，合成一棵。



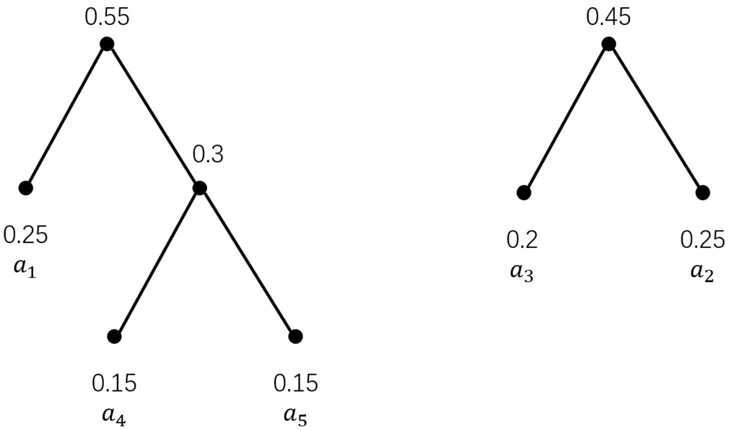
例子：Huffman编码

2. 重复步骤 **1.**。



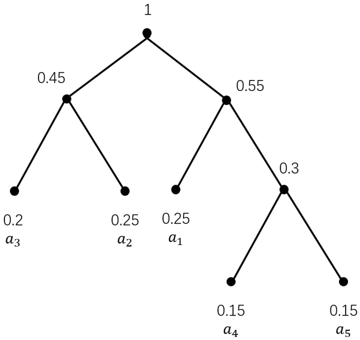
例子：Huffman编码

3. 重复步骤 **1.**。



例子：Huffman编码

4. 重复步骤 **1.**。



有了树之后，左分支是 0，右分支是 1，即可构造码。

主要思想是将更短的码字分配给概率更大的符号。

Huffman编码

例子1：

【例 8. 1】 离散无记忆信源 $S=[s_1, s_2, s_3, s_4, s_5]$，它的一种霍夫曼码如表 8. 1 所示。

表 8. 1 一种霍夫曼码

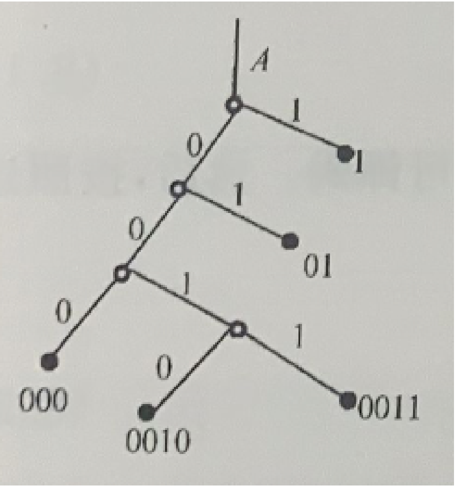
信源符号 s_i	码字长度 l_i	码字 W_i	概率 $P(s_i)$	缩减信源 S_1		S_2		S_3	
s_1	1	1	0. 4	1	0. 4	1	0. 4	1	0. 6 (0)
s_2	2	01	0. 2	01	0. 2	01	0. 4 (0)	00	0. 4 (1)
s_3	3	000	0. 2	000	0. 2 (0)	000	0. 2 (1)	01	
s_4	4	0010	0. 1	0010	0. 2 (1)	001			
s_5	4	0011	0. 1	0011					

它的平均码长为

$$\bar{L}=\sum_{i=1}^{5} P(s_i) l_i = 0.4\times1+0.2\times2+0.2\times3+0.1\times4+0.1\times4$$
$$=2.2 \quad (\text{二元码元/信源符号})$$

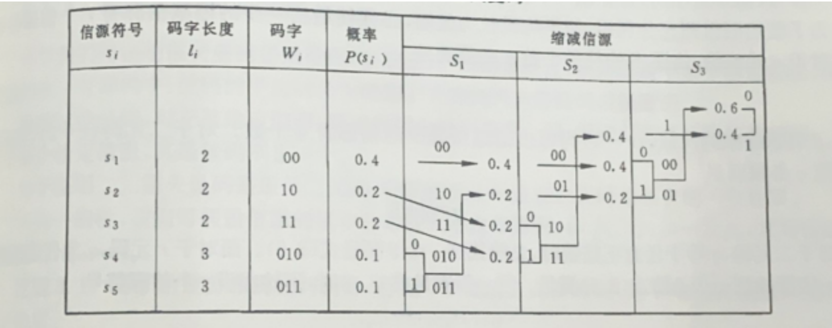
Huffman编码

例子1：



Huffman编码

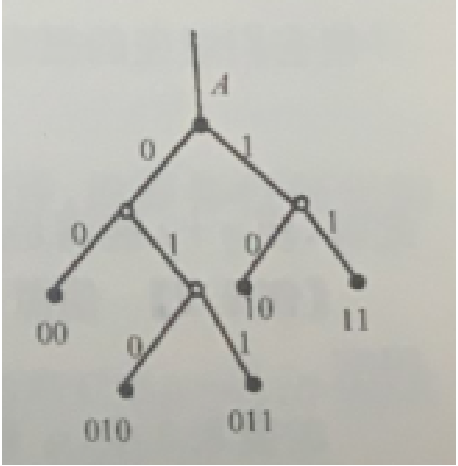
例子2：



信源符号 s_i	码字长度 l_i	码字 W_i	概率 $P(s_i)$	缩减信源 S_1	S_2	S_3
s_1	2	00	0.4	00 0.4	1 0.4	0 0.6
s_2	2	10	0.2	10 0.2	00 0.4	1 0.4
s_3	2	11	0.2	11 0.2	01 0.2	
s_4	3	010	0.1	010 0.2		
s_5	3	011	0.1			

Huffman编码

例子2：



Theorem 5 (Optimal properties)

For any distribution, there exists a binary optimal prefix code $\mathcal{C}$ such that

1. *If $p_j > p_k$, then $\ell_j \leq \ell_k$.*
2. *The two longest codewords have the same length.*
3. *The two longest codewords differ only in the last bit and corresponds to the two least likely symbols.*

Outline of proof:

1. If $p_j > p_k$, we swap their codewords to construct a new code $\mathcal{C}'$. Then
$$L' - L = \sum p_i \ell_i' - \sum p_i \ell_i = (p_j \ell_k + p_k \ell_j) - (p_j \ell_j + p_k \ell_k) = (p_j - p_k)(\ell_k - \ell_j)$$
Since $\mathcal{C}$ is optimal, then $L' - L \geq 0$. Hence $\ell_j \leq \ell_k$ as $p_j > p_k$.
2. If the two longest codewords are not of the same length, then we can delete the last bit of the longer one preserving the prefix condition and achieving lower average length.
3. If there is a maximal length codeword without a sibling, then we can delete the last bit preserving the prefix condition and achieving lower average length.

Theorem 6

If a binary code $\mathcal{C}^*$ is constructed by Huffman coding, then it is a binary optimal code.

Outline of proof: this theorem can be proved by induction.

Let $m = |\mathcal{X}|$ and the code for the source $\mathcal{X}$ is denoted by $\mathcal{C}_m$. Without loss of generality, we assume that $p_1 \geq p_2 \geq \ldots \geq p_m$.

(1) Let $\mathcal{C}_m$ be a code satisfying the optimal properties. Based on $\mathcal{C}_m$, a code $\mathcal{C}_{m-1}$ for $m-1$ symbols is construct as follows.

$$
\begin{array}{ccccccc}
 & \mathcal{C}_{m-1} & & & & \mathcal{C}_m & \\
p_1 & c'_1 & \ell'_1 & & p_1 & c_1 = c'_1 & \ell_1 = \ell'_1 \\
p_2 & c'_2 & \ell'_2 & & p_2 & c_2 = c'_2 & \ell_2 = \ell'_2 \\
\vdots & \vdots & \vdots & \Leftarrow & \vdots & \vdots & \vdots \\
p_{m-2} & c'_{m-2} & \ell'_{m-2} & & p_{m-2} & c_{m-2} = c'_{m-2} & \ell_{m-2} = \ell'_{m-2} \\
p_{m-1} + p_m & c'_{m-1} & \ell'_{m-1} & & p_{m-1} & c_{m-1} = c'_{m-1}0 & \ell_{m-1} = \ell'_{m-1} + 1 \\
 & & & & p_m & c_m = c'_{m-1}1 & \ell_m = \ell'_{m-1} + 1
\end{array}
$$

Outline of proof: this theorem can be proved by induction.

(1') Then we have $L(\mathcal{C}_m) = L(\mathcal{C}_{m-1}) + p_{m-1} + p_m$. Hence

$$\min L(\mathcal{C}_m) \Leftrightarrow \min L(\mathcal{C}_{m-1}).$$

(2) Similarly, we have

$$\min L(\mathcal{C}_{m-1}) \Leftrightarrow \min L(\mathcal{C}_{m-2}) \Leftrightarrow \cdots \Leftrightarrow \min L(\mathcal{C}_3) \Leftrightarrow \min L(\mathcal{C}_2).$$

Hence, the minimization problem is reduced to for two symbols and then we can allot 0 for one the symbol and 1 for the other.

Remark: The optimality of Huffman coding can be extended to any code over D-ary alphabet. Huffman coding is a "greedy" algorithm that the local optimality ensures a global optimality.

作业

Exercise 1.

设一个随机变量的分布律是 $p_1 \le p_2 \le \cdots \le p_m$，其最优编码具有码长 $\ell_1, \ell_2, \cdots, \ell_m$。试问，能否得出对于所有 i，均有 $\ell_i \le \lceil \log \frac{1}{p_i} \rceil$？

作业

Exercise 2.[王育民(2013)]

令离散无记忆信源

$$U = \begin{Bmatrix} a_1 & a_2 \\ 0.6 & 0.4 \end{Bmatrix}$$

(a)求 U 的最佳二元码、平均码长及编码效率。

(b)求 U^2 的最佳二元码、平均码长及编码效率。

(c)求 U^3 的最佳二元码、平均码长及编码效率。

(d)求 U^4 的最佳二元码、平均码长及编码效率。

作业

Exercise 3.

1) 三元最优编码树应该满足什么条件？

2) **[王育民(2013)]**设离散无记忆信源

$$U = \left\{ \begin{array}{cccccc} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ 0.3 & 0.2 & 0.15 & 0.15 & 0.1 & 0.1 \end{array} \right\}$$

试求其二元和三元 Huffman 编码。

谢谢！