第四章无失真信源编码

第一节编码模型与概念

陈兴同

中国矿业大学 数学学院

2021 年 8 月

内容提要

内容提要

内容提要

为什么要编码

信源发出的消息是由信源符号构成的，这些消息通常都有很大冗余，不能直接用于保存或传输，需要进行压缩编码，以便去除冗余，提高传输或保存的效率。比如根据例子 **??**，若考虑英文字母概率分布及其相互依赖性，信源的熵率最小可达为 1.4bits，即平均每个英文字母只包含 1.4bits 的信息量。按将要学习的信源编码定理，可以找到一种编码方案使得平均每个字母在保存或传输时只占用接近 1.4bits；如果不用编码则每个字母就要占用 5bits，这可能造成存储设备的极大浪费以及传输效率的低下，因此必须进行压缩编码。本章主要学习编码的基本概念、常用的编码方法，以及依据大数定律而划分的典型序列、渐近等分性、无失真信源编码定理。

编码一般原则

对信源序列的编码通常采用分组码，即信源消息序列在进入编码器之前通常被分成一定长度的字符串，然后信源编码器再对这些字符串进行编码，每个字符串对应一个“码字”，不同的码字形成码字串构成信源编码器的输出，称为码字序列。一般分组的长度可能相同也可能不同；码字的长度也可能不同。

信源编码时一般需要先知道信源符号或每组消息发生的概率，这种利用信源概率分布进行编码的方法称为**统计编码**；但事先信源的统计特征可能不知道，在编码时需要对信源字符概率分布进行估计或预测，这种在信源概率分布未知情况下对信源进行编码的方法称为**通用编码**。

信源编码模型图

信源编码流程如图 4-1。

Figure: 图 4-1

分组码定义4.1.1

（1）有一个**信源**，它的字符集为 $\mathcal{X}=\{a_1,a_2,\cdots,a_N\}$，它发出的每个字符形成很长的字符串，称为**消息序列**，将每 n 个字符划分成一组，就构成 n 长消息，用向量 $x^{(n)}=(x_1,x_2,\cdots,x_n)$ 表示，全体 n 长的消息集合是笛卡尔积集合 $\mathcal{X}^n=\{(x_1,x_2,\cdots,x_n)|x_i\in\mathcal{X}\}$。

（2）有一个**编码字符集（码符集）**记为 $\mathcal{U}$，通常取为 D 进字符集 $\mathcal{U}=\{0,1,2,\cdots,D-1\}$，其中每个字符称为**码字符**。一个 k 长的码字是由 k 个码字符构成的，用向量 $u^{(k)}=(u_1,u_2,\cdots,c_k)$ 表示，全体 k 长的码取自笛卡尔积集合 $\mathcal{U}^k=\{(u_1,u_2,\cdots,u_k)|u_i\in\mathcal{U}\}$。

续：分组码定义

（3）有一个信源**编码器** $f:\mathcal{X}^n \to \mathcal{U}^* = \bigcup_{k=1}^{\infty}\mathcal{U}^k$, 它将每个消息 $x^{(n)} = (x_1, x_2, \cdots, x_n)$ 编成**码字** $u^{(k)} = (u_1, u_2, \cdots, u_k)$，码字中码符的个数称为**码长**，不同的 n 长消息对应的码长可能不同；全体消息的编码构成的集合称为**码字集**。

（4）有一个信源**译码器**：$g:\mathcal{U}^* \to \mathcal{X}^n$, 它可以将 k 长的码字 $u^{(k)} = (u_1, u_2, \cdots, u_k)$ 译成一个 n 消息 $x^{(n)} = (x_1, x_2, \cdots, x_n)$，交给信宿。

续：分组码定义

（5）编码器 f 与译码器 g 合起来称为**编码方案**。
（6）**编码方案的误差**是指不能正确译码的消息发生的概率

$$P_e = P\left\{g\left[f(X^{(n)})\right] \neq X^{(n)}\right\} = P\left\{x^{(n)} \in \mathcal{X}^n | g(f(x^{(n)})) \neq x^{(n)}\right\}, \tag{4.1}$$

而称 $1 - P_e$ 为**保真度**。
评判一个编码的优劣，除了保真度外，还可以使用平均码长、码率等指标。

平均码长定义4.1.2

设消息集 $\mathcal{X}^n$ 上有概率分布 $p(x^{(n)})$，并且有码字集 $\mathcal{C}=\{f(x^{(n)})|x^{(n)}\in\mathcal{X}^n\}$，对应的码长集合为 $\{l(x^{(n)})|x^{(n)}\in\mathcal{X}^n\}$，称

$$\bar{L}=\sum_{x^{(n)}}p(x^{(n)})l(x^{(n)})$$

为分组编码映射 f 的**平均码长**，它表示每个 n 长消息序列编码时所需码字符的平均个数；而称

$$R=\frac{\bar{L}}{n}$$

为**编码速率简称码率**，它表示编码时平均每个信源字符所需要占用的码字符数，码率越小表明编码的压缩率越高。

非奇异码定义4.1.3

对信源消息的编码可以分成两类：定长码与变长码。码字长度完全相同的编码称为**定长码**，码字长度不完全相同的编码称为**变长码或不等长码**。

*如果任何两个 n 长消息编成的的码字都不同，这相当于编码函数 f 是一个单映射，即若 $x^{(n)} \neq \tilde{x}^{(n)}$ 就有 $f(x^{(n)}) \neq f(\tilde{x}^{(n)})$，则称这种编码方案为**非奇异码或可逆码**；如果至少两个 n 长消息的码字相同，就称这个编码方案为**奇异码**。*

例题 4.1.1：

考虑如下编码的可译性。

字符	概率	码字	码长
1	1/2	0	1
2	1/4	1	1
3	1/8	00	2
4	1/8	11	2

它是变长码，还是非奇异码，但不是唯一可译码，因为码字序列 001101 既可译成 112212 又可译成 3412，不仿称为模糊序列。若译码器收到这样的码字序列，将无法翻译，因此是无用编码。

有限扩展码定义4.1.4

非奇异码是保证正确译码的必要条件，但不是充分条件，因为许多码字“串”在一起形成码字序列时还有可能是奇异的，许多码字“串”在一起就形成了有限扩展码。

*编码 $f:\mathcal{X}^n\to\mathcal{U}^*$ 的**有限扩展码**是映射*
$f^*:\mathcal{X}^*=\bigcup_{n=1}^{\infty}\mathcal{X}^n\to\mathcal{U}^*$，其中

$$f^*(x_1^{(n)},x_2^{(n)},\cdots,x_m^{(n)})=(f(x_1^{(n)}),f(x_2^{(n)}),\cdots,f(x_m^{(n)})),$$

$$x_i^{(n)}\in\mathcal{X}^*,i=1,\cdots,m,m=1,2,\cdots,$$

换句话说：码 f 的有限扩展码 f^* 就是 f 的任意一串有限长度的码字序列。

唯一可译码定义4.1.5

*如果编码 $f: \mathcal{X}^n \to \mathcal{U}^*$ 的有限扩展码 $f^*: \mathcal{X}^* = \bigcup_{n=1}^{\infty} \mathcal{X}^n \to \mathcal{U}^*$ 是非奇异码，则称编码 f 为**唯一可译码**。*

这样，唯一可译码就是指它的任意一串有限长度的码字序列只能被唯一地译成消息序列，不允许有二义性。唯一可译码才是可以实用的编码。

唯一可译码的判断：

为了判断一个编码 $C_0 = \{c_1, c_2, \cdots, c_m\}$ 是否唯一可译码，现在给出一种判断方法，这需要构造一系列的集合 $C_1, C_2, \cdots, C_i, \cdots$，称为**后缀分解集**。

（1） 在码字集 C_0 中寻找具有前缀关系的码字，即若码字 $c, \tilde{c}$ 满足 $\tilde{c} = cs$，则后缀 $s \in C_1$。

（2） 为了构造 C_2, 在码字集 C_0 与 C_1 中寻找具有前缀关系的串，即若对 $\tilde{c} \in C_1$ 有码字 $c \in C_0$ 满足 $\tilde{c} = cs$，则后缀 $s \in C_2$；或者 $c = \tilde{c}s$，则后缀 $s \in C_2$。

（3） 一般地，构造 C_n 的方法是：在码字集 C_0 与 C_{n-1} 中寻找具有前缀关系的串，若对 $\tilde{c} \in C_{n-1}$ 有码字 $c \in C_0$ 满足 $\tilde{c} = cs$，则后缀 $s \in C_n$；或者 $c = \tilde{c}s$，则后缀 $s \in C_n$。

例题 4.1.2

求编码 $\mathcal{C} = \{101, 00110, 10111, 11001\}$ 的后缀分解集。解：

$\mathcal{C}_0$	$\mathcal{C}_1$	$\mathcal{C}_2$	$\mathcal{C}_3$	$\mathcal{C}_4$	$\mathcal{C}_5$	$\mathcal{C}_6$
101	11	001	10	1	01	$\emptyset$
00110				111	0111	
10111					1001	
11001						

为了求 $\mathcal{C}_5$ 元素，对 $\tilde{c} \in \mathcal{C}_4$ 是否有 $c \in \mathcal{C}_0$ 使 $\tilde{c} = cs$ 或 $c = \tilde{c}s$？显然当 $\tilde{c} = 1$ 时有 $c = 101, 10111, 11001$，故 $s = 01, 0111, 1001$；但当 $\tilde{c} = 111$ 时没有 c 使上面两个式子成立。这样就确定了 $\mathcal{C}_5$。其它的集合类似确定。

唯一可译码的充要条件

定理4.1.1

一个码字集是唯一可译码的充要条件是没有一个后缀分解集中包含码字。

证明可参见 [8,10,18]。

根据定理 4.1.1，例题 4.1.2 中的编码是唯一可译码。

D 进树：

码树是编码过程中的有力直观工具，它是一棵叶子结点都是码字结点的 D 进树。

（1）**D 进树**：根据码符的进位制 D，它是从**根结点**出发，根结点最多可产生 D 个**分枝**以及 D 个**一级结点**，每个一级结点又最多可以产生 D 个分枝以及 D 个**二级结点**，如此下去；再为每个分枝分配一个码字符作为这个分枝的权值；这样的树称为 **D 进树**，如图 4-2a 就是一颗三进树。从根到某结点路径中所有分枝个数称为这个结点的**路长**。

完全树：

（2）**完全树**：如果 D 进树中每个非叶子结点都有 D 个分枝，所有叶子结点路长都相等，则这种树称为**满树**；如果一个 D 进树中每个非叶子结点都正好有 D 个分枝，则称这种树为**完全树**。图 4-3a 就是一颗二进满树，图 4-3b 是三进满树，它们当然也是完全树；图 4-3c，图 4-3d 分别是两棵二进与三进完全树。

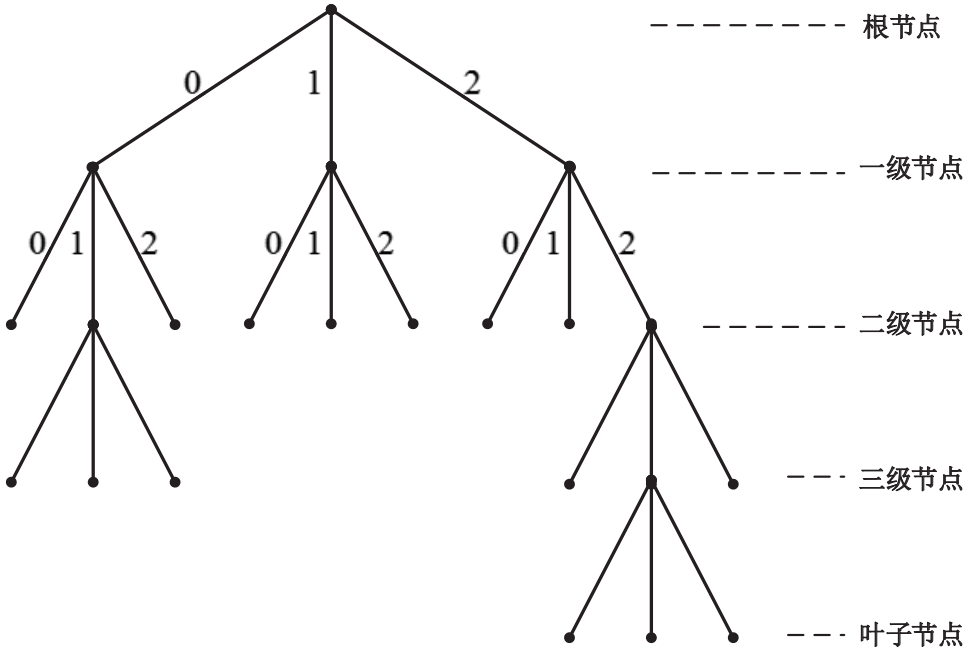
D 进码树：

（3）**D 进码树**：对一个 D 进码字 c，在 D 进满树中从根结点开始寻找分枝上的码字符，若依次排列可以构成码字 c，则最后的那个结点称为**码字结点**。将码字集中每个码字都找到与它对应的码字结点，如果码字结点的后继结点中没有码字结点了，就删除所有后继结点，这样形成的树称为**码树**。比如图 4-2b 是一个 3 进码树，其中的结点 A, B, C, D, E 等等都是码字结点，对应的码字为

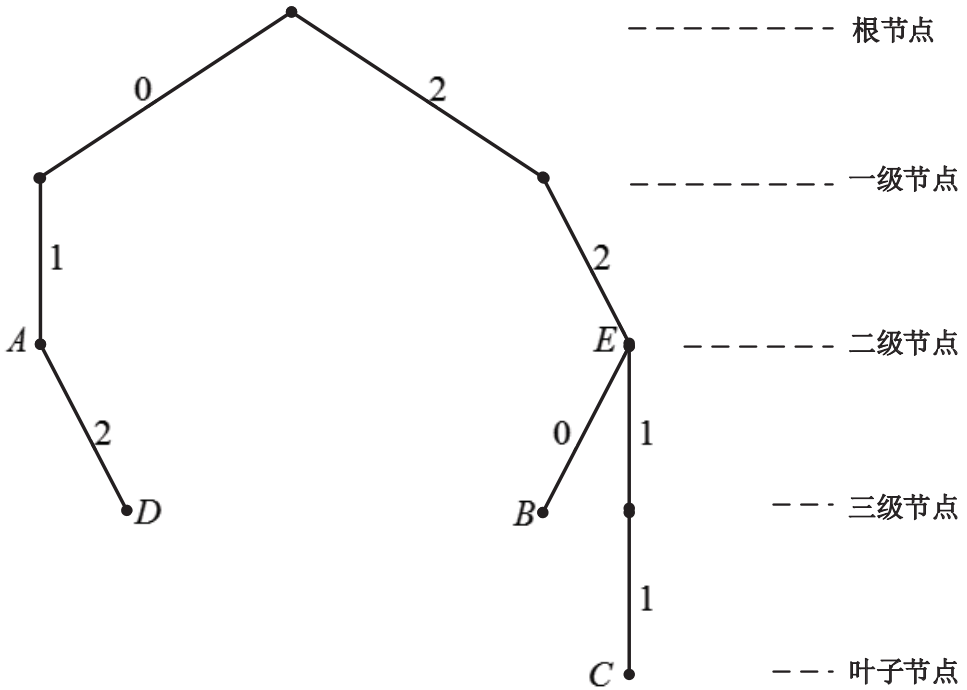
A	B	C	D	E
01	220	2211	012	22

码树也可以是满树，也可以是完全树，也可以是普通的树。

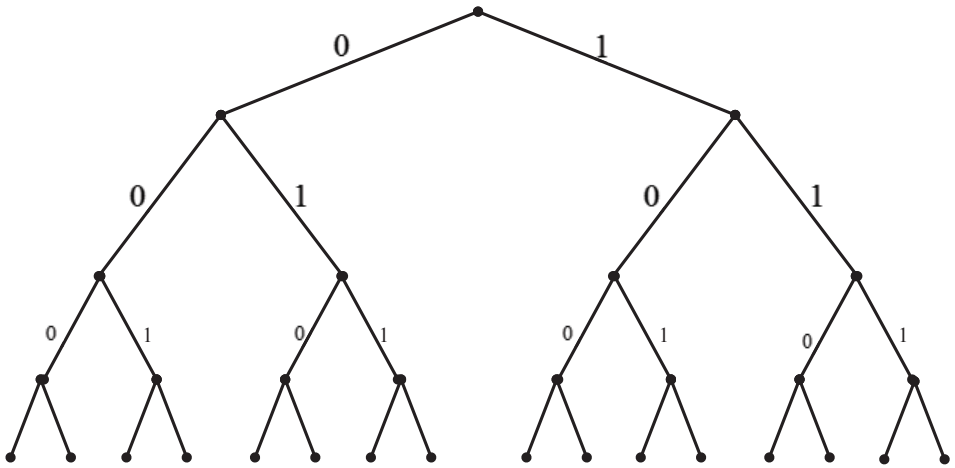
三进树：4-20



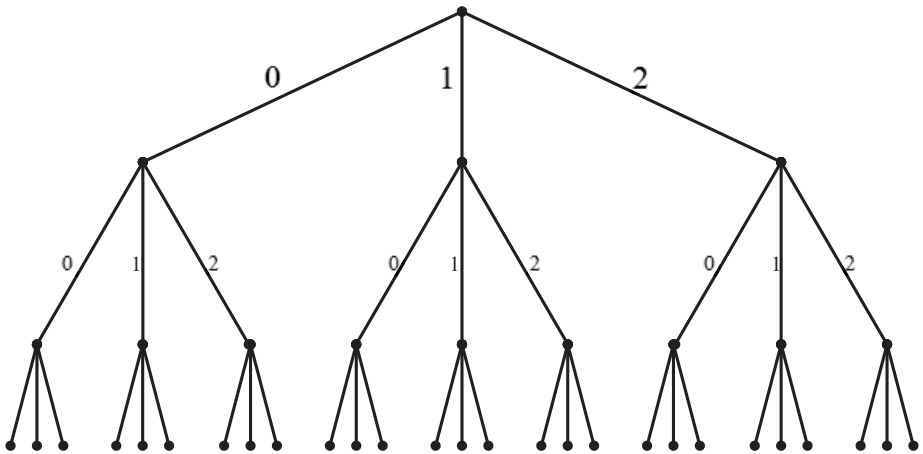
三进码树：4-2b



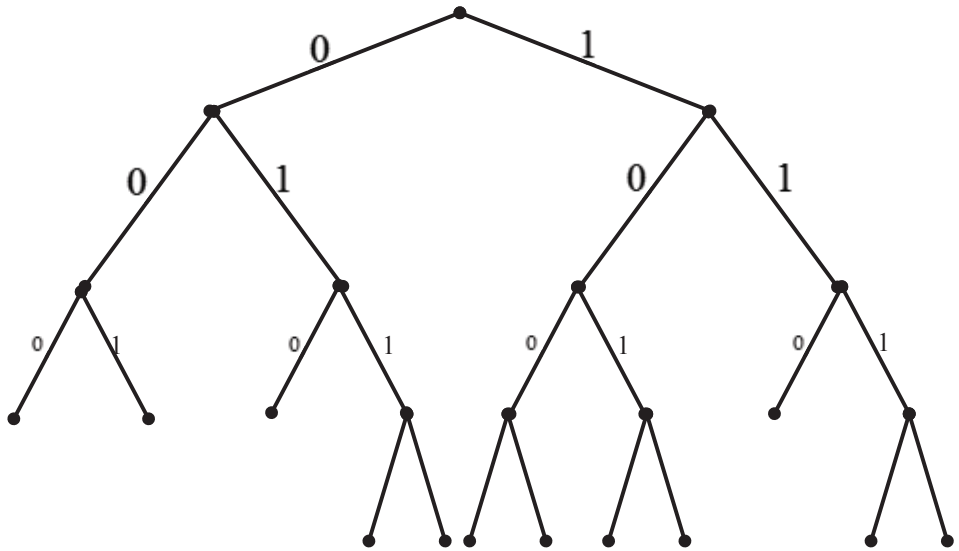
二进完全树：4-3 0



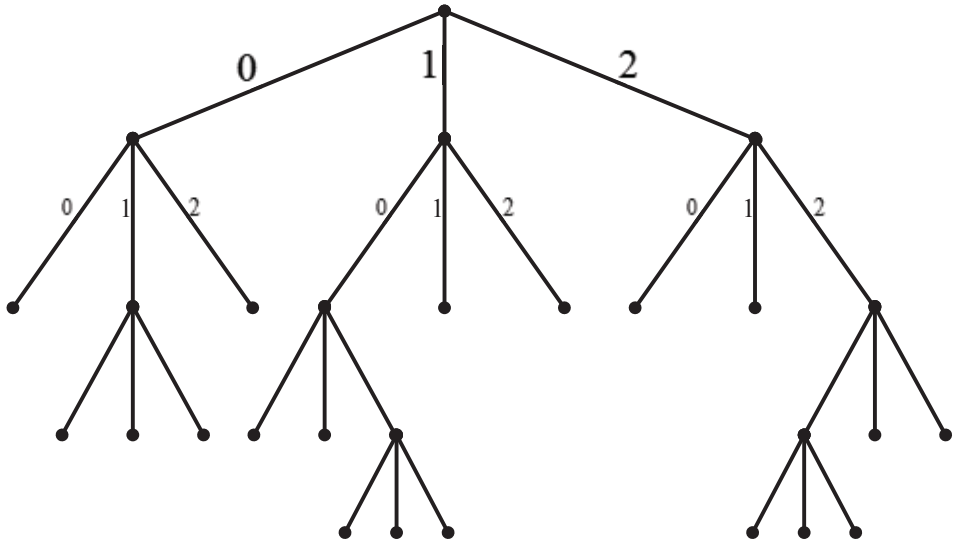
三进完全树:4-36



二进完全树非满树：4-3C



三进完全树非满树：4-3d



引理 4.1.1：

特点

在 D 进码树中

（1） $m(\geq 1)$ 级结点最多有 D^m 个。

（2） 若 $m_1 < m_2$，则某个 m_1 级结点在第 m_2 级结点中的后继结点数最多有 $D^{m_2-m_1}$。

例题 4.1.3

一个二进编码包含四个码字 $C_1 = \{0, 10, 111, 110, 001\}$，试用二进码树来表示。

解：它的码树为图 4-4。它不是一个完全树，更不是满树。

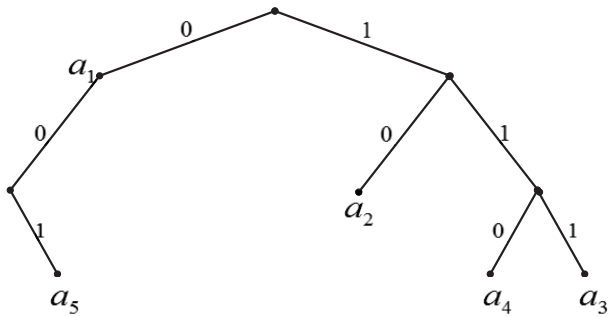


Figure: 4-4