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. 信道编码的基本概念
   - 一般码的定义
   - 一般码的性能参数
   - 一般码的仿真
   - 随机一般码集合
2. 线性分组码
   - 有限域
   - 线性空间
   - 线性分组码
   - 编译码算法
   - 线性分组码的简单例子

一般码的定义

Definition 1 (一般码定义)

设 $\mathcal{A}$ 是一个非空集合，$\mathcal{A}$ 上所有 $n-$重组的全体，记作 $\mathcal{A}^n = \{(a_0, a_1, \cdots, a_{n-1}) | a_i \in \mathcal{A}, 0 \le i < n\}$，也称为 $\mathcal{A}$ 的 $n-$重笛卡儿积。一个码（或码表），记作 $\mathscr{C}(n, M)$，可以表示成一个 $M \times n$ 的阵列

$$\mathscr{C} = \begin{bmatrix} \mathbf{c}^{(0)} \\ \mathbf{c}^{(1)} \\ \vdots \\ \mathbf{c}^{(M-1)} \end{bmatrix},$$

其中 $\mathbf{c}^{(i)} \in \mathcal{A}^n, 0 \le i < M$，称为码字。我们称 $\mathscr{C}$ 是定义在 $\mathcal{A}$ 上的码。

一般码的定义

Example 2

设 $\mathcal{A}=\{0,1\}$，则 $\mathcal{A}$ 上的码通常称为二元码。例如：

$$\mathscr{C}=\begin{bmatrix}0&0&0\\0&1&1\\1&0&1\\1&1&0\end{bmatrix}.$$

若 $\mathcal{A}$ 是有限集合，且 $\mathcal{A}$ 的元素个数 $|\mathcal{A}|>2$，我们称 $\mathcal{A}$ 上的码为多元码。

一般码的定义

Example 3

实数集 $\mathbb{R}$ 上的码有时候也称为星座（constellation），如：

$$\mathscr{C} = \begin{bmatrix} +1 & +1 \\ -1 & +1 \\ -1 & -1 \\ +1 & -1 \end{bmatrix}$$

称为正交相移键控（quadrature phase shift keying，QPSK）信号星座，可以用等间隔分布在平面单位圆上的四个点表示。

一般码的定义

一般码的编译码从概念上来讲是很简单的。编码是从消息集合 $\mathcal{M} = \{0, 1, \cdots, M-1\}$ 到 $\mathscr{C}$ 的一个映射，即 $\phi : \mathcal{M} \longmapsto \mathscr{C}$。若要传送消息 $i \in \mathcal{M}$，编码器输出第 i 个码字 $\mathbf{c}^{(i)}$，即 $\phi(i) = \mathbf{c}^{(i)}$。

一般码的定义

设 $\mathbf{c}^{(i)}$ 经过一个信道之后，接收端收到一个 $n-$重组 $\mathbf{y} \in \mathcal{B}^n$，其中 $\mathcal{B}$ 是信道的输出字符集合。通常情况下，$\mathbf{c}^{(i)}$ 与 $\mathbf{y}$ 之间的对应关系不是确定性的，而是一个“一对多”的对应关系。确切地说，给定 $\mathbf{c}^{(i)}$，$\mathbf{y}$ 是按照某种条件概率分布律分布在接收空间 $\mathcal{B}^n$ 上。一个译码准则就是如何从 $\mathbf{y}$ 推测 $\mathbf{c}^{(i)}$。

从概念上讲，一个译码准则就是把 $\mathcal{B}^n$ 划分成 M 个互不相交的区域，即 $\mathcal{B}^n = \bigcup_{i=0}^{M-1} \mathcal{D}^{(i)}$，且 $\mathcal{D}^{(i)} \bigcap \mathcal{D}^{(j)} = \varnothing, i \neq j$。译码描述为一个映射 $\psi : \mathcal{B}^n \longmapsto \mathcal{M}$。具体地，若 $\mathbf{y} \in \mathcal{D}^{(i)}$，则 $\psi(\mathbf{y}) = i$。有时候，我们也把整个接收空间划分成 $M+1$ 个互不相交的区域 $\mathcal{B}^n = \bigcup_{i=0}^{M} \mathcal{D}^{(i)}$。若接收 $\mathbf{y} \in \mathcal{D}^{(M)}$，则译码器输出“译码失败”。这种情况下，我们称为不完全译码，而之前的称为完全译码。

一般码的定义

上面描述的编译码“算法”从原理上看很简单，但是显然不适合很大的 M。一方面，我们需要足够的空间存储码表；另一方面，我们译码时也要搜索整个码表。

一般码的性能参数

由于接收码字与发送码字之间的依赖关系不是确定性关系，因而译码结果有可能与发送消息不一致。设信道转移概率已知，记为 $P(\mathbf{y}|\mathbf{x}), \forall \mathbf{x} \in \mathcal{A}^n, \forall \mathbf{y} \in \mathcal{B}^n$。注意，若 y 是连续变量，则 $P(y|x)$ 表示条件概率密度，对 y "求和" 可以理解为 "积分"。记 E 是译码错误事件，则给定发送消息是 i 的条件下的译码错误概率是

$$P(E|\text{发送 } i) = \sum_{\mathbf{y} \notin \mathcal{D}^{(i)}} P(\mathbf{y}|\mathbf{c}^{(i)}).$$

若每个消息发送的概率已知，则译码错误概率

$$P(E) = \sum_{i=0}^{M-1} P(\text{发送 } i) \cdot P(E|\text{发送 } i),$$

此概率称为误帧率（frame error rate, FER）。

一般码的性能参数

前面已经看到，一个确定的译码算法对应一个划分。常见的准则之一是最小错误概率译码准则，即寻找一个划分，使得 $P(E)$ 达到最小。而要确定一个划分，就是要对每个 $\mathbf{y} \in \mathcal{B}^n$ 找一个“归宿” $\mathcal{D}^{(i)}$，根据贝叶斯公式，

$$P(\text{发送 } i|\mathbf{y}) = \frac{P(\text{发送 } i) \cdot P(\mathbf{y}|\mathbf{c}^{(i)})}{P(\mathbf{y})}.$$

我们可以计算给定 $\mathbf{y}$ 的条件下的所有后验概率 $P(\text{发送 } i|\mathbf{y}), 0 \le i < M$。

一般码的性能参数

这组概率可以这样直观理解，若我们独立重复地观察所讨论的编码传输系统，则在充分长的观察样本序列中，$\mathbf{y}$ 可能发生了很多次，比如发生了 N 次；在这 N 次中，大致有 $N \cdot P(\text{发送 } i|\mathbf{y})$ 次是对应发送 i 的。因此，为了最小化误码率（最大化正确译码概率），我们应该把 $\mathbf{y}$ 判决为 $\hat{i}$，使得 $P(\text{发送 } \hat{i}|\mathbf{y}) = \max_i P(\text{发送 } i|\mathbf{y})$。

当 $P(\text{发送 } i) = \frac{1}{M}, 0 \leq i < M$ 时，最大后验概率译码可以简化为求 $\hat{i}$，使得 $P(\mathbf{y}|\text{发送 } \hat{i}) = \max_i P(\mathbf{y}|\text{发送 } i)$。这个准则称为最大似然序列译码（maximum likelihood decoding, MLD），其在发送消息等概率时，等价于最大后验概率译码，可以使得误码率最小。在先验概率 $P(\text{发送 } i)$ 未知或者定义不明确时，我们通常采用最大似然序列译码。

当然，若有多个 i，使得 $P(\text{发送 } i|\mathbf{y})$ 达到最大，我们可以按照某个既定规则选择其中一个，或者宣告译码失败。这个译码准则使得误码率达到最小，又称为最大后验概率序列译码（sequence maximum a posteriori decoding, Sequence MAP）。

一般码的性能参数

影响一个码的误码性能的因素很多，其中一个重要的因素是距离特性。

Definition 4

设 $\mathcal{A}$ 是一个非空集合，我们称 $\mathcal{A}$ 为距离空间，是指在 $\mathcal{A}$ 上引入了一个二元实函数 $\rho(x,y)$，满足下列三个条件：对于任意的 $x,y,z\in\mathcal{A}$，
(1) $\rho(x,y)\geqslant 0$，且 $\rho(x,y)=0$ 当且仅当 $x=y$（非负性）；
(2) $\rho(x,y)=\rho(y,x)$（对称性）；
(3) $\rho(x,z)\leqslant\rho(x,y)+\rho(y,z)$（三角不等式：三角形两边之和不小于第三边）。

我们称 ρ 是 $\mathcal{A}$ 上的一个距离。同一个集合上可以根据研究需要，定义不同的距离。若为避免混淆，以 ρ 为距离的距离空间 $\mathcal{A}$ 记作 $(\mathcal{A},\rho)$。

一般码的性能参数

Example 5

对于任意给定的非空集合 $\mathcal{A}$，我们总可以引入汉明（Hamming）距离：

$$\rho(x,y)=\begin{cases}0, & x=y\\ 1, & x\neq y\end{cases},$$

其中 $x,y\in\mathcal{A}$。此时我们称 $(\mathcal{A},\rho)$ 为汉明空间，有时记 ρ 为 d_H。

设 d_H 是 $\mathcal{A}$ 上的汉明距离，则可以定义 $\mathcal{A}^n$ 上的距离：$d_H(\mathbf{x},\mathbf{y})=\sum_{t=0}^{n-1}d(x_t,y_t)$。此距离也称为 $\mathcal{A}^n$ 上的汉明距离。码 $\mathscr{C}$ 的最小汉明距离定义为：$d_{\min}=\min\{d(\mathbf{c}^{(i)},\mathbf{c}^{(j)})|\mathbf{c}^{(i)},\mathbf{c}^{(j)}\in\mathscr{C},i\neq j\}$。

由于任意两个码字至少有 $d_{\min}$ 个位置不同，所以任意两个码字中不可能存在完全相同的 $n-d_{\min}+1$ 位，并且，存在两个码字，其中的 $n-d_{\min}$ 位相同。

一般码的性能参数

我们有如下的命题，称为辛格尔顿（Singleton）界。

Proposition 1

设有限字符集 $\mathcal{A}$ 上的码 $\mathscr{C}(n, M)$ 具有最小汉明距离 $d_{\min}$，则 $M \leq |\mathcal{A}|^{n-d_{\min}+1}$。

证明：从 $\mathscr{C}$ 中任意删除掉 $d_{\min}-1$ 列，则剩余的子阵列仍然具有不同的行。由不同行的个数最多 $|\mathcal{A}|^{n-d_{\min}+1}$ 得证。

一般码的性能参数

给定一个码，我们可以固定一个码字，不妨设为 $\mathbf{c}^{(0)}$，然后考察所有码字与 $\mathbf{c}^{(0)}$ 的汉明距离。这些距离可以用一个母函数的形式记录：

$$A(X) = \sum_{i=0}^{M-1} X^{\rho(\mathbf{c}^{(i)},\mathbf{c}^{(0)})}.$$

这个多项式（合并同类项）的系数也称为 $\mathscr{C}$ 的条件距离谱。形象地说，是站在 $\mathbf{c}^{(0)}$ 的位置四周巡望，看看其他码字离多远。我们也可以考察任何两个码字之间的距离。不同的距离至多有 $\binom{M}{2} = \frac{M(M-1)}{2}$ 种。

一般码的性能参数

Example 6

考虑二元码

$$\mathscr{C} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix},$$

若从 $\mathbf{c}^{(0)} = (0,0,0)$ 的角度看，其汉明距离谱 $A(X) = 1 + 3X^2$，其中，有一个码字距其 0，三个码字距其 2。

Example 7

若一个码是 $n-$维实空间 $\mathbb{R}^n$ 的一个非空子集，我们可以考虑欧几里德（Euclid）距离：$\rho(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{t=0}^{n-1} (x_t - y_t)^2}$。对于欧氏距离空间的码，用平方距离谱取代距离谱更方便。

一般码的性能参数

Example 8

考虑 QPSK 星座，

$$\mathscr{C} = \begin{bmatrix} +1 & +1 \\ -1 & +1 \\ -1 & -1 \\ +1 & -1 \end{bmatrix},$$

若从 $\mathbf{c}^{(0)} = (+1, +1)$ 的角度看，其欧氏平方距离谱 $A(X) = 1 + 2X^4 + X^8$。这个多项式表示有一个星座点（其实就是自身）离 (+1, +1) 的距离为 0，有两个星座点离 (+1, +1) 的平方距离为 4，有一个星座点离 (+1, +1) 的平方距离为 8。

一般码的性能参数

在实际工程中，M 通常可以写成 2^k 的形式。此时，消息集可以看作是 $\{0,1\}^k$。一个 $k-$重二元组，经过编码之后映射成一个 $n-$重码元组。在这种场景下，我们可以定义误比特率（bit error rate, BER）。若发送的消息是 $U \in \{0,1\}^k$，而译码消息是 $\hat{U}$，则 BER 定义为

$$\mathrm{BER} = \frac{\mathbf{E} d_H(U, \hat{U})}{k},$$

其中 $\mathbf{E}$ 表示数学期望。

一般码的仿真

仿真的主要目的是估计 FER 或 BER。仿真模块按照类可以分为三个模块：一个模块是信源—信宿；一个模块是一般码，包括编译码；一个模块是信道。我们建议初学者仅考虑时间离散无记忆信道，其数学描述如下。

若输入一个 $n-$重组 $\mathbf{x} \in \mathcal{A}^n$，则在概率 $P(\mathbf{y}|\mathbf{x})$ 的控制下输出 $\mathbf{y} \in \mathcal{B}^n$，且

$$P(\mathbf{y}|\mathbf{x}) = \prod_{i=0}^{n-1} P(y_i|x_i).$$

在仿真中，我们假定 $P(y_i|x_i)$ 是已知的。更多情况下，我们约定 y_i 与 x_i 之间通过运算式子建立联系，而这个运算过程中有随机数参与。

信源—信宿的角色是产生待发送的数据，比较译码的结果，并统计错误事件。一般码模块要定义码长，定义码表的大小等，也就是要定义阵列 $\mathscr{C}$。此外，一般码模块还要包括编译码函数等。

一般码的仿真

这些模块定义好之后，可以按照下述流程完成仿真。

0. 初始化：num_BE $=0$，num_FE $=0$，tot_Fr $=0$。

1. 循环：

while（tot_Fr $\leq$ max_Fr 且 num_FE $\leq$ max_FE）

  按照均匀分布产生一个消息 $u \in \mathcal{M}$；

  编码得到码字 $\mathbf{c}^{(u)}$；

  经过信道得到 $\mathbf{y}$；

  译码得到 $\hat{u}$；

  tot_Fr $\longleftarrow$ tot_Fr$+1$；

 如果 $\hat{u} \neq u$，

则 num_FE $\longleftarrow$ num_FE$+1$，num_BE $\longleftarrow$ num_BE$+d_H(\hat{u}, u)$。

2. 计算：

$$\text{FER} = \frac{\text{num\_FE}}{\text{tot\_Fr}},$$

$$\text{BER} = \frac{\text{num\_BE}}{\text{tot\_Fr} * k}, \quad \text{其中 } k = \log_2 |\mathcal{M}|.$$

随机一般码集合

一个随机码集合可以描述为一个码表的集合，且其中每个码表均被赋予一个概率（或概率密度）。

Example 9

考虑 $\mathcal{A}=\{0,1\}$ 上的 $(2,2)$ 码集，这样的码表共有 16 个，

$$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}, \cdots, \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}.$$

若我们赋予每个阵列概率 $\frac{1}{16}$，可以得到一个随机码集。当然，我们也可以赋予不同的概率。从概念上讲，任何概率向量 $(p_0, p_1, \cdots, p_{15}), p_i \geq 0, \sum p_i = 1$, 均可以定义随机码集。

随机一般码集合

Example 10

考虑 $\mathbb{R}$ 上的 $(2,2)$ 码集，这样的码表具有 $\begin{pmatrix} c_{00} & c_{01} \\ c_{10} & c_{11} \end{pmatrix}$ 的形式，因而有无穷多个码表。如果我们约定 c_{ij} 是按照正态分布 $\mathcal{N}(0,1)$ 独立产生的，我们同样也得到了一个随机码集。

随机一般码集合

随机码集是 Shannon（香农）研究信道编码定理时引入的重要工具。Shannon 的基本想法可以粗略描述如下。

考虑一个随机码集 (n, M)，给定其中一个特定的码表 $\mathscr{C}$，我们可以定义 FER，这个 FER 可以看作 $\mathscr{C}$ 的函数。进而，我们可以定义码集的平均 FER，即

$$\mathrm{FER} = \sum_{\mathscr{C}} P(\mathscr{C}) \cdot \mathrm{FER}(\mathscr{C}).$$

当 $\mathscr{C}$ 是定义在连续集上时，上面的求和可以换成积分。

一个简单的事实是，对于某个 $\epsilon > 0$，如果可以证明 $\mathrm{FER} \le \epsilon$，则一定有某个特定的码表使得 $\mathrm{FER}(\mathscr{C}) \le \epsilon$。在很多情况下，分析一个具体码的性能比较困难，而给出一个平均 FER 的上界却相对“容易”（当然，对于初学者也不那么容易）。

线性分组码

有限域

设 $\mathbb{F}$ 是一个非空集合，至少包含两个不同的元素。在 $\mathbb{F}$ 上定义两种二元运算，分别记作“$+$”与“$\times$”，即
$+$：$\forall \alpha, \beta \in \mathbb{F}, \exists! \gamma \in \mathbb{F}$（“$\exists!$”表示“存在且唯一”），使得 $\gamma = \alpha + \beta$，称为和。
$\times$：$\forall \alpha, \beta \in \mathbb{F}, \exists! \gamma \in \mathbb{F}$，使得 $\gamma = \alpha \times \beta$，称为积。
为简单起见，我们通常记 α 与 β 的积为 $\alpha \cdot \beta$ 或 $\alpha\beta$。需要指出的是，所谓二元运算是指两个操作数对应一个确定的结果。上述和与积的运算符号“$+$”与“$\times$”只是表示符号，与实数的和运算、积运算或“大相径庭”。

有限域

我们称 $\mathbb{F}$ 为域，是指所定义的二元运算满足以下九条规律：

1. $(\mathbb{F},+)$ 是一个交换群，即
   1.1 交换律：$\forall\alpha,\beta\in\mathbb{F}$，有 $\alpha+\beta=\beta+\alpha$。
   1.2 结合律：$\forall\alpha,\beta,\gamma\in\mathbb{F}$，有 $(\alpha+\beta)+\gamma=\alpha+(\beta+\gamma)$。
   1.3 零元：$\forall\alpha\in\mathbb{F},\exists\theta\in\mathbb{F}$，使得 $\alpha+\theta=\theta+\alpha=\alpha$。
   1.4 负元：$\forall\alpha\in\mathbb{F},\exists\beta$，使得 $\alpha+\beta=\theta$。

可以证明，“零元”是唯一的，通常简记为 0。也可以证明，给定 $\alpha\in\mathbb{F}$，负元 β 也是唯一的，简记为 $-\alpha$（相当于实数中的相反数）。

有限域

2. $(\mathbb{F}\setminus\{0\},\times)$ 是一个交换群，即
   2.1 交换律：$\forall\alpha,\beta\in\mathbb{F}\setminus\{0\}$，有 $\alpha\cdot\beta=\beta\cdot\alpha$。
   2.2 结合律：$\forall\alpha,\beta,\gamma\in\mathbb{F}\setminus\{0\}$，有 $(\alpha\cdot\beta)\cdot\gamma=\alpha\cdot(\beta\cdot\gamma)$。
   2.3 幺元：$\forall\alpha\in\mathbb{F}\setminus\{0\},\exists e\in\mathbb{F}$，使得 $\alpha\cdot e=e\cdot\alpha=\alpha$。
   2.4 逆元：$\forall\alpha\in\mathbb{F}\setminus\{0\},\exists\beta$，使得 $\alpha\cdot\beta=e$。

同样可以证明，“幺元”是唯一的，通常简记为 1。也可以证明，给定 $\alpha\in\mathbb{F}\setminus\{0\}$，逆元 β 也是唯一的，简记为 α^{-1}（相当于实数中的倒数）。

3. 分配律：$\forall\alpha,\beta,\gamma\in\mathbb{F}$，有 $\alpha\cdot(\beta+\gamma)=\alpha\cdot\beta+\alpha\cdot\gamma$。

有限域

Example 11

所有实数构成的集合在通常意义下构成一个域，记作 $\mathbb{R}$。我们常见的例子还有有理数域（记作 $\mathbb{Q}$）、复数域（记作 $\mathbb{C}$）。但是全体整数构成的集合在通常的运算下不构成一个域。考虑如下例子：2 是一个整数，但我们找不到整数 x，使得 $2x = 1$。

通俗地讲，一个域就是可以做“加、减、乘、除”四项基本运算的集合。我们之前遇到的域多是无限集合，而在编码领域还经常用到有限域，即元素个数 $|\mathbb{F}| < \infty$ 的域。

Example 12

最简单的有限域 $\mathbb{F} = \{0, 1\}$，其运算规定如下：

$0 + 0 = 0$，$0 + 1 = 1$，$1 + 1 = 0$；

$0 \cdot 0 = 0$，$0 \cdot 1 = 0$，$1 \cdot 1 = 1$。

通常，对于素数 p，集合 $\mathbb{F} = \{0, 1, \cdots, p-1\}$ 在模 p 意义下做加法与乘法，构成一个域，记作 $\mathbb{F}_p$，也称为素数域。

有限域

Example 13

有限域 $\mathbb{F}_3 = \{0, 1, 2\}$ 的运算可以用下述两个表格定义。

$+$	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

$\times$	0	1	2
0	0	0	0
1	0	1	2
2	0	2	1

同样地，我们可以定义 $\mathbb{F}_5$，$\mathbb{F}_7$，等等。

有限域

Example 14

$\mathbb{Z}_4 = \{0, 1, 2, 3\}$ 在模 4 运算下，不是域。因为 $2 \cdot 2 = 4 = 0 \pmod 4$，这不符合我们之前的认识：即非零元素之积不应该为零。但是按照域的定义，$\mathbb{Z}_4$ 违反了哪一条呢？比如，我们可以断言，找不到 x，使得 $2x = 1$。详细证明略去。

但是，如果我们规定如下运算，

+	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

×	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	3	1
3	0	3	1	2

我们可以验证 $\mathbb{F}_4 = \{0, 1, 2, 3\}$ 构成一个域。

有限域

从上面的例子来看，域不仅与集合有关，还与集合上定义的运算有关。对于有限域，有如下优美的结论。

Proposition 2

当且仅当 $q = p^m$（p 是素数，m 是正整数）时，存在（在同构意义下是唯一存在的）有限域 $\mathbb{F}_q$。

注意，当 $m > 1$ 时，$\mathbb{F}_{p^m}$ 的运算法则不是简单的模 q 运算。有限域由于其元素个数有限，有许多组合计数的问题，具有简单而美的结构。

线性空间

设 $\mathbb{F}$ 是一个域，考虑其 $n-$重笛卡儿积 $\mathbb{F}^n$。我们在 $\mathbb{F}^n$ 上定义如下运算，称为向量加法。

对于 $\mathbf{a} = (a_1, a_2, \cdots, a_n) \in \mathbb{F}^n, \mathbf{b} = (b_1, b_2, \cdots, b_n) \in \mathbb{F}^n$，定义 $\mathbf{c} = \mathbf{a} + \mathbf{b} = (c_1, c_2, \cdots, c_n)$，其中 $c_t = a_t + b_t, 1 \leq t \leq n$。

我们也可以定义一个称为数乘的运算。对于 $\lambda \in \mathbb{F}, \mathbf{a} = (a_1, a_2, \cdots, a_n) \in \mathbb{F}^n$，定义数乘 $\lambda \cdot \mathbf{a} = (\lambda a_1, \lambda a_2, \cdots, \lambda a_n)$，也简记为 $\lambda\mathbf{a}$。

向量加法可以看作是“信号叠加”，而数乘可以看作是“信号缩放”，这两个运算都是线性运算。

线性空间

我们可以验证上述定义的两个运算满足如下八条规律。因此，$\mathbb{F}^n$ 是一般线性空间的一个特例。

1. 向量加法：
   1.1 交换律：$\mathbf{a}+\mathbf{b}=\mathbf{b}+\mathbf{a}$。
   1.2 结合律：$(\mathbf{a}+\mathbf{b})+\mathbf{c}=\mathbf{a}+(\mathbf{b}+\mathbf{c})$。
   1.3 零向量：$\mathbf{a}+\mathbf{0}=\mathbf{0}+\mathbf{a}=\mathbf{a}$。
   1.4 负向量：$\forall\mathbf{a},\exists\mathbf{b}$，使得 $\mathbf{a}+\mathbf{b}=\mathbf{0}$。
2. 数乘
   2.1 规范性：$1\cdot\mathbf{a}=\mathbf{a}$。
   2.2 累积性：$(\lambda_1\cdot\lambda_2)\mathbf{a}=\lambda_1\cdot(\lambda_2\mathbf{a})$。
3. 分配律
   3.1 $\lambda(\mathbf{a}+\mathbf{b})=\lambda\mathbf{a}+\lambda\mathbf{b}$。
   3.2 $(\lambda_1+\lambda_2)\mathbf{a}=\lambda_1\mathbf{a}+\lambda_2\mathbf{a}$。

线性空间

下面我们着重考虑 $\mathbb{F}_q^n$，即有限域上的 n−重组全体构成的线性空间。在此约定下，我们不仅可以考虑一般线性空间中的维数、线性相关、线性无关、秩、极大线性无关组等概念，还可以计数。

Definition 15

$\mathbb{F}_q^n$ 的一个非空子集 U，若在向量加法与数乘下封闭，则称 U 为一个线性子空间，即 $\forall \mathbf{a}, \mathbf{b} \in U$，有 $(\mathbf{a} - \mathbf{b}) \in U$；$\forall \mathbf{a} \in U, \lambda \in \mathbb{F}$，有 $\lambda \cdot \mathbf{a} \in U$。

线性空间

Example 16

考虑有限域 $\mathbb{F}_2=\{0,1\}$ 及其上定义的线性空间 $\mathbb{F}_2^3$，这个线性空间包括 8 个向量 $\{\mathbf{v}=(v_1,v_2,v_3), v_i\in\mathbb{F}_2\}$。这个空间的自然基是 $\mathbf{e}_1=(0,0,1),\mathbf{e}_2=(0,1,0),\mathbf{e}_3=(1,0,0)$，就是说：

1. $\mathbf{e}_1,\mathbf{e}_2,\mathbf{e}_3$ 线性无关；
2. 任何 $\mathbb{F}_2^3$ 中的向量都可以由 $\mathbf{e}_1,\mathbf{e}_2,\mathbf{e}_3$ 线性表出。

我们可以验证，$\mathbf{a}_1=(0,0,1),\mathbf{a}_2=(0,1,1),\mathbf{a}_3=(1,1,1)$ 也满足上面两条性质，因而也是一组基。

线性分组码

Definition 17

设 $\mathbb{F}$ 是一个有限域。一个维数是 k，长度是 n 的线性分组码，记作 $\mathscr{C}[n,k]$，是 $\mathbb{F}^n$ 的一个 k 维线性子空间。

线性分组码

Example 18

我们可以验证，

$$\mathscr{C} = \{(0,0,0),(0,1,1),(1,0,1),(1,1,0)\}$$

中的向量在向量加法与数乘运算下封闭，因而构成一个线性子空间，也称为一个码，记作 $\mathscr{C}[3,2]$，因其是二维。

线性分组码

既然 $\mathscr{C}[n, k]$ 是线性子空间，则存在一组基。设 $\{\mathbf{g}_0, \mathbf{g}_1, \cdots, \mathbf{g}_{k-1}\}$ 是这样一组基，构造一个矩阵

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_0 \\ \mathbf{g}_1 \\ \vdots \\ \mathbf{g}_{k-1} \end{bmatrix},$$

这个矩阵称为生成矩阵，缘由如下：任何一个码字 $\mathbf{c} \in \mathscr{C}[n, k]$，总可以找到一组元素 $u_0, u_1, \cdots, u_{k-1} \in \mathbb{F}$，使得 $\mathbf{c} = u_0\mathbf{g}_0 + u_1\mathbf{g}_1 + \cdots + u_{k-1}\mathbf{g}_{k-1}$，即 $\mathbf{c} = \mathbf{u}\mathbf{G}$。

线性分组码

对于给定的线性分组码，我们可以定义与之对偶的码。

Definition 19

设 $\mathscr{C}[n,k]$ 是一个线性分组码，我们记 $\mathscr{C}^{\perp}$ （“$\perp$”表示垂直）是与之对偶的码，其中 $\mathbf{x}\in\mathscr{C}^{\perp}$ 当且仅当 $\mathbf{x}$ 与所有 $\mathscr{C}$ 中的码字正交，即 $\forall \mathbf{c}\in\mathscr{C}, \sum_{i=0}^{n-1} x_i c_i = 0$。

需要指出的是，上述“点积”形式是域 $\mathbb{F}$ 上的运算。可以验证，对偶码自身也是线性分组码，维数是 $n-k$。设

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_0 \\ \mathbf{h}_1 \\ \vdots \\ \mathbf{h}_{(n-k-1)} \end{bmatrix},$$

是对偶码的生成矩阵，则称其为原码的校验矩阵。

线性分组码

综上所述，对于线性分组码 $\mathscr{C}[n,k]$，我们至少有两种描述，即

$$\begin{aligned}\mathscr{C} &= \{\mathbf{c}|\mathbf{c} = \mathbf{uG}, \mathbf{u} \in \mathbb{F}^k\},\\ &= \{\mathbf{c}|\mathbf{c} = \mathbf{Hc}^{\mathsf{T}} = \mathbf{0}, \mathbf{c} \in \mathbb{F}^n\}.\end{aligned}$$

其中，“T”表示转置。注意，一个码可有不同形式的生成矩阵与校验矩阵。

编译码算法

线性分组码的编码算法比较简单，设 $\mathscr{C}[n,k]$ 是一个线性分组码，$\mathbf{G}$ 是一个生成矩阵。编码可以表示成一个线性映射

$$\varphi : \mathbb{F}^k \to \mathbb{F}^n$$

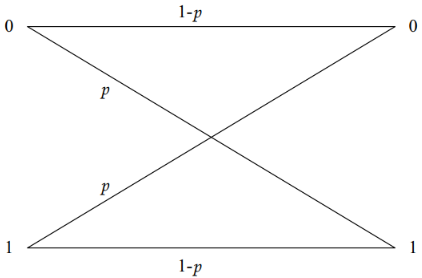
使得，对于给定的 $\mathbf{u} \in \mathbb{F}^k$，有唯一码字 $\mathbf{c} = \mathbf{uG}$ 与之对应。这种一般的编码算法的复杂度与 $\mathbf{G}$ 的稀疏程度有关。在实际工程中，常用的是系统编码方法，此情形对应生成的矩阵 $\mathbf{G}$ 具有形式 $\mathbf{G} = [\mathbf{I}\ \mathbf{P}]$，其中 $\mathbf{I}$ 是 k 阶单位矩阵，而 $\mathbf{P}$ 是 $k \times (n-k)$ 的矩阵。系统码有个校验矩阵，形式为 $[-\mathbf{P}^{\mathsf{T}}\ \mathbf{I}]$。对于系统编码，码字具有形式 $\mathbf{c} = (\mathbf{u}, \mathbf{uP})$。就是说信息向量 $\mathbf{u}$ “原封不动”地出现在码字中。我们有如下命题。

Proposition 3

任何一个线性分组码 $\mathscr{C}[n,k]$（必要时经过分量置换），均存在一个系统形式的生成矩阵。

编译码算法

线性分组码的译码算法需要结合信道模型来讨论。我们首先考虑二进制对称信道（binary symmetry channel, BSC）模型，如下图所示：

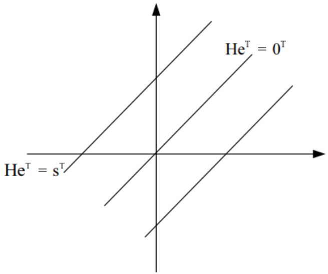


编译码算法

二进制对称信道是无记忆信道，输入是 $\{0,1\}$，以 $p(<\frac{1}{2})$ 的概率发生错误。设 $\mathbf{c}=(c_0,c_1,\cdots,c_{n-1})$ 是发送码字，则接收向量 $\mathbf{r}=(r_0,r_1,\cdots,r_{n-1})$ 可以表示成 $\mathbf{r}=\mathbf{c}+\mathbf{e}$，其中 $\mathbf{e}$ 称为错误图样向量，“$+$”是模 2 运算。我们说信道是 BSC 是指，$\mathbf{e}$ 是一个独立同分布的随机二进制向量的样本，其分量取 1 的概率是 p。在接收端，当收到 $\mathbf{r}$ 之后，我们可以计算 $\mathbf{S}=\mathbf{H}\mathbf{r}^{\mathsf{T}}$，其中 $\mathbf{H}$ 是校验矩阵。由于 $\mathbf{H}\mathbf{c}^{\mathsf{T}}=\mathbf{0}$，我们可以得到 $\mathbf{S}=\mathbf{H}\mathbf{e}^{\mathsf{T}}$，我们称 $\mathbf{S}$ 是伴随式。如果 $\mathbf{S}=\mathbf{0}$，我们认为没有错误发生。若 $\mathbf{S}\neq\mathbf{0}$，则一定有错误发生。利用这个性质，我们可以实现检错。而纠错是指，我们想从 $\mathbf{S}=\mathbf{H}\mathbf{e}^{\mathsf{T}}$ 中解出 $\mathbf{e}$。这是一个线性方程组，包含有 $n-k$ 个方程，n 个未知数 $\{e_i,0\leq i\leq n-1\}$。这个方程组有 2^k 个解，译码就是从中选择一个解。

编译码算法

对于一般的线性方程组 $\mathbf{He}^\mathsf{T}=\mathbf{s}^\mathsf{T}$，通常的解法是求出齐次线性方程组 $\mathbf{He}^\mathsf{T}=\mathbf{0}^\mathsf{T}$ 的所有解，然后再求出一个特解（有可能不存在，但在译码这里一定是存在的）。那么，$\mathbf{He}^\mathsf{T}=\mathbf{s}^\mathsf{T}$ 的所有解就可以表示为特解 + 齐次线性方程组通解的形式。解方程组的过程也可以从几何的角度去描述。齐次线性方程组 $\mathbf{He}^\mathsf{T}=\mathbf{0}^\mathsf{T}$ 的解空间是一个线性子空间，在这里实际上就是线性分组码 $\mathscr{C}$ 本身，含有 2^k 个码字。而一般的线性方程组 $\mathbf{He}^\mathsf{T}=\mathbf{s}^\mathsf{T}$ 的解空间相当于把 $\mathscr{C}$ 进行了一个平移，如下图示意。



编译码算法

所以，若 $\mathbf{He}^{\mathrm{T}}=\mathbf{s}^{\mathrm{T}}$ 有解的话，与 $\mathbf{He}^{\mathrm{T}}=\mathbf{0}^{\mathrm{T}}$ 的解的个数是一样多的，同为 2^k 个。现在的问题是，如何从这 2^k 个解中挑选出一个作为译码器的输出。
这个挑选规则与信道特性紧密相关，在 BSC 信道条件下，一个错误图样 $\mathbf{e}$ 发生的概率是

$$P(\mathbf{e})=p^{W_{\mathbf{H}}(\mathbf{e})}(1-p)^{n-W_{\mathbf{H}}(\mathbf{e})}$$

其中，$W_{\mathbf{H}}(\mathbf{e})$ 表示 $\mathbf{e}$ 的汉明重量。可以看出，$P(\mathbf{e})$ 是 $W_{\mathbf{H}}(\mathbf{e})$ 的减函数（注意 $p<\frac{1}{2}$）。由此，若 $\mathbf{e}_1$ 与 $\mathbf{e}_2$ 是两个解，但 $W_{\mathbf{H}}(\mathbf{e}_1)<W_{\mathbf{H}}(\mathbf{e}_2)$，我们应该选择哪一个呢？一个合理的选择是 $\mathbf{e}_1$，因为它较 $\mathbf{e}_2$ 发生的机会大，这就是最小汉明距离译码。在 BSC 信道条件下等价于最大似然译码。对于码参数比较小的码，我们通常列一个表，给出 $\mathbf{s}$ 与 $\mathbf{e}$ 之间的对应关系。当然，若某个 $\mathbf{s}$ 对应的解中有两个 $\mathbf{e}$，它们的重量相等，且同为最轻，则我们可以随意选择一个 $\mathbf{e}$，或者报告一个译码失败的信息。

线性分组码例子

Example 20 (重复码)

设 $u \in \mathbb{F}_2$ 是一个待传比特，一个简单的编码是 $u \longrightarrow \mathbf{c} = (u, u, \cdots, u)$，即把 u 重复 n 次。这个码的生成矩阵是 $\mathbf{G} = [1, 1, \cdots, 1]_{1\times n}$，对应的校验矩阵是 $\mathbf{H} = [\mathbf{1}^T\ \mathbf{I}]$，即

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 0 & \cdots & 0 \\ 1 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \cdots & 1 \end{bmatrix}_{(n-1)\times n}.$$

线性分组码例子

该码的码率是 $\frac{1}{n}$。若重复码在 BSC 信道上使用，则可用大数逻辑译码，即在接收端，根据一个码字中 1 的个数与 0 的个数的多少来判决。若这个数目相等（当 n 是偶数时才有可能相等），则宣告译码失败。
设 BSC 的错误概率是 p，则大数逻辑译码的错误概率是

$$P_b = \sum_{t \geq \lceil \frac{n}{2} \rceil} \binom{n}{t} p^t (1-p)^{n-t}$$

可以证明，当 $p < \frac{1}{2}$ 时，P_b 随着 n 的增大趋于零，可以达到“可靠”通信。但是，要付出的代价是码率 $\frac{1}{n} \to 0$，这是通信系统要避免的。

线性分组码例子

Example 21 (奇偶校验码)

设 $\mathbf{u} = [u_0, u_1, \cdots, u_{n-2}] \in \mathbb{F}_2^{n-1}$ 是一个待传信息，奇偶校验码的编码算法是计算 $c_{n-1} = \sum_{i=0}^{n-i} u_i$，对应码字是 $\mathbf{c} = [u_0, u_1, \cdots, u_{n-2}, c_{n-1}]$，其中 $\mathbf{u}$ 称为信息位，c_{n-1} 称为奇偶校验位。

奇偶校验码的码率是 $\frac{n-1}{n}$，生成矩阵是 $\mathbf{G} = [\mathbf{I}\ \mathbf{1}^T]_{(n-1)\times n}$，校验矩阵是 $\mathbf{H} = [\mathbf{1}]_{1\times n}$。
可以验证，码长是 n 的重复码与奇偶校验码互为对偶码。前者的最小汉明距离是 n，而后者的最小汉明距离是 2。因而奇偶校验码在 BSC 上不具有纠错能力，但是可以检错。事实上，奇偶校验码可以发现任何奇数个错误。这两个码分别记为 $\mathscr{C}[n, 1, n]$ 和 $\mathscr{C}[n, n-1, 2]$。

线性分组码例子

Example 22 (汉明码)

前两个例子是从编码的角度引出的，定义码的同时也定义了编码算法。汉明码比较方便的定义是从校验矩阵出发。设 $m > 1$ 是一个整数，考虑 $\mathbb{F}_2^m$ 中的非零向量，共有 $2^m - 1$ 个。以它们为列，构成一个 $m \times (2^m - 1)$ 的矩阵。

线性分组码例子

$$
\mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 1 & 1 & \cdots & 1 \\ 1 & 0 & 1 & \cdots & 1 \end{bmatrix}.
$$

该矩阵的秩是 m。因此，$\mathscr{C} = \{\mathbf{c} | \mathbf{H}\mathbf{c}^{\mathsf{T}} = 0\}$ 具有参数：码长 $2^m - 1$，维数 $2^m - 1 - m$。这样定义的码就是汉明码。汉明码的最小汉明距离是 3，可以纠正一个位错误，论证如下。

Proposition 4

一个线性分组码的最小汉明距离是 $d_{\min}$，当且仅当其校验矩阵 $\mathbf{H}$ 的任意 $d_{\min} - 1$ 列线性无关且存在某 $d_{\min}$ 列线性相关。

线性分组码例子

从汉明码的定义可以看出，$\mathbf{H}$ 的任何两列不同，所以最小距离至少为 3。由于可以找到三列相加等于 0，我们由上述命题知道 $d_{\text{min}} = 3$。假定错误图样 $\mathbf{e}$ 的重量为 1，即只有一位错误，则 $\mathbf{S} = \mathbf{H}\mathbf{e}^{\mathsf{T}}$ 刚好是错误位置对应的列。因此，可由 $\mathbf{S}^{\mathsf{T}}$ 找出错误的位置。为帮助理解，取 $m = 3$ 为例，这就是常常用来作为例子的 [7,4,3] 汉明码，其校验矩阵是

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

设一个码字 $\mathbf{c}$，经过一个 BSC 信道，收到 $\mathbf{y}$，计算 $\mathbf{S}^{\mathsf{T}} = \mathbf{H}\mathbf{y}^{\mathsf{T}}$。

若 $\mathbf{S}^{\mathsf{T}} = \mathbf{0}^{\mathsf{T}}$，则我们认为无错；若 $\mathbf{S}^{\mathsf{T}} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$，我们认为第一位发生了错误。

作业

Exercise 1. [田宝玉(2008)]
设二元码为 $C_b = \{11100, 01001, 10010, 00111\}$。
(1)计算此码的最小距离 d_{min}。
(2)计算此码的码率，假定码字等概分布。
(3)采用最大似然译码准则，当通过二元对称信道传输，接收序列分别为 10000, 01100 和 00100 时，应分别译成什么码字?

作业

Exercise 2. [田宝玉(2008)]
一个二元对称信道的转移概率矩阵为 $\mathbf{P} = \begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix}$, $(p < 1/2)$，信道输入符号 0，1 的概率分别为 ω，$1-\omega$。
(1)求利用MAP准则的判决函数和平均错误率。
(2)求利用ML准则的判决函数和平均错误率。
(3)什么情况下,上述两准则的判决结果相同?

作业

Exercise 3. [王育民(2013)]
给出 $\mathbb{F}_7$ 的加法和乘法运算表，并找出每个非零元素的逆元素。

Exercise 4. [田宝玉(2008)]
一个线性分组码的校验矩阵为

$$\mathbf{H} = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \end{pmatrix}$$

求该码的生成矩阵和码的最小距离。

谢谢！