UNIVERSITAT JAUME I

Grado en Matemática Computacional

Trabajo Final de Grado

Una introducción a los códigos correctores de errores

Autor:
Elena Cantero López

Tutor académico:
Carlos Galindo Pastor

Fecha de lectura: 24 de Julio de 2023
Curso académico 2022/2023

Agradecimientos

A mi tutor, Carlos Galindo, por su ayuda e implicación en la realización del trabajo. A mis padres y a mi hermana por su apoyo y paciencia.

Gracias.

Resumen

La teoría de corrección de errores se inició en el año 1948 para minimizar los errores producidos cuando se envía una gran cantidad de información. En este trabajo mostraremos los aspectos más elementales de esta teoría, incluyendo una introducción a la decodificación por síndromes y al problema principal de la teoría de la codificación lineal.

Palabras clave

Códigos correctores de errores; síndrome; problema principal de la codificación.

Keywords

Error-correcting codes; syndrome; the main problem of coding theory.

Índice general

Capítulo 1

Introducción

El artículo de Shannon [29] puede considerarse como el trabajo científico que produjo el nacimiento de la teoría de códigos correctores de errores. Los *códigos correctores de errores* son una herramienta imprescindible cuando se envía una gran cantidad de información por medios electrónicos. Estos códigos almacenan tanto la información a ser enviada, como información extra para corregir los errores que se producen debido al ruido del canal de transmisión o de los mecanismos que manejan la información. Una de las primeras utilidades de estos códigos se produjo a la hora de enviar fotos desde el espacio, ya que, el canal (el espacio) es ruidoso. La aparición en las últimas décadas de multitud de dispositivos que almacenan y envían grandes ficheros hace imprescindible el uso de estos códigos que garantizan la fiabilidad de estos dispositivos de uso diario (teléfonos móviles, ordenadores, aplicaciones que manipulan fotos, etc.).

La llegada de la primera computadora cuántica no ha hecho perder el interés en la corrección de errores sino que lo ha aumentado, ya que las computadoras cuánticas cometen, de momento, muchos más errores que las computadoras clásicas.

En este trabajo, hacemos una introducción muy elemental a la teoría de códigos correctores de errores. Nuestro propósito es únicamente explicar su utilidad y algunos de sus comportamientos y problemas asociados básicos. En nuestra redacción hemos considerado únicamente objetos simples para construir códigos correctores, como cuerpos finitos y espacios vectoriales sobre ellos. También explicamos las técnicas más básicas para el codificado y decodificado. Además estudiamos algunos aspectos elementales del principal problema de la teoría de códigos correctores.

Esta memoria sólo pretende dejar al lector listo para profundizar en esta interesante teoría. Nosotros hemos utilizado básicamente la referencia [11]. Para profundizar en el estudio de estos códigos algunos textos de interés son [32], [27], [26], [20], [14] y [17].

Capítulo 2

Primeras nociones sobre códigos

2.1. Códigos

En este trabajo vamos a estudiar algunos aspectos elementales de los códigos correctores de errores, así como su origen e importancia.
Los *códigos correctores de errores* se emplean para la corrección de errores en mensajes que se transmiten a través de canales con ruido. Los canales de transmisión pueden ser: una línea telefónica o una comunicación vía satélite, entre muchos otros. Algunos ejemplos de ruido son aquellas distorsiones producidas por interferencias, un error humano o un material defectuoso.
La función de estos códigos es añadir al mensaje original una redundancia que permita, al receptor, reconstruir el mensaje correctamente aunque este contenga errores. Este proceso de modificación del mensaje original se denomina codificación. Así mismo, cuando se recibe el mensaje y se retira la redundancia, se está realizando un proceso de decodificación.
Las palabras-código de un código binario son simplemente sucesiones finitas de 0s y 1s. En estos casos la redundancia incluida en las palabras-código consiste en añadir un número adecuado de 0s y 1s dependiendo del mensaje. Con esta redundancia hacemos que la probabilidad de que el receptor reconstruya el mensaje original sea mayor. Por tanto, lo más conveniente sería que las palabras-código tengan poca similitud entre ellas, ya que así es menos probable confundirlas debido a los errores introducidos por el ruido.
Definimos un código q-ario como una secuencia de símbolos pertenecientes a un conjunto $F_q = \{\lambda_1, \lambda_2, ..., \lambda_q\}$ con q elementos. Por ende, si aplicamos esta notación, el código binario sería equivalente a un código 2-ario. Si las palabras-código tienen una medida fija de n símbolos, se dice que pertenecen a un código de bloque de longitud n. Por tanto podemos decir que $(F_q)^n$ contiene todas las sucesiones posibles de n elementos del conjunto F_q. Y además su cardinal sería q^n.
Vamos a definir con cierto detalle la distancia de la que usualmente se habla cuando queremos

ver la separación entre varias palabras-código. Esta es la distancia de Hamming, que es la que vamos a utilizar a lo largo de este trabajo y la denotaremos como $d_H(x, y)$ siendo x e y dos palabras-código (también llamadas vectores).

Definición 1. Dados dos vectores x e y en $(F_q)^n$, la distancia de Hamming entre ellos $(d_H(x, y))$ es el número de coordenadas en que difieren.

La distancia de Hamming es una métrica porque cumple:

1. $d_H(x, y) = 0$ si y solo si $x = y$.
2. $d_H(x, y) = d_H(y, x)$ para todo $x, y \in (F_q)^n$.
3. $d_H(x, y) \leq d_H(x, z) + d_H(z, y)$ para todo $x, y, z \in (F_q)^n$ (desigualdad triangular).

Un código q-ario de bloque de longitud n es un subconjunto $C \subseteq (F_q)^n$. Hemos indicado que para codificar una palabra de un mensaje se le añade redundancia. Veamos ahora el proceso de decodificación. Si se envía la palabra-código $x \in C$ y se recibe y, ésta se puede decodificar de distintas maneras. En este trabajo vamos a utilizar la decodificación llamada de *vecino más cercano*. Si proponemos x' en C como el mensaje decodificado, se trata de encontrar aquel $x' \in C$ tal que la $d_H(y, x')$ es la más pequeña posible de elementos de C a y. Esto maximizará la efectividad, es decir, la probabilidad de que se corrijan los errores si se cumplen ciertas condiciones en el canal. En un canal adecuado la probabilidad de cada símbolo de ser recibido incorrectamente es la misma y es $< \frac{1}{2}$. Además, una vez se reciba un símbolo por error, la probabilidad de que el resto de símbolos sean erróneos será la misma.
En el caso de un código binario, si suponemos que p es la probabilidad de que un símbolo se envíe erróneamente, entonces la probabilidad de que se envíe correctamente será $1 - p$. Por tanto, si la longitud del mensaje es n, la probabilidad de que no haya ningún error será $(1-p)^n$, de igual manera, la probabilidad de que todos los símbolos se reciban erróneamente será p^n. Así, la probabilidad de que el vector tenga errores en i posiciones específicas es $p^i(1 - p)^{n-i}$.

A continuación vamos a introducir un parámetro que ayuda a medir la eficacia en la corrección de errores dentro de un código C. Este parámetro es la distancia mínima $(d(C))$ que es la menor distancia entre todas las palabras-código existentes en el conjunto C. Matemáticamente se denota como:

$$d(C) = \text{mín}\{dH(x, y)|x, y \in C, x \neq y\}.$$

A partir de esta definición tenemos el siguiente teorema.

Teorema 1. *Sea $C \subseteq (F_q)^n$ un código de bloque.*

1. *C puede detectar hasta t errores en cualquier palabra-código si $d(C) \geq t + 1$.*
2. *C puede corregir hasta e errores en cualquier palabra-código si $d(C) \geq 2e + 1$.*

Demostración.

1. Vamos a suponer que $d(C) \geq t+1$, entonces queremos demostrar que dicho código C puede detectar hasta t errores. Supongamos que se envía una palabra-código x que contiene una

cantidad de errores $\leq t$. Sabemos que $d(C)$ es la mínima distancia entre cualquiera de las palabras-código de C y que $d(C) \geq t+1$. Entonces, como solo hay t errores o menos, el vector recibido y será distinto a cualquier palabra-código. Por tanto, el error podrá ser detectado.

2. Suponemos que $d(C) \geq 2e+1$, que una palabra-código x se envía y que se recibe el vector y, donde hay un número de errores $\leq e$. Por tanto, la distancia de Hamming entre x e y, $d_H(x,y)$, es menor o igual que e. Si tomamos x', una palabra-código distinta de x, entonces $d_H(x',y)$ es mayor o igual que $e+1$. En efecto, si $d_H(x',y)$ fuera menor o igual que e, resultaría que por la desigualdad triangular tendríamos que $d(x,x') \leq d_H(x,y) + d_H(x',y) \leq 2e$. Esto contradice el hecho de que $d(C)$ es mayor o igual que $2e+1$. Así que x es la palabra-código más cercana a y.

□

Ahora veremos un corolario donde se indica cuantos errores se pueden detectar y corregir en un código donde se conoce su distancia mínima.

Corolario 1. *Si la distancia mínima de un código C es d, entonces C puede ser usado para:*

1. *Detectar hasta $d-1$ errores.*
2. *Corregir hasta $\lfloor (d-1)/2 \rfloor$ errores en cualquier palabra-código, donde $\lfloor . \rfloor$ indica parte entera por debajo.*

Demostración.

1. Para probar 1 basta observar que $d \geq t+1$ si, y solo si, $t \leq d-1$.
2. El apartado 2 se deduce a partir de que $d \geq 2e+1$ si, y solo si, $e \leq (d-1)/2$.

□

2.2. Parámetros de un código

A continuación, y en el resto del trabajo, diremos que un código C tiene parámetros (n, M, d). Donde n representa la longitud de cada elemento del código, es decir, el número de símbolos que formarán cada palabra-código; M simboliza el número de palabras-código que contiene C y d la distancia mínima entre las palabras-código de C distintas.
Teniendo en cuenta la notación anterior, un buen código debe tener una longitud n pequeña para hacer la transmisión de mensajes más rápida, un número de palabras M grande para permitir la transmisión de una gran variedad de mensajes y una distancia mínima d grande para poder corregir una cantidad elevada de errores.

Ahora trataremos el problema principal en teoría de códigos correctores de errores, donde el objetivo es conseguir optimizar uno de los parámetros cuando tenemos los valores de los otros dos. En base a esto damos una nueva notación: $A_q(n,d)$, que representa el número máximo M de palabras de un código q-ario de longitud n y distancia mínima d prefijada. Este problema se resuelve fácilmente para $d=1$ y $d=n$, lo vamos a ver en el siguiente teorema.

Teorema 2. *Con la notación anterior, se cumple:*

1. $A_q(n,1) = q^n$.
2. $A_q(n,n) = q$.

Demostración.

1. Para que la distancia mínima de un código sea al menos 1, se debe cumplir que las palabras-código que lo forman sean distintas. Por tanto, el código de mayor M será un código q-ario de parámetros $(n, M, 1)$, es decir, el conjunto $(F_q)^n$ donde $M = q^n$.
2. Supongamos que C es un código q-ario con parámetros (n, M, n), entonces 2 palabras-código cualesquiera de C difieren en n posiciones, es decir, todas las posiciones contienen símbolos distintos. En consecuencia, se cumple que $A_q(n,n) \leq q$. Por otro lado, sabemos que un código q-ario de repetición es un código con parámetros (n, q, n), es decir, $A_q(n,n) = q$.

□

Definición 2. Dos códigos q-arios son equivalentes si uno se puede obtener a partir del otro por combinación de operaciones de los siguientes tipos:

1. Permutaciones de las posiciones del código.
2. Permutaciones de los símbolos apareciendo en una posición fija.

Para comprender mejor las operaciones anteriores vamos a suponer que tenemos el código dispuesto en una matriz de tamaño $M \times n$ cuyas filas son las distintas palabras-código. Entonces, el primer tipo de operaciones consiste en la permutación de las columnas de la matriz, mientras que el segundo tipo consiste en re-etiquetar los símbolos de una columna dada. Mediante estos cambios conseguimos seguir teniendo la misma distancia entre las palabras-código y, por tanto, obtener otro código con los mismos parámetros (n, M, d). En definitiva, conseguimos corregir el mismo número de errores.

Lema 1. *Cualquier código q-ario con parámetros (n,M,d) sobre un alfabeto {0,1,...,q-1} es equivalente a un código de parámetros (n,M,d) que contiene el vector* $0 := (0, 0, \ldots, 0)$.

Demostración. Elegimos unas palabras-código $x_1 x_2 \cdots x_n$ cualesquiera y para cada $x_i \neq 0$ aplicamos la permutación:

$$\begin{pmatrix} 0 & x_i & j \\ \downarrow & \downarrow & \downarrow \\ x_i & 0 & j \end{pmatrix} \text{ para todo } j \neq 0, x_i$$

a los símbolos en la posición i. □

Los siguientes resultados afectan solo a códigos binarios sobre F_2. Aquí nuestro alfabeto F_2 es el cuerpo finito $\mathbb{F}_2$ del que hablaremos luego. Las operaciones $+, \cdot$ vienen dadas de la forma siguiente: $0 + 0 = 1 + 1 = 0$; $0 + 1 = 1 + 0 = 1$; $0 \cdot 0 = 1 \cdot 0 = 0 \cdot 1 = 0$ y $1 \cdot 1 = 1$.

Lema 2. *Si x e y pertenecen al espacio vectorial* $(\mathbb{F}_2)^n$*, entonces* $d_H(x,y) = w(x+y)$*, donde w(x) denota el peso del vector x y representa el número de 1s que hay en la palabra-código x.*

Demostración. La suma $x+y$ produce como resultado un 1 en las posiciones donde los símbolos de x e y difieren y un 0 donde son iguales (ambos 0 o ambos 1). □

En el siguiente resultado, dados $x = (x_1, \ldots, x_n)$ e $y = (y_1, \ldots, y_n) \in (\mathbb{F}_2)^n$, escribiremos $x \cap y = (x_1y_1, \ldots, x_ny_n) \in (\mathbb{F}_2)^n$.

Lema 3. *Si $x, y \in (\mathbb{F}_2)^n$, entonces $d_H(x, y) = w(x) + w(y) - 2w(x \cap y)$.*

Demostración. Sabemos que $d_H(x, y) = w(x + y)$, es decir, la distancia de Hamming entre x e y es la suma de números 1s que hay en x más la suma de números 1s que hay en y menos el número de posiciones en las que ambos contienen un 1. Por tanto, se puede escribir de la siguiente manera: $d_H(x, y) = w(x) + w(y) - 2w(x \cap y)$. □

Teorema 3. *Suponemos que d es un entero positivo impar. Entonces, existe un código binario de parámetros (n, M, d), si, y solo si, existe un código binario de parámetros $(n+1, M, d+1)$.*

Demostración.
$\Longleftarrow$ Suponemos que C es un código binario con parámetros (n, M, d), donde d es impar. Sea C' el código de longitud $n + 1$ obtenido de C extendiendo cada palabra-código x de C a x' según la siguiente regla:

$$x = x_1x_2...x_n \Longrightarrow x' = \begin{cases} x_1x_2 \ldots x_n0 \text{ si } w(x) \text{ es par.} \\ x_1x_2 \ldots x_n1 \text{ si } w(x) \text{ es impar.} \end{cases}$$

Equivalentemente, podemos definir x' = $x_1x_2 \ldots x_nx_{(n+1)}$, donde

$$x_{(n+1)} = \sum_{i=1}^{n} x_i$$

calculada con módulo 2.
Esta construcción de C' se crea a partir de C y consiste en añadir una verificación de paridad general al código C. Como $w(x')$ es par para toda palabra-código x' de C', se obtiene del Lema 3 que $d_H(x', y')$ es par para todo x', y' en C'. Como consecuencia, $d(C')$ es par, además, claramente $d \leq d(C') \leq d + 1$, y como d es impar, tenemos que $d(C') = d + 1$. Por tanto, C' es un código con parámetros $(n + 1, M, d + 1)$, donde d es impar.

$\Longrightarrow$ Sea D un código con parámetros (n+1, M, d+1) donde d es impar. Elegimos x e y palabras-código de D tales que $d_H(x, y) = d+1$. Después elegimos una posición en la que x e y difieren y la eliminamos de todas las palabras-código. Obtenemos como resultado un código de parámetros (n, M, d). □

Una consecuencia inmediata del resultado anterior es el siguiente corolario.

Corolario 2. *Si d es impar, entonces $A_2(n + 1, d + 1) = A_2(n, d)$. Equivalentemente, si d es par, entonces $A_2(n, d) = A_2(n - 1, d - 1)$.*

Definición 3. Dados n y m dos enteros tales que, $0 \leq m \leq n$, el coeficiente binomial n sobre m, se representa de la siguiente forma $\binom{n}{m}$ y se define como sigue:

$$\binom{n}{m} = \frac{n!}{m!(n-m)!}.$$

Lema 4. *Si tenemos un conjunto que contiene n elementos, el número de selecciones desordenadas de m objetos distintos que se pueden extraer de este es* $\binom{n}{m}$.

Demostración. Una selección ordenada de m objetos distintos de un conjunto que contiene n elementos se puede crear de $n(n-1)\cdots(n-m+1) = \frac{n!}{n-m}$ maneras distintas.
El primer objeto se puede elegir de entre n símbolos, el segundo debe ser distinto del primero, por tanto tiene $n-1$ elementos posibles para escoger y así sucesivamente. Como se trata de una selección desordenada, debemos tener en cuenta que hay $m(m-1)\cdots 2\cdot 1 = m!$ maneras de ordenar los m objetos elegidos, así pues, el número de selecciones desordenadas es $\frac{n!}{m!(n-m)!}$. □

Definición 4. Para cualquier elemento u en $(F_q)^n$ y cualquier entero $r \geq 0$, la esfera de radio r y centro u, denotada por $S(u,r)$, es el conjunto $\{v \in (F_q)^n | d_H(u,v) \leq r\}$.

Lema 5. *Una esfera de radio r en* $(F_q)^n$ $(0 \leq r \leq n)$ *contiene exactamente* $\binom{n}{0} + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \ldots + \binom{n}{r}(q-1)^r$ *vectores.*

Demostración. Sea u un elemento fijo en $(F_q)^n$. Nos interesa saber cuantos elementos v distan exactamente m unidades de u, donde $m \leq n$. Las m posiciones donde v difiere de u pueden ser elegidas de $\binom{n}{m}$ maneras, y en cada una de estas posiciones la entrada de v puede ser elegida de $q-1$ maneras para ser diferente que la entrada de u. Por tanto, el número de vectores que están a una distancia exacta de m unidades de u son $\binom{n}{m}(q-1)^m$ y así, el número total de vectores en $S(u,r)$ es $\binom{n}{0} + \binom{n}{1}(q-1) + \cdots + \binom{n}{r}(q-1)^r$. □

Nota 1. *Los números* $\binom{n}{m}$ *se llaman coeficientes binomiales por su rol en el teorema del binomio, el cual para cualquier entero positivo n muestra que:*

$$(1+x)^n = 1 + \binom{n}{1}x + \binom{n}{2}x^2 + \cdots + \binom{n}{n}x^n.$$

Teorema 4. *(Empaquetamiento esférico o cota de Hamming). Un código q-ario de parámetros (n,M,2t+1) satisface la siguiente condición:*

$$M\left[\binom{n}{0} + \binom{n}{1}(q-1) + \cdots + \binom{n}{t}(q-1)^t\right] \leq q^n. \tag{2.1}$$

Demostración. Suponemos que C es un código q-ario de parámetros $(n, M, 2t+1)$. Como hemos podido ver, dos esferas cualesquiera de radio t centradas en distintas palabra-código pueden no tener vectores en común. Por tanto, el número total de vectores en las M esferas de radio t centradas en las M palabras-código viene dado por la parte izquierda de 2.1. Este número debe ser menor o igual a q^n, que es el número total de vectores en $(F_q)^n$. □

Definición 5. Un código que alcanza la cota del empaquetamiento esférico, es decir, que produce la igualdad en la inecuación 2.1, se denomina perfecto. Por lo tanto, en un código, t, de corrección de errores t perfecto, las M esferas de radio t centradas en las palabras-código 'llenan' todo el espacio $(F_q)^n$ sin superponerse. En otras palabras, todo vector en $(F_q)^n$ está a una distancia menor que t de una palabra-código. El código de repetición binaria

$$\begin{cases} 00\ldots0 \\ 11\ldots1 \end{cases}$$

de longitud n, donde n es impar, es un código perfecto de parámetros $(n, 2, n)$. Estos códigos, junto a los códigos que contienen solo una palabra-código o que son el conjunto completo $(F_q)^n$, se conocen como los códigos perfectos triviales. El problema de encontrar todos los códigos perfectos ha supuesto para los matemáticos uno de los mayores retos de la teoría del código.

Definición 6. Un diseño de bloques equilibrados consiste en un conjunto S de v elementos, llamados puntos o variedades, y una colección de b subconjuntos de S, llamados bloques. De tal manera que para valores fijos enteros positivos k, r y λ se tienen las siguientes características.

1. Cada bloque contiene exactamente k puntos.
2. Cada punto aparece en r bloques exactamente.
3. Cada par de puntos aparecen juntos en λ bloques exactamente.

Por tanto un diseño S se suele llamar un (b, v, r, k, λ)-diseño.

Ahora vamos a ver un ejemplo. Si tomamos $S = \{1, 2, 3, 4, 5, 6, 7\}$ y tenemos los siguientes subconjuntos de S: $\{1, 2, 4\}$, $\{2, 3, 5\}$, $\{3, 4, 6\}$, $\{4, 5, 7\}$, $\{5, 6, 1\}$, $\{6, 7, 2\}$, $\{7, 1, 3\}$. Es fácil verificar que cada par de elementos de S aparecen juntos exactamente en un bloque. Entonces, los subconjuntos forman los bloques de un $(7, 7, 3, 3, 1)$-diseño.

Definición 7. La matriz de incidencia $A = [a_{ij}]$ de un diseño de bloques es una matriz de tamaño $v \times b$, donde las filas corresponden a las variedades $x_1, x_2, \ldots, x_v$ y las columnas a los bloques $B_1, B_2, \ldots, B_b$ y cuyas entradas (i, j) están definidas por:

$$a_{ij} = \begin{cases} 1 \text{ si } x_i \in B_j \\ 0 \text{ si } x_i \notin B_j \end{cases}$$

Veremos ahora un ejemplo. Sea A la matriz de incidencia de la figura 2.2 y sea B la matriz de dimensión 7×7 obtenida a partir de A, reemplazando los 0s por 1s y los 1s por 0s. Entonces tenemos C, el código de longitud 7, cuyas 16 palabras-código son las filas $a_1, a_2, \ldots, a_7$ de la matriz A, las filas $b_1, b_2, \ldots, b_7$ de la matriz B y los vectores adicionales $0 = 0000000$ y

$1 = 1111111$. Por tanto, C es un (7, 16, 3)-código.

		Bloques						
		B_1	B_2	B_3	B_4	B_5	B_6	B_7
	1	1	0	0	0	1	0	1
	2	1	1	0	0	0	1	0
Variedades	3	0	1	1	0	0	0	1
	4	1	0	1	1	0	0	0
	5	0	1	0	1	1	0	0
	6	0	0	1	0	1	1	0
	7	0	0	0	1	0	1	1

(2.2)

2.3. Introducción a los cuerpos finitos

Para facilitar el uso y análisis de los códigos correctores de errores es conveniente usar una estructura algebraica. Esto es especialmente útil para tener un alfabeto en el cual es posible sumar, restar, multiplicar y dividir sin restricción. En otras palabras, queremos darle a F_q una estructura de cuerpo. Veamos la definición.

Definición 8. Un cuerpo $\mathbb{F}$ es un conjunto de elementos con dos operaciones: $+$ (llamada adición) y $\cdot$ (multiplicación) que satisface las siguientes propiedades.

1. $\mathbb{F}$ es cerrado para $+$ y $\cdot$, es decir, si $a, b \in \mathbb{F}$ entonces $a + b$ y $a \cdot b \in \mathbb{F}$.
2. Leyes conmutativas: $a + b = b + a$, $a \cdot b = b \cdot a$.
3. Leyes asociativas: $(a + b) + c = a + (b + c)$, $a \cdot (b \cdot c) = (a \cdot b) \cdot c$.
4. Leyes distributivas: $a \cdot (b + c) = a \cdot b + a \cdot c$.
5. $a + 0 = a$ para todo $a \in \mathbb{F}$.
6. $a \cdot 1 = a$ para todo $a \in \mathbb{F}$.
7. Para cada $a \in \mathbb{F}$ existe un elemento inverso para la suma o adición $(-a) \in \mathbb{F}$ tal que $a + (-a) = 0$.
8. Para cada $a \neq 0$ en $\mathbb{F}$, existe un elemento inverso de la multiplicación $a^{-1} \in \mathbb{F}$ tal que $a \cdot a^{-1} = 1$.

Las dos siguientes propiedades de un cuerpo se deducen fácilmente de la definición.

Lema 6. *En cualquier cuerpo $\mathbb{F}$ se cumplen las siguientes propiedades:*

1. *$a0 = 0$ para todo $a \in \mathbb{F}$.*
2. *$ab = 0$ implica que $a = 0$ o $b = 0$. Así, el producto de dos elementos de un cuerpo, que sean distintos de 0, será distinto de 0.*

Demostración.

1. Tenemos $a0 = a(0+0) = a0 + a0$. Añadiendo el inverso respecto de la suma de $a0$ en ambos sitios obtenemos: $0 = a0 + (-a0) = a0 + a0 + (-a0) = a0 + 0 = a0$. Por ello, $a0 = 0$.
2. Supongamos que $ab = 0$. Si $a \neq 0$, entonces a tiene un inverso para la multiplicación y, en consecuencia, $b = 1 \cdot b = (a^{-1}a)b = a^{-1}(ab) = a^{-1}0 = 0$. Por tanto, $ab = 0$ implica $a = 0$ o $b = 0$.

□

Definición 9. Un conjunto de elementos con dos propiedades internas + y · que satisface las propiedades de cuerpos desde la (1) hasta la (7), pero no necesariamente la (8) se denomina anillo.

Nota 2. Por comodidad, emplearemos la palabra anillo para referirnos a anillo conmutativo o abeliano con la identidad.

Definición 10. Un cuerpo finito es un cuerpo que tiene un número finito de elementos. A este número se le llama orden del cuerpo. El siguiente resultado fundamental sobre cuerpos finitos fue demostrado por Evariste Galois (1811-1832).

Teorema 5. *Existe un cuerpo de orden q si y solo si q es una potencia de un número primo. Además, si q es una potencia de un número primo, entonces solo hay un cuerpo de ese orden.Un cuerpo de orden q a menudo se denomina cuerpo de Galois de orden q y se denota como $GF(q)$. La demostración de este teorema se puede encontrar en [16].*

Definición 11. Sea m un número entero positivo fijo. Dos enteros a y b se llaman congruentes módulo m, simbolizado por $a \equiv b \mod m$, si $a - b$ es divisible por m, es decir, si $a = km + b$ para algún entero k. Escribimos $a \not\equiv b \mod m$ si a y b no son congruentes módulo m. Cada número entero, cuando se divide por m, tiene un residuo (o resto) principal único igual a uno de los números enteros en el anillo de congruencia $\mathbb{Z}_m$ $\{0, 1, \ldots, m-1\}$. Se demuestra fácilmente que dos números enteros son congruentes módulo m si, y solo si, tienen los mismos residuos principales en la división por m.

Teorema 6. *Suponemos que $a \equiv a' \mod m$ y $b \equiv b' \mod m$. Entonces:*

1. $a + b \equiv a' + b' \mod m$.
2. $ab \equiv a'b' \mod m$.

Demostración. Sabemos que $a = a' + km$ y $b = b' + lm$ para algunos enteros k y l. Entonces:

1. $a + b = a' + b' + (k + l)m$ y así $a + b \equiv a' + b' \mod m$.
2. $ab = a'b' + (kb' + a'l + klm)m$ y por ello, $ab \equiv a'b' \mod m$.

El Teorema 6 permite calcular congruencias sin trabajar con números grandes. Obsérvese, que si $a \equiv a' \mod m$, entonces el uso repetido de (2) muestra que, para todo entero positivo n, $a^n \equiv (a')^n \mod m$. □

Teorema 7. *$\mathbb{Z}_m$ es un cuerpo si, y solo si, m es un número primo.*

Demostración. Primero, suponemos que m no es primo. Entonces $m = ab$ para algunos enteros positivos $a, b < m$. Entonces $ab \equiv 0 \mod m$, con $a \not\equiv 0 \mod m$ y $b \not\equiv 0 \mod m$.
En consecuencia, en $\mathbb{Z}_m$ el producto de los elementos distintos de 0, a y b, es 0 y, por tanto, por el Lema 6, se puede afirmar que $\mathbb{Z}_m$ no es un cuerpo.

Ahora suponemos que m es primo. Para demostrar que $\mathbb{Z}_m$ es un cuerpo, es suficiente mostrar que todo elemento distinto de 0 perteneciente a $\mathbb{Z}_m$ tiene un inverso respecto a la multiplicación. Tomamos a un elemento distinto de 0 en $\mathbb{Z}_m$ y consideramos los $m-1$ elementos $1a, 2a, \ldots, (m-1)a$. Estos elementos son distintos de 0, porque los elementos ia no pueden tener al número primo m como divisor si i y a no lo tienen. Además, los elementos son distintos unos de otros porque $ia = ja$ implica que $(i-j)a \equiv 0 \mod m$ lo que obliga a que m sea un divisor de $(i-j)a$ y por ello m es un divisor de $i-j$, ya que m es primo y no divide a a. En consecuencia, $i = j$, ya que ambos $i, j \in \{1, 2, \ldots, m-1\}$.

Así, que en $\mathbb{Z}_m$ los $m-1$ elementos $1a, 2a, \ldots, (m-1)a$ deben coincidir como conjunto con los elementos $1, 2, \ldots, m-1$, y uno de ellos, ja, debe ser igual a 1. Entonces j será el inverso de a. □

2.4. Espacios vectoriales sobre cuerpos finitos

En este apartado vamos a asumir que q es una potencia de un número primo y denotamos por $\mathbb{F}_q$ ó $GF(q)$ el cuerpo finito de q elementos, a los que llamaremos escalares. El conjunto $GF(q)^n$ de todas las n-tuplas ordenadas sobre $GF(q)$ se denotará por $V(n, q)$ y sus elementos se llamarán vectores. Definimos dos operaciones dentro de $V(n, q)$:

1. Suma de vectores: si $x = (x_1, x_2, \ldots, x_n)$ e $y = (y_1, y_2, \ldots, y_n) \in V(n, q)$ entonces $x + y = (x_1 + y_1, x_2 + y_2, \ldots, x_n + y_n)$.
2. Multiplicación de un vector por un escalar: si $x = (x_1, x_2, \ldots, x_n) \in V(n, q)$ y $a \in GF(q)$ entonces $ax = (ax_1, ax_2, \ldots, ax_n)$.

Se puede demostrar fácilmente que $V(n, q)$ satisface los axiomas de espacio vectorial sobre el cuerpo $\mathbb{F}_q$. Es decir, para todo $u, v, w \in V(n, q)$ y para todo $a, b \in GF(q)$ se cumple que:

1. $u + v \in V(n, q)$.
2. $(u + v) + w = u + (v + w)$.
3. El vector formado por todo 0s, $0 = (0, 0, \ldots, 0) \in V(n, q)$ y satisface que $u + 0 = 0 + u = u$.
4. Dado $u = (u_1, u_2, \ldots, u_n) \in V(n, q)$, el elemento $-u = (-u_1, -u_2, \ldots, -u_n) \in V(n, q)$ y satisface $u + (-u) = 0$.
5. $u + v = v + u$.
6. $av \in V(n, q)$.
7. $a(u + v) = au + av$, $(a + b)u = au + bu$.

8. $(ab)u = a(bu)$.
9. $1u = u$, donde 1 es la identidad multiplicativa de $GF(q)$.

Un subconjunto de $V(n,q)$ se llama subespacio vectorial de $V(n,q)$ si es un espacio vectorial bajo la suma y multiplicación escalar definidas en $V(n,q)$. El conjunto trivial 0 y el espacio completo $V(n,q)$ son subespacios vectoriales de $V(n,q)$. Un subespacio vectorial se llama no trivial si contiene al menos un vector distinto de 0.

Teorema 8. *Un conjunto $C \neq \emptyset$ incluido en $V(n,q)$ es un subespacio vectorial de $V(n,q)$ si y sólo si C es cerrado bajo la suma y la multiplicación escalar, es decir, si y sólo si C satisface estas dos condiciones:*

1. *Si $x, y \in C$ entonces $x + y \in C$.*
2. *Si $a \in GF(q)$ y $x \in C$ entonces $ax \in C$.*

Demostración. Se demuestra fácilmente que si C satisface los axiomas 1 y 2 del Teorema 8 de espacio vectorial, entonces C satisface todos los axiomas 1-9 de la subsección 2,4 (con $V(n,q)$ reemplazado por C) para un espacio vectorial. Para mostrar que $0 \in C$, elegimos cualquier $x \in C$, entonces, por (2), $0 = 0x$ y esto, por tanto, pertenece a C. La propiedad (2) también muestra que si $v \in C$, entonces $-v \in C$, para $-v = (-1)v$. □

Una combinación lineal de r vectores $v_1, v_2, \ldots, v_r$ pertenecientes a $V(n,q)$, es un vector de la forma $a_1v_1 + a_2v_2 + \cdots + a_rv_r$, donde los elementos a_i representan escalares.
Un conjunto de vectores $\{v_1, v_2, \ldots, v_r\}$ es linealmente dependiente si existen los escalares no todos cero, $a_1, a_2, \ldots, a_r$ tales que

$$a_1v_1 + a_2v_2 + \cdots + a_rv_r = 0.$$

Por tanto, diremos que los vectores $v_1, v_2, \ldots, v_r$ son linealmente independientes, si cuando se cumple

$$a_1v_1 + a_2v_2 + \cdots + a_rv_r = 0,$$

entonces $a_1 = a_2 = \cdots = a_r = 0$.
Sea C un subespacio vectorial de $V(n,q)$. Un subconjunto de C,

$$\{v_1, v_2, \ldots, v_r\}$$

recibe el nombre de conjunto generador de C si todo vector de C puede ser expresado como una combinación lineal de $v_1, v_2, \ldots, v_r$. Un conjunto linealmente independiente que además es un conjunto generador de C recibe el nombre de base de C.

Teorema 9. *Suponemos que C es un subespacio vectorial no trivial de $V(n,q)$. Entonces cualquier conjunto generador de C contiene una base de C.*

Demostración. Suponemos que $\{v_1, v_2, ..., v_r\}$ es un conjunto generador de C. Si es linealmente dependiente, entonces existen escalares $a_1, a_2, ..., a_r$, al menos uno distinto de 0, tales que:

$$a_1v_1 + a_2v_2 + ... + a_rv_r = 0.$$

Si tomamos a_j distinto de 0 entonces

$$v_j = -(a_j)^{-1} \sum_{i=1, i \neq j}^{r} a_i v_i$$

y, por tanto, v_j es una combinación lineal de otras v_i. De tal manera v_j es redundante y puede ser omitida del conjunto $\{v_1, v_2, ..., v_r\}$ para tener un conjunto generador de C más pequeño. Entonces, podemos omitir generadores redundantes, uno cada vez, hasta que consigamos un conjunto generador linealmente independiente. Como se utiliza un conjunto finito, el proceso debe finalizar. □

Como cualquier subespacio vectorial C de $V(n, q)$ contiene un conjunto generador finito, por el Teorema 9 se deduce que todo subespacio vectorial no trivial tiene base. Una base puede ser pensada como un conjunto generador minimal, que no contiene ningún generador redundante.

Teorema 10. *Suponemos que $\{v_1, v_2, \ldots, v_k\}$ es una base de un subespacio vectorial C de $V(n, q)$. Entonces:*

1. *Todo vector de C puede ser expresado de manera única como una combinación lineal de vectores de la base.*
2. *El subespacio C contiene exactamente q^k vectores.*

Demostración.

1. Suponemos que un vector x de C se representa de dos maneras como una combinación lineal de $v_1, v_2, \ldots, v_k$. Concretamente de las siguientes formas:

$$x = a_1 v_1 + a_2 v_2 + \ldots + a_k v_k,$$

$$x = b_1 v_1 + b_2 v_2 + \ldots + b_k v_k.$$

   Entonces $(a_1 - b_1)v_1 + (a_2 - b_2)v_2 + \cdots + (a_k - b_k)v_k = 0$. Pero el conjunto $\{v_1, v_2, \ldots, v_k\}$ es linealmente independiente y $a_i - b_i = 0$ para $i = 1, 2, \ldots, k$. Es decir, $a_i = b_i$ para $i = 1, 2, \ldots, k$.

2. Por la demostración del primer apartado, los q^k vectores $\sum_{i=1}^{k} a_i v_i$ tales que (a_i pertenece a $GF(q)$) son precisamente los vectores distintos de C.

□

Del Teorema 10 se deduce que cualesquiera dos bases de un subespacio C contienen el mismo número de vectores. Por ello, $|C| = q^k$, y este número k se llama la dimension del subespacio C, se denota por $\dim(C)$. Ya hemos mostrado una base de $V(n, q)$ que tiene n vectores, por lo que $\dim(V(n, q)) = n$.

En este apartado se ha escrito utilizando información extraída de las referencias: [1], [8], [15], [19], [20], [22], [24], [31] y [33].

Capítulo 3

Introducción a los códigos lineales

3.1. Códigos lineales

En este apartado asumiremos que el alfabeto F_q es el cuerpo de Galois $\mathbb{F}_q = GF(q)$, donde q es una potencia de un número primo, y consideraremos el espacio vectorial $V(n,q) = (\mathbb{F}_q)^n$. En este trabajo escribiremos el vector $(x_1, x_2, \ldots, x_n)$ como $x_1x_2\cdots x_n$. Un *código lineal* sobre $GF(q)$ es simplemente un subespacio vectorial de $V(n,q)$, para algún entero positivo n. Es decir, un subconjunto C de $V(n,q)$ es un código lineal si, y solo si:

1. $u + v \in C$, para todo u y v en C.
2. $au \in C$ para todo $u \in C$, $a \in GF(q)$.

En concreto, un código binario es lineal si, y solo si, la suma de dos palabras-código cualesquiera da como resultado una palabra-código.
Si C es un subespacio vectorial de $V(n,q)$ y su dimensión es k, entonces el código lineal C se dice que tiene parámetros $[n,k]$, o, si especificamos la distancia mínima d, tiene parámetros $[n,k,d]$.

Nota 3.

1. Un código q-ario $[n,k,d]$ es un código q-ario (n,q^k,d) por el Teorema 10. Sin embargo, no todo código con parámetros (n,q^k,d) es un código lineal de parámetros $[n,k,d]$.
2. El vector $0 = (0,0,\ldots,0)$ automáticamente pertenece a un código lineal.
3. Los códigos lineales a veces se llaman códigos de grupo.

El peso $w(x)$ de un vector $x \in V(n,q)$ se define como el número de entradas de x distinto de 0. Una de las propiedades más útiles de un código lineal es que su distancia mínima es igual al valor más pequeño de los pesos de sus palabras-código distintas de 0. Para demostrar esto necesitamos un lema muy sencillo.

Lema 7. *Si $x, y \in V(n,q)$, entonces*

$$d(x,y) = w(x-y).$$

Demostración. El vector $x - y$ tiene valores distintos de 0 cuando x e y difieren. □

Nota 4. Para $q = 2$, el Lema 7 es igual que el Lema 3, teniendo en cuenta que "+" es lo mismo que "−" si trabajamos con módulo 2.

Teorema 11. *Sea C un código lineal y sea $w(C)$ el peso más pequeño de las palabras-código de C distintas de 0. Entonces*

$$d(C) = w(C).$$

Demostración. Existen x e y palabras-código de C tales que $d(C) = d(x, y)$. Entonces, por el Lema 7,

$$d(C) = w(x - y) \geq w(C),$$

puesto que $x - y$ es una palabra-código del código lineal C.
Por otra parte, para alguna palabra-código $x \in C$,

$$w(C) = w(x) = d(x, 0) \geq d(C),$$

ya que 0 pertenece al código lineal C. Por tanto, $d(C) \geq w(C)$ y $w(C) \geq d(C)$, lo que demuestra que $d(C) = w(C)$. □

A continuación veremos algunas de las ventajas y desventajas de los códigos lineales.

1. Ventaja 1. En un código general con M palabras-código, para encontrar la mínima distancia debemos realizar $\binom{M}{2} = \frac{1}{2}M(M-1)$ comparaciones. Sin embargo, por el Teorema 11 podemos buscar la mínima distancia de un código lineal examinando solo los pesos de $M - 1$ palabras-código distintas de 0.
2. Ventaja 2. Para especificar un código no lineal, debemos numerar todas las palabras-código. Podemos especificar un código lineal de parámetros $[n, k]$ simplemente dando bases del espacio vectorial con k palabras-código.

   Definición 12. Una matriz de tamaño $k \times n$ cuyas filas forman una base de un código lineal de parámetros $[n, k]$ se llama *matriz generadora* del código.

3. Ventaja 3. Hay buenos procesos para codificar y decodificar un código lineal, esto se verá más adelante.

Ahora vamos a ver algunas desventajas:

1. Desventaja 1. Los códigos q-arios lineales no están definidos a menos que q sea una potencia prima. Sin embargo, hay códigos q-arios, donde q no es una potencia prima, que pueden obtenerse de códigos lineales sobre un gran alfabeto.
2. Desventaja 2. La restricción a códigos lineales parece una restricción a códigos más débiles y resta generalidad. Sin embargo, los códigos que son óptimos suelen ser frecuentemente lineales.

3.1.1. Equivalencia de códigos lineales

La definición de equivalencia dada anteriormente se modifica para códigos lineales, usando solo permutaciones del tipo producto, cuando estas se realizan con escalares no nulos. Así, dos códigos lineales sobre $GF(q)$ son equivalentes si uno puede ser obtenido a partir del otro mediante una combinación de operaciones de los siguientes tipos:

1. Permutaciones de las posiciones del código.
2. Multiplicación (por un escalar distinto de 0) de los símbolos que aparecen en una posición fija.

Teorema 12. *Dos matrices de tamaño $k \times n$ generan códigos lineales con parámetros $[n, k]$ equivalentes sobre $GF(q)$ si una de las dos matrices puede obtenerse a partir de la otra mediante una secuencia de operaciones de los siguientes tipos:*

1. *Permutación de filas.*
2. *Multiplicación de una fila por un escalar distinto de cero.*
3. *Adición del producto de una fila por un escalar a otra.*
4. *Permutación de columnas.*
5. *Multiplicación de cualquier columna por un escalar distinto de 0.*

Demostración. Las operaciones de fila (1,2,3) preservan la independencia lineal de las filas de una matriz generadora y simplemente reemplazan una base por otra del mismo código. Por otra parte, las operaciones del tipo (4,5) convierten una matriz generadora en otra que produce un código equivalente. □

Teorema 13. *Sea G una matriz generadora de un código de parámetros $[n, k]$. Entonces, realizando las operaciones del Teorema 12, se puede transformar G en una matriz en forma estándar.*

$$[I_k|A],$$

donde I_k es la matriz identidad de dimensión $k \times k$, y A es una matriz de dimensión $k \times (n-k)$.

Demostración. En una secuencia de transformaciones de la matriz G, denotamos por $g_{i,j}$ la entrada número (i, j) de la matriz y por $r_1, r_2, \ldots, r_k$ y $c_1, c_2, \ldots, c_n$ las filas y las columnas, respectivamente, de esta matriz. □

A continuación vamos a ver un proceso que consta de 3 pasos. Este proceso se aplica para $j = 1, 2, ..., k$ la j-ésima aplicación transforma la columna c_j en su forma deseada (con 1 en la j-ésima posición y 0 en el resto), dejando sin cambios las primeras $j - 1$ columnas ya convenientemente

transformadas. Supongamos entonces que G ya ha sido transformada

$$\begin{bmatrix} 1 & 0 & \cdots & 0 & g_{1,j} & \cdots & g_{1n} \\ 0 & 1 & \cdots & 0 & g_{2,j} & \cdots & g_{2n} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & g_{j-1,j} & \cdots & g_{j-1,n} \\ 0 & 0 & \cdots & 0 & g_{j,j} & \cdots & g_{j,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & g_{k,j} & \cdots & g_{k,n} \end{bmatrix}$$

1. Paso 1. Si $g_{j,j} \neq 0$ se pasa directamente al paso 2. Si $g_{j,j} = 0$ y si, además, para algunas $i > j$ se cumple que $g_{i,j} \neq 0$, entonces, se realiza el intercambio de r_j y r_i. Si se da el caso de que $g_{j,j} = 0$ y $g_{i,j} = 0$ para todo $i > j$, entonces elegimos h como $g_{j,h} \neq 0$ e intercambiamos c_j y c_h.
2. Paso 2. Ahora tenemos $g_{j,j} \neq 0$. Multiplicamos r_j por $g_{j,j}^{-1}$
3. Finalmente, si $g_{j,j} = 1$. Para cada $i = 1, 2, \ldots, k$, con $i \neq j$, reemplazamos r_i por $r_i - g_{i,j} \cdot r_j$.

La columna c_j ahora tiene la forma deseada. Después de aplicar estos tres pasos, la matriz generadora tendrá la forma estándar.

3.2. Codificando y decodificando con un código lineal

3.2.1. Codificando con un código lineal

Sea C un código lineal de parámetros $[n, k]$ sobre $GF(q)$ con G matriz generadora. Contiene q^k palabras-código y, por tanto, pueden ser usadas para comunicar cualesquiera q^k mensajes distintos. Identificamos esos mensajes con las q^k k-tuplas de $V(k, q)$ y codificamos un mensaje de tipo vector $u = u_1 u_2 \cdots u_k$ simplemente multiplicándolo por la parte derecha por G. Suponiendo que las filas de G son $r_1, r_2, \ldots, r_k$, entonces

$$uG = \sum_{i=1}^{k} u_i r_i.$$

Por tanto, uG es una palabra-código de C, ya que es una combinación lineal de las filas de la matriz generadora. Se ha de tener en cuenta que la función codificadora $u \to uG$ mapea el espacio vectorial $V(k, q)$ en un subespacio vectorial de $V(n, q)$ de dimensión k.
La regla de codificación es más sencilla si G está expresada de forma estándar. Suponemos que $G = [I_k|A]$, donde $A = [a_{i,j}]$ es una matriz de dimensión $k \times (n - k)$. Entonces el mensaje del vector u se codifica de la siguiente manera

$$x = uG = x_1 x_2 \cdots x_k x_{k+1} \cdots x_n,$$

donde $x_i = u_i, 1 \leq i \leq k$, son los dígitos de los mensajes y

$$x_{k+i} = \sum_{j=1}^{k} a_{ji} u_j, 1 \leq i \leq n-k.$$

son los dígitos de chequeo. Estos últimos representan la redundancia que se ha añadido al mensaje para protegerlo del ruido.

3.2.2. Decodificando con un código lineal

Suponemos que la palabra-código $x = x_1 x_2 \cdots x_n$ se envía a través de un canal y que al receptor le llega el vector $y = y_1 y_2 \cdots y_n$. Definimos el vector error e como:

$$e = y - x = e_1 e_2 \cdots e_n.$$

El decodificador debe decidir a partir de y qué palabra-código x ha sido transmitida, es decir, qué vector error e ha tenido lugar. Un ejemplo de esquema de decodificación, llamado de *vecino más cercano para códigos lineales*, fue creado por Slepian en 1960 y utiliza el hecho de que un código lineal es un subgrupo del grupo aditivo $V(n, q)$.

Definición 13. Suponemos que C es un código con parámetros $[n, k]$ sobre $GF(q)$ y a es cualquier vector perteneciente a $V(n, q)$. Entonces, el conjunto $a + C$ definido por

$$a + C = \{a + x | x \in C\}$$

es una clase del conjunto cociente $V(n, q)/C$ que llamaremos *clase de* C a partir de ahora.

Lema 8. *Suponemos que $a + C$ es una clase de C y que $b \in a + C$. Entonces,*

$$b + C = a + C.$$

Demostración. Sabiendo que $b \in a + C$, tenemos que $b = a + x$, para algún $x \in C$. Ahora, si $b + y \in b + C$, entonces

$$b + y = (a + x) + y = a + (x + y) \in a + C.$$

Por tanto, $b + C \subseteq a + C$. Por otro lado, si $a + z \in a + C$, entonces

$$a + z = (b - x) + z = b + (z - x) \in b + C.$$

Por tanto, $a + C \subseteq b + C$, y entonces $b + C = a + C$. □

El siguiente teorema es un caso particular del teorema de Lagrange para subgrupos.

Teorema 14. *Suponemos que C es un código de parámetros $[n, k]$ sobre $GF(q)$. Entonces*

1. *Todo vector de $V(n, q)$ pertenece a alguna clase de C.*
2. *Toda clase contiene exactamente q^k vectores.*
3. *Dos clases o son disjuntas o coinciden totalmente, no pueden coincidir solo parcialmente.*

En resumen, las clases de C constituyen una partición del espacio $V(n, q)$.

Demostración.

1. Si $a \in V(n,q)$, entonces $a = a + 0 \in a + C$.
2. La aplicación de C a $a + C$ definida por:
$$x \to a + x,$$
para todo $x \in C$, es inyectiva. Por esto, los cardinales siguientes cumplen, $|a+C| = |C| = q^k$.
3. Supongamos que las clases $a + C$ y $b + C$ se solapan. Entonces, para algún vector v, tenemos que $v \in (a + C) \cap (b + C)$. Así que, para algún $x, y \in C$,
$$v = a + x = b + y.$$
Por esto, $b = a + (x - y) \in a + C$, así que por el Lema 8, $b + C = a + C$.

□

Definición 14. El vector que tiene peso mínimo en una clase se llama líder de esa clase. En el caso de que hubiera varios vectores con el mismo peso mínimo, se elegiría uno de ellos al azar para ser el líder de clase.

El Teorema 14 muestra que $V(n,q)$ está dividido en clases disjuntas de C:
$$V(n,q) = (0 + C) \cup (a_1 + C) \cup \cdots \cup (a_s + C), \tag{3.1}$$
donde $s = q^{n-k} - 1$ y por el Lema 8, podemos tomar $0, a_1, \ldots, a_s$ como los líderes de clase. Una matriz estándar (Slepian) para un código de parámetros $[n,k]$ tiene dimensión $q^{n-k} \times q^k$, ya que contiene todos los vectores de $V(n,q)$. La primera fila está formada por el código C con un 0 en el extremo izquierdo, mientras que el resto de filas están formadas por las clases $a_i + C$ (con el orden correspondiente) y los líderes de cada clase en el extremo izquierdo. Una matriz estándar se construye de la siguiente manera:

1. Paso 1. Enumerar las palabras de C, empezando por 0, en la primera fila de la matriz.
2. Paso 2. Elegir cualquier vector a_1, que no esté en la primera fila, de peso mínimo. Enumerar la clase $a_1 + C$ para formar la segunda fila poniendo a_1 debajo de 0 y $a_1 + x$ debajo de x para cada $x \in C$.
3. Paso 3. Para los vectores que no están ni en la fila 1 ni en la 2, se elige a_2 con el peso mínimo y se enumera la clase a_2+C tal y como se ha hecho en el paso dos. Así obtendremos la fila 3.
4. Paso 4. Se continúa realizando los mismos pasos hasta que todas las clases se hayan enumerado y todo vector $V(n,q)$ aparezca exactamente una vez.

3.2.3. Probabilidad de corrección de errores

Para simplificarlo, en esta sección vamos a tener en cuenta únicamente códigos lineales binarios. Suponemos que el canal es binario y simétrico con p como probabilidad de error. Vimos en la introducción que la probabilidad de que el vector error sea un vector de peso i es $p^i(1-p)^{n-i}$, así obtenemos el siguiente teorema.

Teorema 15. *Sea C un código binario de parámetros $[n, k]$. Para $i = 0, 1, \ldots, n$, representamos por α_i el número de líderes de clase de peso i. Entonces, la probabilidad $P_{corr}(C)$ de que el vector recibido, decodificado por medio de una matriz estándar, sea la palabra-código que fue enviada viene dada por:*

$$P_{corr}(C) = \sum_{i=0}^{n} \alpha_i p^i (1-p)^{n-i}.$$

Nota 5. Si tenemos $d(C) = 2t + 1$ ó $2t + 2$, entonces C puede corregir t errores cualesquiera. Por eso, todo vector de peso $\leq t$ es un líder de clase y entonces, $\alpha_i = \binom{n}{i}$ para $0 \leq i \leq t$. Pero para $i > t$, α_i puede ser extremadamente difícil de calcular y son desconocidos incluso para algunas familias de códigos. En el caso de los códigos perfectos no existe esa dificultad, ya que los vectores error corregidos por un código perfecto de parámetros $[n, k, 2t+1]$ son precisamente aquellos vectores de peso $\leq t$. Aquí tenemos $\alpha_i = \binom{n}{i}$ para $0 \leq i \leq t$ y $\alpha_i = 0$ para $i > t$.

Un código lineal C de parámetros $[n, k]$ usa n símbolos para mandar k mensajes. Se dice que tiene una tasa de $R(C) = k/n$. La tasa de un código es la relación entre el número de símbolos del mensaje y el número total de símbolos enviados, por lo que un buen código tendrá una tasa alta.

Definición 15. La capacidad $\varphi(p)$ de un canal binario simétrico con probabilidad de error p es:

$$\varphi(p) = 1 + p \log_2 p + (1-p) \log_2 (1-p).$$

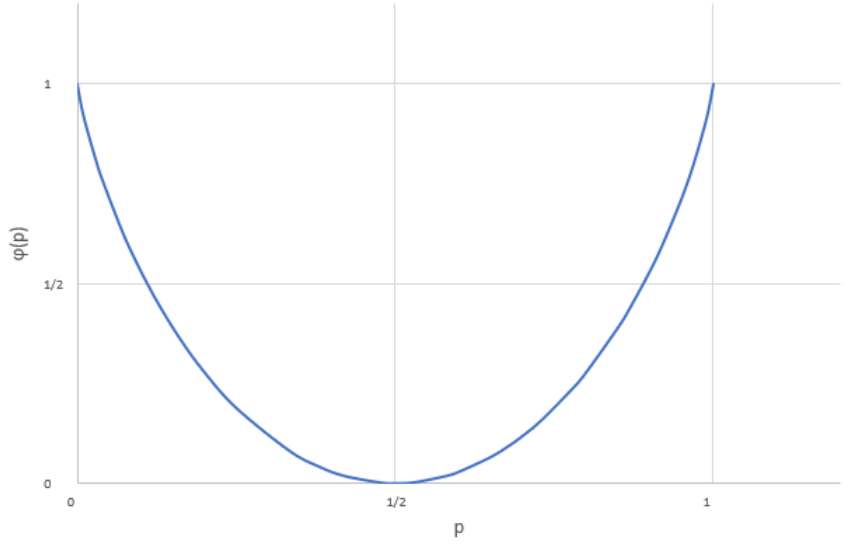

Figura 3.1: Gráfica de la capacidad $\varphi(p)$

Teorema 16. *Supongamos que un canal es binario y simétrico con probabilidad de error p. Supongamos que R es un número que satisface $R \geq \varphi(p)$. Entonces, para cualquier $\epsilon > 0$ y para un n suficientemente grande existe un código C de parámetros $[n, k]$ con tasa $k/n \geq R$, tal que $P_{err}(C) < \epsilon$.*

La demostración de este teorema solamente ha sido realizada mediante métodos probabilísticos y, por tanto, no nos sirve para saber como se construyen dichos códigos. La demostración se puede ver en el libro *Elements of Information Theory* [5]

3.2.4. Probabilidad de detección de errores

Suponemos ahora que el código lineal binario se usa solo para la detección de errores. El decodificador fallará al detectar errores que han ocurrido si, y solo si, el vector que se recibe y, es una palabra-código diferente de la palabra x que fue enviada. Por ello, la probabilidad $P_{nodetectar}(C)$ de que una palabra-código incorrecta sea recibida es independiente de la palabra-código enviada y viene dado por el siguiente teorema.

Teorema 17. *Sea C un código binario de parámetros $[n, k]$ y sea A_i el número de palabras-código de C de peso i. Entonces, si C se usa para detectar errores, la probabilidad de que un mensaje incorrecto sea recibido sin ser detectado es:*

$$P_{nodetectar}(C) = \sum_{i=1}^{n} A_i p^i (1-p)^{n-i}.$$

3.3. El código dual, la matriz de control de paridad y la decodificación por síndromes

Un código lineal puede darse a través de una matriz generadora, pero también con una matriz de control de paridad. Para esto debemos introducir ciertas definiciones.
El producto interior $u \cdot v$ de los vectores $u = u_1 u_2 \cdots u_n$ y $v = v_1 v_2 \cdots v_n$ en $V(n, q)$ es el escalar definido por:

$$u \cdot v = u_1 v_1 + u_2 v_2 + \cdots + u_n v_n.$$

En el caso que $u \cdot v = 0$, entonces u y v son ortogonales.

Lema 9. *Para cualesquiera u, v y w pertenecientes a $V(n, q)$ y $\lambda, \mu \in GF(q)$, se cumplen las siguientes igualdades:*

1. $u \cdot v = v \cdot u$.
2. $(\lambda u + \mu v) \cdot w = \lambda(u \cdot w) + \mu(v \cdot w)$.

Dado un código lineal C de parámetros $[n, k]$, el código dual de C, que se denota por $C^{\perp}$, se define como el conjunto de los vectores de $V(n, q)$ que son ortogonales a toda palabra-código de C, es decir:

$$C^{\perp} = \{v \in V(n, q) | v \cdot u = 0 \text{ para todo } u \in C\}.$$

Después del lema anterior, vamos a probar que $C^{\perp}$ es un código lineal de dimensión $n - k$.

Lema 10. *Supongamos que C es un código de parámetros $[n, k]$ con matriz generadora G. Entonces, un vector v de $V(n, q)$ pertenece a $C^{\perp}$ si, y solo si, v es ortogonal a todas las filas de G. Es decir, $v \in C^{\perp}$ si y sólo si $vG^T = 0$, donde G^T representa la matriz transpuesta de G.*

Demostración. La parte del *solo si* es fácil, ya que las columnas de G son palabras-código. Para la parte del *si*, suponemos que las filas de G son $r_1, r_2, \ldots, r_k$ y que $v \cdot r_i = 0$ para cada i. Si u es cualquier palabra-código de C, entonces $u = \sum_{i=1}^{k} \lambda_i r_i$ para algunos escalares λ_i y de esta forma:

$$v \cdot u = \sum_{i=1}^{k} \lambda_i (v \cdot r_i) = \sum_{i=1}^{k} \lambda_i 0 = 0.$$

Por tanto, v es ortogonal a todas las palabras-código de C y por ello está en $C^{\perp}$. □

Teorema 18. *Suponemos que C es un código lineal sobre $GF(q)$ con parámetros $[n, k]$. Entonces, el código dual $C^{\perp}$ de C es un código lineal con parámetros $[n, n - k]$.*

Demostración. Primeramente mostraremos que $C^{\perp}$ es un código lineal. Supongamos que $v_1, v_2 \in C^{\perp}$ y $\lambda, \mu \in GF(q)$. Entonces, para todo $u \in C$,

$$(\lambda v_1 + \mu v_2) \cdot u = \lambda(v_1 \cdot u) + \mu(v_2 \cdot u) = \lambda 0 + \mu 0 = 0.$$

Por tanto, $\lambda v_1 + \mu v_2 \in C^{\perp}$ y entonces $C^{\perp}$ es lineal.
Ahora mostraremos que $C^{\perp}$ tiene dimensión $n - k$. Sea $G = [g_{i,j}]$ una matriz generadora de C. Entonces, por el Lema 10, los elementos de $C^{\perp}$ son los vectores $v = v_1 v_2 \cdots v_n$ cumpliendo que $\sum_{j=1}^{n} g_{ij} v_j = 0$ para $i = 1, 2, \ldots, k$.
Este es un sistema de k ecuaciones homogéneas e independientes con n incógnitas y es un resultado estándar del álgebra lineal que el espacio de solución $C^{\perp}$ tiene dimensión $n - k$. Para completitud, lo probamos a continuación.
Es obvio que si los códigos C_1 y C_2 son equivalentes, entonces $C_1^{\perp}$ y $C_2^{\perp}$ también lo son. Por tanto, es suficiente mostrar que $\dim(C^{\perp}) = n - k$ cuando C tiene la matriz generadora con la forma estándar

$$\begin{bmatrix} 1 & \cdots & 0 & a_{1,1} & \cdots & a_{1,n-k} \\ \vdots & \ddots & \vdots & \vdots & & \vdots \\ 0 & \cdots & 1 & a_{k1} & \cdots & a_{k,n-k} \end{bmatrix}.$$

Entonces

$$C^{\perp} = \left\{ (v_1, v_2, \ldots, v_n) \in V(n, q) | v_i + \sum_{j=1}^{n-k} a_{ij} v_{k+j} = 0, i = 1, 2, \ldots, k \right\}.$$

Claramente, para cada una de las q^{n-k} opciones de $(v_{k+1}, \ldots, v_n)$, hay un único vector $(v_1, v_2, \ldots, v_n)$ en $C^\perp$. Por tanto, $|C^\perp| = q^{n-k}$ y entonces $\dim(C^\perp) = n - k$. □

Teorema 19. *Para cualquier código C de parámetros $[n, k]$ se tiene que $(C^\perp)^\perp = C$.*

Demostración. Claramente, $C \subseteq (C^\perp)^\perp$, ya que todo vector de C es ortogonal a todo vector en $C^\perp$. Pero $\dim((C^\perp)^\perp) = n - (n - k) = k = \dim(C)$, y por tanto, $C = (C^\perp)^\perp$. □

Definición 16. Una *matriz de control de paridad H* para un código C de parámetros $[n, k]$ es, por definición, una matriz generadora de $C^\perp$.
Así, H es una matriz de dimensión $(n - k) \times n$ que cumple $GH^T = 0$, donde H^T representa la transpuesta de H y 0 es una matriz formada por 0s. Entonces, por el Lema 10 y el Teorema 18, tenemos que si H es una matriz de control de paridad de C, entonces

$$C = [x \in V(n, q) | xH^T = 0].$$

De esta manera cualquier código lineal está completamente definido por una matriz de control de paridad.

El siguiente teorema muestra una manera fácil de construir una matriz de control de paridad para un código lineal a partir de una matriz generadora, y también al revés.

Teorema 20. *Si $G = [I_k|A]$ es la forma estándar de la matriz generadora de un código lineal C de parámetros $[n, k]$, entonces la matriz de control de paridad de C es $H = [-A^T|I_{n-k}]$.*

Demostración. Suponemos que

$$G = \left[\begin{array}{ccc|ccc} 1 & & 0 & a_{11} & \cdots & a_{1,n-k} \\ & \ddots & & \vdots & & \vdots \\ 0 & & 1 & a_{k1} & \cdots & a_{k,n-k} \end{array}\right].$$

Sea

$$H = \left[\begin{array}{ccc|ccc} -a_{11} & \cdots & -a_{k1} & 1 & & 0 \\ \vdots & & \vdots & & \ddots & \\ -a_{1,n-k} & \cdots & -a_{k,n-k} & 0 & & 1 \end{array}\right].$$

Entonces H tiene el tamaño requerido de una matriz de control de paridad y sus filas son linealmente independientes. Por lo tanto, es suficiente con demostrar que cada fila de H es ortogonal a cada fila de G. Esto se deduce de que el producto escalar de la fila i de G y la j de M da como resultado:

$$0 + \cdots + 0 + (-a_{ij}) + 0 + \cdots + 0 + a_{ij} + 0 + \cdots + 0 = 0.$$

□

Definición 17. Una *matriz de control de paridad H* se dice que está en forma estándar si $H = [B|I_{n-k}]$.

La demostración del Teorema 20 muestra que si un código está dado por una matriz de control de paridad en su forma estándar $H = [B|I_{n-k}]$, entonces una matriz generadora para el código es $G = [I_k| - B^T]$. Muchos códigos, como por ejemplo los códigos Hamming, se definen mediante una matriz de control de paridad o, equivalentemente, un conjunto de ecuaciones de control de paridad. Si un código viene dado por una matriz de control de paridad H, que no está en forma estándar, entonces H puede ser reducida a esta forma de la misma manera que para una matriz generadora.

3.3.1. Decodificación por síndrome

Supongamos que H es una matriz de control de paridad de un código C. Entonces, para cualquier vector $y \in V(n, q)$, se llama *síndrome* de y al vector de dimensión $1 \times (n - k)$

$$S(y) = yH^T.$$

llamado síndrome de y.

Nota 6.

1. Si las filas de H son $h_1, h_2, \ldots, h_{n-k}$, entonces $S(y) = (y \cdot h_1, y \cdot h_2, \ldots, y \cdot h_{n-k})$.
2. $S(y) = 0$ si, y sólo si, $y \in C$.
3. También se define el síndrome de y como el vector columna Hy^T (o sea, el vector transpuesto de $S(y)$).

Lema 11. *Dos vectores u y v están en la misma clase de C si, y sólo si, tienen el mismo síndrome.*

Demostración. Se demostrarán ambas partes a la vez.
Sabemos que u y v están en la misma clase si, y sólo si, $u+C = v+C$, lo que ocurre únicamente si $u-v \in C$. Además, sabemos que $u-v \in C$ si, y sólo si, $(u-v)H^T = 0$, y operando obtenemos que $uH^T = vH^T$. Por último, sabemos que $uH^T = vH^T$ ocurre solamente si $S(u) = S(v)$. □

Corolario 3. *Existe una correspondencia biunívoca entre clases y síndromes.*

En la decodificación de la matriz estándar, si n es pequeño, no hay dificultad para ubicar el vector recibido y en la matriz. Pero si n es grande, podemos ahorrar mucho tiempo utilizando el síndrome para averiguar qué clase contiene a y. Hacemos esto de la siguiente manera.
Se calcula el síndrome $S(e)$ para cada líder de clase e y se extiende la matriz estándar enumerando los síndromes en una columna extra.
El proceso de decodificación es el siguiente:

1. Para el vector recibido y, calculamos $S(y) = yH^T$.
2. Sea $z = S(y)$, localizamos z en la primera columna de la tabla de búsqueda.
3. Se decodifica y como $y - f(z)$.

Vemos ahora un ejemplo. Tomamos

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

y por el Teorema 20, una matriz de control de paridad es:

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

Entonces, los síndromes de los líderes de clases son

$$\begin{aligned} S(0000) &= 00 \\ S(1000) &= 11 \\ S(0100) &= 01 \\ S(0010) &= 10. \end{aligned}$$

La matriz estándar será:

líderes de clase				síndromes
0000	1011	0101	1110	00
1000	0011	1101	0110	11
0100	1111	0001	1010	01
0010	1001	0111	1100	10.

Ahora el algoritmo de decodificación consiste: dado y, el vector que se recibe, calcular $S(y) = yH^T$ y localizar $S(y)$ en la columna de síndromes de la matriz. Localizamos y en la correspondiente fila y la decodificamos como la palabra-código en la parte superior de la columna que contiene y. Por ejemplo, si se recibe 1111, entonces $S(1111) = 01$ y, por tanto, 1111 aparece en la tercera fila de la matriz.
Cuando programamos un ordenador para hacer una matriz estándar decodificadora, necesitamos guardar solo dos columnas (los síndromes y los líderes de clase) en la memoria del ordenador. Esta matriz se llama tabla de búsqueda por síndrome. La tabla de búsqueda por síndrome de el código anterior es:

síndrome z	líderes de clase $f(z)$
00	0000
11	1000
01	0100
10	0010.

El proceso de decodificación es el siguiente:
Paso 1 Para un vector recibido y calcula $S(y) = yH^T$.
Paso 2 Sea $z = S(y)$, y localizando a z en la primera columna de la tabla de búsqueda.
Paso 3 Decodificar y como $y - f(z)$.
Por ejemplo, si $y = 1111$, entonces $S(y) = 01$ y lo decodificamos como $1111 - 0100 = 1011$.

3.3.2. Decodificación incompleta

La decodificación incompleta es una combinación de corrección y detección de errores, esta última se usa cuando es probable que la corrección proporcione la palabra-código incorrecta. Es decir, si $d(C) = 2t + 1$ ó $2t + 2$, adoptamos el siguiente esquema por el cual garantizamos la corrección de t errores o menos en cualquier palabra-código, en algunos casos, se detectan más de t errores.
Organizamos las clases de la matriz estándar en orden creciente según peso de los líderes de clases, y dividimos la matriz en dos partes: la superior, que comprende aquellas clases cuyos líderes tienen pesos menores o iguales a t y la inferior, que comprende las clases restantes. Si el vector recibido está en la parte superior, lo decodificamos como de costumbre, si y está en la parte inferior, sabemos que se han producido más de t errores y solicitamos la retransmisión.

Un esquema de decodificación incompleto es adecuado, especialmente, para códigos con distancia mínima par. Esto es así porque, si $d(C) = 2t + 2$, entonces se podrán corregir hasta t errores y simultáneamente detectar $t + 1$ errores.
Cuando se lleva a cabo una decodificación incompleta mediante una tabla de consulta de síndromes, podemos prescindir de la matriz estándar en el esquema de decodificación, y en la construcción de la tabla. Esto se debe a que sabemos que los líderes de clases están en la parte superior de la matriz (son todos los vectores con peso $\leq t$), mientras que los de la mitad inferior no se usan en la decodificación. En definitiva, solo almacenamos la parte superior de una tabla de búsqueda de síndromes.

A continuación vamos a ver un ejemplo.
Consideramos el código lineal con parámetros $[10, 8]$ sobre $GF(11)$ con la siguiente matriz de control de paridad:

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{bmatrix}.$$

Se ha elegido H, que es una matriz que no está en forma estándar con la finalidad de obtener un buen algoritmo decodificador. Sea C un código 10-ario que se obtiene a partir de un código 11-ario, eliminando las palabras-código que contienen el dígito 10. Es decir, C es el código de 10 dígitos de números decimales $x = x_1x_2 \cdots x_{10}$ cumpliendo las dos ecuaciones de control de paridad:

$$\sum_{i=1}^{10} x_i \equiv 0 \mod 11 \qquad \text{y} \qquad \sum_{i=1}^{10} ix_i \equiv 0 \mod 11.$$

Mediante el principio de inclusión-exclusión, se puede mostrar que C contiene 82644629 palabras-código pero no vamos a ver la demostración. Las palabras-código de C pueden ser enumeradas usando una matriz generadora en forma estándar. El primer paso es poner H en forma estándar mediante operaciones sobre las filas. Llamamos f_1 y f_2 a la primera y segunda fila de la matriz

respectivamente.

$$H \longrightarrow (f_1 \to f_1 + f_2) \begin{bmatrix} 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 0 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{bmatrix}$$

$$\longrightarrow (f_1 \to (-1)f_1) \text{ y } (f_2 \to (-1)f_2) \begin{bmatrix} 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\ 10 & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 \end{bmatrix}$$

$$\longrightarrow (f_2 \to f_2 - 2f_1) \begin{bmatrix} 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\ 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 0 & 1 \end{bmatrix}.$$

Usando el Teorema 20 tenemos que

$$G = \left[\begin{array}{c|c} & 2 \ 8 \\ & 3 \ 7 \\ & 4 \ 6 \\ I_8 & 5 \ 5 \\ & 6 \ 4 \\ & 7 \ 3 \\ & 8 \ 2 \\ & 9 \ 1 \end{array}\right],$$

y entonces $C = \{(x_1, x_2, \ldots, x_8, 2x_1 + 3x_2 + \cdots + 9x_8, 8x_1 + 7x_2 + \cdots + x_8)\}$, donde $x_1, x_2, \ldots, x_8$ toman valores 0,1,2,...,9 y se omiten las palabras que tienen el dígito 10 en cualquiera de los dos últimos lugares de las coordenadas.

Ahora describimos un esquema de decodificación por síndrome incompleto que corregirá cualquier error único y que simultáneamente detectará cualquier error doble que surja de la transposición de dos dígitos de una palabra-código.

Suponemos que $x = (x_1, x_2, \ldots, x_{10})$ es la palabra-código transmitida y que $y = (y_1, y_2, \ldots, y_{10})$ es el vector recibido. El síndrome

$$(A, B) = yH^T = \left(\sum_{i=1}^{10} y_i, \sum_{i=1}^{10} iy_i\right)$$

se calcula (módulo 11).

Suponemos que ha ocurrido un único error, así que para j y k distintas de 0 tenemos, $(y_1, y_2, \ldots, y_{10}) = (x_1, \ldots, x_{j-1}, x_j + k, x_{j+1}, \ldots, x_{10})$.

Entonces

$$A = \sum_{i=1}^{10} y_i = \left(\sum_{i=1}^{10} x_i\right) + k \equiv k \qquad \text{y}$$

$$B = \sum_{i=1}^{10} iy_i = \left(\sum_{i=1}^{10} ix_i\right) + jk \equiv jk$$

ambas calculadas (módulo 11).

Por tanto, la magnitud de error k viene dada por A y la posición de error j viene dada por el valor de B/A. Entonces, el esquema de decodificación es como vemos a continuación:

1. Si $(A, B) = (0, 0)$, entonces y es una palabra-código donde suponemos que no hay errores.
2. Si $A \neq 0$ y $B \neq 0$, entonces suponemos que ha ocurrido un único error que se corrige eliminando A de la entrada número (B/A) de y.
3. Si $A = 0$ o $B = 0$, pero no ambos, entonces se han detectado al menos dos errores. Este caso siempre surge si se transponen dos dígitos de una palabra-código, pues entonces $A = 0$ y $B \neq 0$.

Por ejemplo, suponemos que $y = 0610271355$. Calculamos que $A = 8$ y que $B = 6$. Por tanto, $B/A = 6 \cdot 8^{-1} = 6 \cdot 7 = 42 = 9$, así que el dígito número 9 debe ser $5 - 8 = -3 = 8$.

3.4. El problema principal de la teoría de la codificación lineal

En la introducción se habló del problema principal de la teoría de la codificación. Este problema consistía en encontrar $A_q(n, d)$, es decir, el mayor valor de M para el cual existe un código q-ario de parámetros (n, M, d). A continuación, se va a tratar el mismo problema, pero, restringiéndolo a códigos lineales.
Si q es una potencia prima, denotamos por $B_q(n, d)$ el mayor valor de M para el cual existe un código lineal de parámetros (n, M, d) sobre $GF(q)$. Claramente, $B_q(n, d)$ es siempre una potencia de q y $B_q(n, d) \leq A_q(n, d)$. Por tanto, nos referimos al problema de encontrar $B_q(n, d)$ como el problema principal de la teoría de la codificación lineal conocido como MLCT por sus siglas en inglés.

Si tomamos los valores de q y d fijos, el problema se presenta de la siguiente forma:

Versión 1 del problema MLCT. Para una longitud dada (n), encontrar la mayor dimensión k tal que exista un código de parámetros $[n, k, d]$ sobre $GF(q)$. Entonces, para esta k se tiene que $B_q(n, d) = q^k$.

La redundancia r de un código de parámetros $[n, k, d]$ se define como $n - k$, es decir, el número de símbolos que tiene una palabra-código. Ahora vamos a ver una versión alternativa.

Versión 2 del problema MLCT. Para una redundancia dada (r), encontrar la longitud máxima n tal que exista un código de parámetros $[n, n - r, d]$ sobre $GF(q)$.

Resolver la versión 1 para todos los valores de n equivale a resolver la versión 2 para todos los valores de r. Esto se debe a que, en ambos casos, conocemos exactamente los valores de n y k para los cuales existe un código de parámetros $[n, k, d]$. Esta equivalencia se podrá ver en un teorema que trataremos más adelante. Por otro lado, la versión 2 nos da una aproximación más natural que se verá en el proximo teorema, pero para ello debemos dar unas definiciones.

Definición 18. Un conjunto de parámetros (n, s) en $V(r, q)$ es un conjunto de n vectores pertenecientes a $V(r, q)$ con la propiedad de que cualquier subconjunto s elementos son linealmente independientes.

Se denota por $\text{máx}_s(r, q)$ el mayor valor de n para el cual existe un conjunto de parámetros (n, s) en $V(r, q)$. Un conjunto de parámetros (n, s) en $V(r, q)$ que cumple $n = \text{máx}_s(r, q)$ se llama optimal. El problema del empaquetado para $V(r, q)$ es determinar los valores de $\text{máx}_s(r, q)$ y los conjuntos de parámetros (n, s) optimales. Este problema fue considerado por Bose en 1947 con interés estadístico y en 1961 por su relación con la teoría de códigos, que viene dada por el Teorema 22. Necesitamos aquí un resultado previo.

Teorema 21. *Suponemos que C es un código lineal de parámetros $[n, k]$ sobre $GF(q)$ con matriz de paridad H. Entonces, la distancia mínima de C es d si, y solo si, cualesquiera $d-1$ columnas de H son linealmente independientes pero algunas d columnas son linealmente dependientes.*

Demostración. Por el Teorema 11 sabemos que la mínima distancia de C es igual al menor peso de las palabras-código distintas de 0. Sea $x = x_1x_2 \cdots x_n$ un vector en $V(n, q)$. Entonces $x \in C$ si, y solo si, $xH^T = 0$, y esto es equivalente a $x_1H_1 + x_2H_2 + \cdots + x_nH_n = 0$, donde $H_1, H_2, \ldots, H_n$ representan las columnas de H.
Entonces, para cada palabra-código x de peso d, hay un conjunto de d columnas linealmente dependientes que pertenecen a H. Por otro lado, si existiera un conjunto de $d-1$ columnas linealmente dependientes de H llamadas $H_{i_1}, H_{i_2}, \ldots, H_{i_{d-1}}$, entonces existirían los escalares $x_{i_1}, x_{i_2}, \ldots, x_{i_{d-1}}$, distintos de 0 tales que

$$x_{i_1}H_{i_1} + x_{i_2}H_{i_2} + \cdots + x_{i_{d-1}}H_{i_{d-1}} = 0$$

Pero entonces el vector $x = (0 \cdots 0x_{i_1}0 \cdots 0x_{i_2}0 \cdots 0x_{i_{d-1}}0 \cdots 0)$, teniendo x_{i_j} en la posición i_j para $j = 1, 2, \ldots, d-1$ y 0s en el resto, cumpliría que $xH^T = 0$ y por tanto, sería una palabra-código de peso menor que d. □

Teorema 22. *Existe un código de parámetros $[n, n-r, d]$ sobre $GF(q)$ si, y solo si, existe un conjunto de parámetros $(n, d-1)$ en $V(r, q)$.*

Demostración. Supongamos que C es un código de parámetros $[n, n-r, d]$ sobre $GF(q)$ con H como matriz de control de paridad. Entonces, por el Teorema 21, las columnas de H forman un conjunto de parámetros $(n, d-1)$ en $V(r, q)$. Por otro lado, suponemos que K es un conjunto de parámetros $(n, d-1)$ en $V(r, q)$. Si formamos una matriz H de dimensión $r \times n$ con los vectores de K como sus columnas, entonces, de nuevo por el Teorema 21, H es la matriz de control de paridad de un código de parámetros $[n, n-r]$ cuya distancia mínima es, al menos, d. □

Corolario 4. *Dados los valores enteros positivos q, d y r, el mayor valor de n para el cual existe un código de parámetros $[n, n-r, d]$ sobre $GF(q)$ es $\text{máx}_{d-1}(r, q)$.*

A partir del Corolario 4 podemos deducir que, el problema principal de la teoría de la codificación lineal, concretamente la versión dos de este, es equivalente a encontrar el $\text{máx}_{d-1}(r, q)$. A continuación, vamos a ver que los valores de $B_q(n, d)$ vienen dados también por la solución de este problema.

Teorema 23. *Suponemos que* $\text{máx}_{d-1}(r-1,q) \leq n \leq \text{máx}_{d-1}(r,q)$. *Entonces* $B_q(n,d) = q^{n-r}$.

Demostración. Como $n \leq \text{máx}_{d-1}(r,q)$, existe un código de parámetros $[n, n-r, d]$ sobre $GF(q)$ y, por tanto, $B_q(n,d) \geq q^{n-r}$. Si $B_q(n,d)$ fuera estrictamente mayor que q^{n-r}, entonces existiría un código de parámetros $[n, n-r+1, d]$, lo que implica que $n \leq \text{máx}_{d-1}(r-1,q)$, contradiciendo la hipótesis. □

Consideraremos a continuación el problema MLCT para valores crecientes de la distancia mínima d. Los casos $d = 1$ y $d = 2$ se tratan fácilmente. Por lo tanto, consideraremos primero el problema para $d = 3$ y lo resolveremos para todos los valores de q y r. Luego consideraremos el caso $d = 4$, resolviendo el problema MLCT para $q = 2$ y dando los resultados conocidos para $q > 2$. Para los casos de d mayor que 4, se sabe muy poco en cuanto a los resultados generales.

3.4.1. El problema MLCT para $d = 3$

Antes de comenzar a tratar el problema se va a ver una nota que nos servirá para comprender algunos conceptos necesarios.

Nota 7. Cualquier vector v perteneciente a $V(r,q)$ tiene exactamente $q-1$ múltiplos escalares distintos de 0, formando el conjunto $\{\lambda v | \lambda \in GF(q), \lambda \neq 0\}$. En efecto, los $q^r - 1$ vectores distintos de 0 pertenecientes a $V(r,q)$ se pueden particionar en $(q^r-1)/(q-1)$ conjuntos que llamaremos clases, tales que dos vectores son escalares múltiples uno respecto a otro si y solo si están en la misma clase. Si elegimos un vector de cada clase obtenemos un conjunto de $(q^r-1)/(q-1)$ vectores, donde ningún par de ellos son linealmente dependientes. Entonces, por el Teorema 21, tomando esas columnas como las columnas de H se produce una matriz de control de paridad de un código de parámetros $[(q^r-1)/(q-1), (q^r-1)/(q-1)-r, 3]$. Este código se llama código Hamming q-ario y se denota por $\text{Ham}(r,q)$. Hay que recordar que para definir $\text{Ham}(r,q)$ se deben elegir diferentes matrices de control de paridad, pero cualquier matriz puede ser obtenida a partir de otra por permutación de las columnas y/o multiplicación de estas por escalares que sean distintos de 0. Por tanto, los códigos de Hamming son códigos lineales que están definidos de manera única por sus parámetros.

Teorema 24. *Dada una redundancia* r*, la máxima longitud* n *de un código de parámetros* $[n, n-r, 3]$ *sobre* $GF(q)$ *es* $(q^r-1)/(q-1)$. *Es decir,* $\text{máx}_2(r,q) = (q^r-1)/(q-1)$.

Demostración. Por el Corolario 4, el valor de n es necesariamente $\text{máx}_2(r,q)$, el mayor valor de un conjunto $(n,2)$ en $V(r,q)$. Ahora un conjunto S de vectores en $V(r,q)$ es un conjunto de parámetros $(n,2)$ si, y solo si, ningún vector de S es un múltiplo escalar de cualquier vector de S. Como se puede ver en la Nota 7, los $q^r - 1$ vectores distintos de 0 que pertenecen a $V(r,q)$ se particionan en $(q^r-1)/(q-1)$ clases, cada clase contiene $q-1$ vectores cuyos escalares son múltiplos del resto. Entonces, el conjunto más grande con parámetros $(n,2)$ es un conjunto de $(q^r-1)/(q-1)$ vectores, uno de cada una de esas clases. □

Los códigos optimales de parámetros $[n, n-r, 3]$ con $n = (q^r - 1)/(q-1)$ son códigos $\text{Ham}(r, q)$, definidos en la Nota 7. La solución al problema MLCT, concretamente a la versión 1, se consigue a partir de los Teoremas 23 y 24 y la mostramos en el siguiente teorema.

Teorema 25. *$B_q(n, 3) = q^{n-r}$, donde r es el único entero que cumple $(q^{r-1} - 1)/(q-1) < n \leq (q^r - 1)/(q-1)$.*

Nota 8.

1. Es fácil expresar $B_q(n, 3)$ como una función explícita de q y n. Por el Teorema 24, existe un código de parámetros $[n, n-r, 3]$ sobre $GF(q)$ si y sólo si $n \leq (q^r)/(q-1)$. A su vez, esto último ocurre si, y sólo si, $r \geq \log_q\{n(q-1)+1\}$, lo que equivale a que $n - r \leq n - \log_q\{n(q-1)+1\}$. Por tanto, se cumple que $B_q(n, 3) = q^{\lfloor n - \log_q\{n(q-1)+1\}\rfloor}$.
2. Para construir un código lineal de parámetros $(n, M, 3)$ con $M = B_q(n, 3)$, se debe encontrar el menor entero r tal que $n \leq (q^r - 1)/(q-1)$. Se escribe como una matriz de control de paridad, con n vectores columna de $V(r, q)$ tales que ninguna columna es múltiple de otra. Esa matriz de control de paridad siempre puede obtenerse borrando columnas de la matriz de control de paridad de un código Hamming ($Ham(r, q)$). Por tanto, los mejores códigos lineales correctores de un solo error con una longitud prefijada son o bien códigos Hamming o códigos acortados de Hamming.

Antes de ver el caso $d = 4$, es importante observar que un conjunto de parámetros (n, s) se puede ver como un conjunto de puntos en el espacio proyectivo asociado $PG(r-1, q)$ que vemos a continuación.

Definición 19. Dado el espacio vectorial $V(r, q) = \{(a_1, a_2, \ldots, a_r) | a_i \in GF(q)\}$, le asociamos una estructura combinatoria $PG(r-1, q)$ que está formada por puntos y líneas que se definen de la siguiente manera.
Los puntos de $PG(r-1, q)$ son los espacios vectoriales unidimensionales de $V(r, q)$. Las líneas de $PG(r-1, q)$ son espacios vectoriales bidimensionales de $V(r, q)$. El punto P pertenece a la línea L si, y solo si, P es un subespacio de L. $PG(r-1, q)$ recibe el nombre de espacio proyectivo de dimensión $r-1$ sobre $GF(q)$. Cada punto P de $PG(r-1, q)$, como subespacio de $V(r, q)$ de dimensión 1, se genera por un único vector distinto de 0. Entonces, si $a = (a_1, a_2, \ldots, a_r) \in P$, se cumple que

$$P = \{\lambda a | \lambda \in GF(q)\}.$$

En la práctica, identificamos el punto P con cualquier vector distinto de 0 que lo contenga. Dicho de otra forma, los puntos de $PG(r-1, q)$ son los vectores distintos de 0 de $V(r, q)$ con la regla de que si $a = (a_1, a_2, \ldots, a_r)$ y $b = (b_1, b_2, \ldots, b_r)$ son dos de esos vectores, entonces

$$a = b \text{ en } PG(r-1, q) \text{ si y solo si } a = \lambda b \text{ en } V(r, q)$$

para algún λ distinto de 0.

Ahora vamos a ver algunas propiedades elementales de $PG(r-1, q)$.

Lema 12. *En $PG(r-1, q)$ se cumplen las siguientes afirmaciones.*

1. *Su número de puntos es* $(q^r - 1)/(q - 1)$.
2. *Dados dos puntos cualesquiera, hay una única recta que los contiene.*
3. *Cada recta contiene exactamente* $q + 1$ *puntos.*
4. *Cada punto aparece en* $(q^{r-1} - 1)/(q - 1)$ *rectas.*

Demostración.

1. Como cada uno de los $q^r - 1$ vectores distintos de 0 de $V(r, q)$ tiene $q-1$ escalares distintos de 0, el número de puntos de $PG(r - 1, q)$ es $(q^r - 1)/(q - 1)$.
2. Si a y b son distintos puntos de $PG(r - 1, q)$, entonces la única línea que pasa por ellos contiene los puntos $\lambda a + \mu b$, donde λ y μ son escalares y al menos uno de ellos es distinto de 0.
3. En 2, hay $q^2 - 1$ elecciones para el par (λ, μ), pero como estamos identificando múltiplos por escalares, el número de puntos distintos en la línea es $(q^2 - 1)/(q - 1) = q + 1$.
4. Sea t el número de rectas en las cuales aparece un punto P. Sea X el conjunto $\{(Q, L)|Q$ es un punto $\neq P, L$ es una línea que contiene a ambos P y $Q\}$. Contamos los elementos de X de dos maneras distintas. Para cada una de las $(q^r - 1)/(q - 1) - 1$ opciones de Q, hay una única línea L que contiene a P y a Q. Entonces

$$|X| = (q^r - 1)/(q - 1) - 1 = (q^r - q)/(q - 1).$$

   Por otro lado, para cada una de las t líneas que pasan por P, hay (por el apartado 3), q puntos Q distintos de P que se encuentran en L. Así,

$$|X| = tq.$$

   Operando con ambas expresiones obtenemos que $t = (q^{r-1} - 1)/(q - 1)$.

□

Definición 20. El espacio proyectivo $PG(2, q)$ se llama en realidad plano proyectivo sobre $GF(q)$. Por el Lema 12 se sabe que $PG(2, q)$ es un diseño de parámetros $(q^2 + q + 1, q + 1, 1)$ simétrico, por tanto es un plano proyectivo como se definió en el apartado 2.2.

Nota 9.

1. Los puntos de $PG(r - 1, q)$ pueden ser etiquetados haciendo que la coordenada, distinta de 0, situada más a la izquierda sea igual a 1.
2. Si $q = 2$, los puntos de $PG(r - 1, 2)$ vienen dados por los vectores distintos de 0 pertenecientes a $V(r, 2)$.

Definición 21. Un conjunto K de n puntos en $PG(r - 1, q)$ es un conjunto de parámetros (n, s) si los vectores que representan los puntos de K forman un conjunto de parámetros (n, s) en el espacio vectorial subyacente $V(r, q)$.

Nota 10.

1. Dos ventajas de trabajar con $PG(r-1,q)$ son: primero, que se pueden usar algunos argumentos de conteo para obtener cotas superiores de $\text{máx}_s(r,q)$ y, segundo, que muchos conjuntos de parámetros (n,s) óptimos resultan ser configuraciones geométricas naturales.

2. Un conjunto de parámetros $(n,2)$ en $PG(r-1,q)$ es simplemente un conjunto de n puntos distintos de $PG(r-1,q)$. Así que describimos el código Hamming $Ham(r,q)$ como un código con la matriz de control de paridad H, cuyas columnas son los diferentes puntos de $PG(r-1,q)$. Distintas representaciones de esos puntos como vectores darán como resultado diferentes códigos, que serán equivalentes entre si.

3.4.2. El problema MLCT para d=4

La longitud máxima de un código de parámetros $[n, n-r, 4]$, para una r dada, es igual al valor de $\text{máx}_3(r,q)$, que es la mayor medida de un conjunto de parámetros $(n,3)$ en $V(r,q)$. Un conjunto de parámetros $(n,3)$ en el plano $PG(2,q)$ se llama un n-arco, mientras que un conjunto de parámetros $(n,3)$ en $PG(r-1,q)$, para $r>3$, se llama n-gorro. Como 3 puntos de $PG(r-1,q)$ son linealmente dependientes si y solo si son colineales, puede describirse un n-arco/n-gorro como un conjunto de n puntos, de los que ningún conjunto de tres son colineales. El problema de determinar los valores de $\text{máx}_3(r,q)$, fue primeramente considerado por Bose en 1942. Esto fue rápidamente resuelto para $q=2$, para todo r, y para $r \leq 4$, para todo q. Sin embargo, a pesar de que fue un problema muy importante, se ha resuelto sólo para los pares adicionales $(r,q)=(4,3)$ y $(5,3)$. Los valores conocidos de $\text{máx}_3(r,q)$ se pueden observar en la siguiente expresión.

$\text{máx}_3(r,2) = 2^{r-1}$.

$$\text{máx}_3(3,q) = \begin{cases} q+1, & \text{si } q \text{ es impar,} \\ q+2, & \text{si } q \text{ es par.} \end{cases}$$

$$\text{máx}_3(4,q) = \begin{cases} q^2+1, & \text{si } q \text{ es impar,} \\ q^2+2, & \text{si } q \text{ es par.} \end{cases}$$

$\text{máx}_3(5,3) = 20$.

$\text{máx}_3(6,3) = 56$.

Ahora veremos algunas demostraciones.

La determinación de $\text{máx}_3(r, 2)$

Aquí tratamos el problema de encontrar códigos binarios lineales optimales con $d = 4$. El siguiente Teorema muestra que debemos obtener esos códigos a partir de los códigos optimales de mínima distancia 3 simplemente añadiendo un dígito de chequeo de la paridad.

Teorema 26. *Suponemos que d es impar. Entonces existe un código binario de parámetros $[n, k, d]$ si, y solo si, existe un código binario de parámetros $[n + 1, k, d + 1]$.*

Demostración. La demostración del Teorema 3 es válida bajo la restricción de que los códigos sean lineales. Esto se debe a que un código extendido (es decir, el código obtenido de un dígito de chequeo de la paridad) es lineal. □

Corolario 5. *Suponemos que d es par. Entonces*

1. $B_2(n, d) = B_2(n - 1, d - 1)$.
2. $\text{máx}_{d-1}(r, 2) = \text{máx}_{d-2}(r - 1, 2) + 1$.

Demostración.

1. Se obtiene inmediatamente del Teorema 26.
2. El hecho de que $n \leq \text{máx}_{d-1}(r, 2)$, ocurre únicamente si existe un código binario de parámetros $[n, n-r, d]$. Esto último sabemos que sucede si y sólo si existe un código binario de parámetros $[n-1, n-r, d-1]$, y este existe si, y solo si, $n-1 \leq \text{máx}_{d-2}(r-1, 2)$, que es equivalente a $n \leq \text{máx}_{d-2}(r-1, 2) + 1$.

□

Corolario 6. $\text{máx}_3(r, 2) = 2^{r-1}$.

Demostración. Por el Teorema 24, $\text{máx}_2(r, 2) = 2^r - 1$. Entonces $\text{máx}_3(r, 2) = (2^{r-1} - 1) + 1 = 2^{r-1}$. □

El código binario optimal con $d = 4$ y redundancia r es el código Hamming extendido $Ham(r - 1, 2)$. Como vemos, la matriz de control de paridad de este código es:

$$\bar{H} = \begin{bmatrix} & & & 0 \\ & & H & \vdots \\ & & & 0 \\ 1 & 1 & \cdots & 1 \end{bmatrix},$$

donde H es una matriz de control de paridad para $\text{Ham}(r-1, 2)$, tal que las columnas de H son justamente los puntos de $PG(r-2, 2)$. Las columnas de $\bar{H}$ forman un 2^{r-1}-gorro en $PG(r-1, 2)$. $\bar{H}$ consiste en los puntos de $PG(r-1, 2)$ que no pertenecen al subespacio $\{(x_1, \ldots, x_r) | x_r = 0\}$. Geométricamente, se puede describir como el complemento de un hiperplano.

La determinación de $\text{máx}_3(3, q)$

Primeramente se darán algunos ejemplos de buenos códigos lineales con $d = 4$ y redundancia 3. Después, probaremos que esos códigos son optimales mostrando que no pueden existir códigos de mayor longitud.

Teorema 27. *Sean* $a_1, a_2, \ldots, a_{q-1}$ *los elementos distintos de 0 de* $GF(q)$.

1. *La matriz*
$$H = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 0 \\ a_1 & a_2 & \cdots & a_{q-1} & 0 & 0 \\ a_1^2 & a_2^2 & \cdots & a_{q-1}^2 & 0 & 1 \end{bmatrix}$$
es la matriz de control de paridad de un código de parámetros $[q+1, q-2, 4]$. *Equivalentemente, las columnas de* H *forman un* $(q+1)$*-arco en* $PG(2, q)$.

2. *Si* q *es par, entonces la matriz*
$$H^* = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 0 & 0 \\ a_1 & a_2 & \cdots & a_{q-1} & 0 & 1 & 0 \\ a_1^2 & a_2^2 & \cdots & a_{q-1}^2 & 0 & 0 & 1 \end{bmatrix}$$
es la matriz de control de paridad de un código de parámetros $[q+2, q-1, 4]$. *Equivalentemente, las columnas de* H^* *forman un* $(q+2)$*-arco en* $PG(2, q)$.

Demostración.

1. Es suficiente demostrar que cualesquiera 3 columnas de H son linealmente independientes. Cualesquiera 3 de las primeras $q-1$ columnas de H son una matriz de Vandermonde y, por ello, son linealmente independientes. Para cualesquiera 3 columnas que incluyan una o dos de las últimas dos columnas, el determinante debe estar expandido para dar de nuevo el determinante de una matriz de Vandermonde.

2. Hemos mostrado en la parte 1 que cualesquiera 3 columnas de H^* son linealmente independientes, con la excepción de que hayan 3 de la siguiente forma:
$$\begin{bmatrix} 1 \\ a_i \\ a_i^2 \end{bmatrix}, \begin{bmatrix} 1 \\ a_j \\ a_j^2 \end{bmatrix}, y \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}.$$
El determinante de la matriz A formada por esas 3 columnas es igual a $a_i^2 - a_j^2$. Como q es par, $GF(q)$ tiene la característica 2. Entonces, $a_i^2 - a_j^2 = (a_i - a_j)^2$. Como $a_i \neq a_j$, entonces el determinante de A es distinto de 0.

□

Corolario 7. *Se cumple que* $\text{máx}_3(3, q) \geq \begin{cases} q+1, & \text{si } q \text{ es impar,} \\ q+2, & \text{si } q \text{ es par.} \end{cases}$

Nota 11. El $(q+1)$-arco formado por las columnas de H en el Teorema 27 es la cónica $\{(x,y,z) \in PG(2,q) | yz = x^2\}$

Ahora demostraremos que los códigos/arcos dados en el Teorema 27 son optimales.

Teorema 28.

1. *Para cualquiera potencia prima q, se cumple que* $\text{máx}_3(3,q) \leq q+2$.
2. *Si q es impar, entonces* $\text{máx}_3(3,q) \leq q+1$.

Demostración.

1. Sea H una matriz de control de paridad con forma estándar par un código C de parámetros $[n, n-3, 4]$ sobre $GF(q)$, con $n = \text{máx}_3(3,q)$, tenemos que:

$$H = \begin{bmatrix} a_1 & a_2 & \cdots & a_{n-3} & 1 & 0 & 0 \\ b_1 & b_2 & \cdots & b_{n-3} & 0 & 1 & 0 \\ c_1 & c_2 & \cdots & c_{n-3} & 0 & 0 & 1 \end{bmatrix}.$$

   Como cualesquiera tres columnas de H son linealmente independientes, el determinante formado por cualesquiera 3 columnas es distinto de 0. Como el determinante formado por dos de las tres últimas columnas y una de las primeras $n-3$ columnas es distinto de 0, observamos que los valores a_i, b_i y c_i son todos distintos de 0. Multiplicando la i-ésima columna por a_i^{-1} para $i = 1, 2, \ldots, n-3$, obtenemos que C es equivalente a un código en el cual todos los valores a_i son 1. Entonces podemos suponer que:

$$A = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 0 & 0 \\ b_1 & b_2 & \cdots & b_{n-3} & 0 & 1 & 0 \\ c_1 & c_2 & \cdots & c_{n-3} & 0 & 0 & 1 \end{bmatrix},$$

   donde vemos que los b_i y c_i son distintos de 0. Como el determinante formado por la última columna y dos de las primeras $n-3$ columnas es distinto de 0, los valores de b_i son valores de $GF(q)$ diferentes entre ellos y distintos de 0. Entonces, $n-3 \leq q-1$ y por tanto, $n \leq q+2$.

2. Supongamos que q es impar y razonemos por contradicción. Asumimos entonces que existe un código C con parámetros $[q+2, q-1, 4]$. Como en 1, podemos suponer que C tiene como matriz de control de paridad a H:

$$H = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 & 0 & 0 \\ b_1 & b_2 & \cdots & b_{q-1} & 0 & 1 & 0 \\ c_1 & c_2 & \cdots & c_{q-1} & 0 & 0 & 1 \end{bmatrix},$$

   donde $b_1, b_2, \ldots, b_{q-1}$ son distintos de 0, distintos entre si, y pertenecientes a $GF(q)$, igual para los valores de $c_1, c_2, \ldots, c_{q-1}$. O sea, son todos los elementos de $GF(q)$ no nulos en cierto orden. Los determinantes distintos de 0 de la siguiente forma:

$$\det \begin{bmatrix} 1 & 1 & 1 \\ b_i & b_j & 0 \\ c_i & c_j & 0 \end{bmatrix}$$

hacen que los elementos $b_1c_1^{-1}, b_2c_2^{-1}, \ldots, b_{q-1}c_{q-1}^{-1}$, que son distintos entre si, distintos de 0 y pertenecen a $GF(q)$, vuelvan a ser los elementos de $GF(q)$ en cierto orden. Entonces, los 3 productos $\prod_{i=1}^{q-1} b_i$, $\prod_{i=1}^{q-1} c_i$ y $\prod_{i=1}^{q-1}(b_ic_i^{-1})$ tienen como resultado -1. Pero entonces:

$$\prod_{i=1}^{q-1}(b_ic_i^{-1}) = \left(\prod_{i=1}^{q-1} b_i\right)\left(\prod_{i=1}^{q-1} c_i\right)^{-1} = (-1)(-1)^{-1} = 1.$$

Como es evidente $1 \neq -1$ si q es impar, entonces se llega a la contradicción.

□

Del Corolario 7 y del Teorema 28 obtenemos el siguiente teorema.

Teorema 29. $\text{máx}_3(3,q) = \begin{cases} q+1, & \text{si } q \text{ es impar,} \\ q+2, & \text{si } q \text{ es par.} \end{cases}$

Nota 12. *Segre probó que para q impar, todo $(q+1)$-arco en $PG(2,q)$ es una cónica. Esto implica que el código optimal de parámetros $[q+1, q-2, 4]$ es único. Para q par, los $(q+2)$-arcos optimales en $PG(2,q)$ no son por lo general únicos y su clasificación no se conoce.*

La determinación de $\text{máx}_3(4,q)$, para q impar

Aquí queremos adoptar una aproximación geométrica, por ello vamos a realizar una explicación un poco más detallada sobre la terminología relacionada con los espacios proyectivos $PG(r-1,q)$. Para definir $PG(r-1,q)$ a partir del espacio vectorial $V(r,q)$, se debe recordar que los puntos y líneas de $PG(r-1,q)$ son los subespacios vectoriales de $V(r,q)$ de dimensión 1 y 2, respectivamente. Más generalmente, definimos un t-espacio en $PG(r-1,q)$ como un subespacio vectorial de $V(r,q)$ de dimensión $t+1$. Entonces, un 0-espacio representa un punto y un 1-espacio representa una recta. Por tanto, un 2-espacio sería plano y un $(r-2)$-espacio en $PG(r-1,q)$ representaría un hiperplano. Se ha de tener en cuenta que la dimensión t de un t-espacio en $PG(r-1,q)$ es siempre una menos que la correspondiente a la dimensión del espacio vectorial. Normalmente identificamos un t-espacio en $PG(r-1,q)$ con el conjunto de puntos que contiene. El número de puntos en un t-espacio es $(q^{t+1}-1)/(q-1)$, puesto que un subespacio perteneciente a $V(r,q)$ de dimensión $(t+1)$ contiene $q^{t+1}-1$ vectores distintos de 0, cada uno de los cuales tiene $q-1$ escalares múltiples distintos de 0. Un t-espacio es simplemente una copia de $PG(t,q)$ respecto a las propiedades de incidencia de sus subespacios. En particular, un gorro en $PG(r-1,q)$ debe cortar a un $(t-1)$-espacio como mucho en $\text{máx}_3(t,q)$ puntos, teniendo en cuenta que cualquier subconjunto de un gorro es un gorro.
Ahora podemos obtener una cota superior de $\text{máx}_3(4,q)$ para q impar.

Teorema 30. *Si q es impar, entonces* $\text{máx}_3(4,q) \leq q^2+1$.

Demostración. Suponemos que K es un n-gorro en $PG(3,q)$ de tamaño máximo. Sean P_1 y P_2 dos puntos cualesquiera de K y sea L la recta a la cual pertenecen los dos puntos. Como no hay 3 puntos de K que sean colineales, L no contiene otros puntos de K. A través de la recta L pasan $q+1$ planos, y cada punto de K, distinto de P_1 y P_2, aparece en solo uno de esos planos.

Como q es impar, a partir del Teorema 28(2), ningún plano puede contener mas de $q+1$ puntos de K. En particular, un plano que pasa por L puede contener como mucho $q-1$ puntos además de P_1 y P_2. Por tanto,

$$n \leq 2 + (q+1)(q-1) = q^2 + 1.$$

□

A continuación, mostraremos que existen (q^2+1) gorros en $PG(3,q)$, cuando q es impar.

Teorema 31. *Suponemos que q es impar y que b no es cuadrado en $GF(q)$. Entonces el conjunto:*

$$Q = \{(x,y,z,w) \in PG(3,q) | zw = x^2 - by^2\}$$

es un (q^2+1)-gorro en $PG(3,q)$.

Demostración. Como b no es cuadrado, el único punto de Q donde $z=0$ es $(0,0,0,1)$. Cada uno de los puntos restantes deben ser representados por un vector con $z=1$, por tanto se debe escribir

$$Q = \{(0,0,0,1), (x,y,1,x^2-by^2) | (x,y) \in V(2,q)\}.$$

Esto muestra que $|Q| = q^2+1$. Debemos mostrar que no hay 3 puntos de Q que sean colineales. Claramente $(0,0,0,1)$ no puede ser colineal con otros dos puntos de Q porque solo hay un punto en Q de la forma $(x,y,1,*)$ para cualquier par (x,y). Ahora, sean $a_1 = (x_1, y_1, 1, x_1^2 - by_1^2)$ y $a_2 = (x_2, y_2, 1, x_2^2 - by_2^2)$ dos puntos cualesquiera de Q distintos de $(0,0,0,1)$. Supongamos, por contradicción, que la recta que une a_1 y a_2 contiene un tercer punto de Q. Entonces, para algún escalar distinto de 0 λ, $a_1 + \lambda a_2 \in Q$, es decir, que el punto $(x,y,z,w) = (x_1 + \lambda x_2, y_1 + \lambda y_2, 1 + \lambda, x_1^2 - by_1^2 + \lambda x_2^2 - \lambda by_2^2)$ satisface $zw = x^2 - by^2$. Esta contradicción implica que

$$\lambda x_1^2 + \lambda x_2^2 - \lambda by_1^2 - \lambda by_2^2 = 2\lambda x_1 x_2 - 2\lambda by_1 y_2.$$

Como $\lambda \neq 0$, tenemos que

$$(x_1 - x_2)^2 = b(y_1 - y_2)^2.$$

lo que es imposible ya que b no es un cuadrado. □

Si juntamos los dos Teoremas 30 y 31 tenemos el siguiente teorema.

Teorema 32. *Si q es impar, entonces* $\text{máx}_3(4,q) = q^2 + 1$.

Nota 13.

1. El conjunto Q del Teorema 31 es un ejemplo de una cuádrica elíptica. Para q impar, cualquier cuádrica elíptica es un (q^2+1)-gorro y a la inversa, cualquier (q^2+1)-gorro es una cuádrica elíptica. Esto implica que el código óptimo de parámetros $[q^2+1, q^2-3, 4]$ es único.
2. Para $q = 2^h$, con $h > 1$, también se cumple que $\text{máx}_3(4,q) = q^2 + 1$.

Los valores de $B_q(n,4)$, para $n \leq q^2+1$

Por medio del Teorema 23, podemos traducir instantáneamente los resultados de $\text{máx}_3(r,q)$ para $r=2$ y 3 en resultados de $B_q(n,4)$.

Teorema 33. *Si q es impar, entonces*

$$B_q(n,4) = \begin{cases} q^{n-3}, & \text{para } 4 \leq n \leq q+1 \\ q^{n-4}, & \text{para } q+2 \leq n \leq q^2+1. \end{cases}$$

Si q es par, entonces

$$B_q(n,4) = \begin{cases} q^{n-3}, & \text{para } 4 \leq n \leq q+2 \\ q^{n-4}, & \text{para } q+3 \leq n \leq q^2+1. \end{cases}$$

Nota 14. Para $r=3$ y $r=4$ el problema de empaquetamiento para gorros en $PG(r-1,q)$ fue fácil de resolver debido a la existencia de configuraciones geométricas naturales llamados gorros optimales. Sin embargo, en $PG(r-1,q)$, para $r \geq 5$, los gorros no surgen de manera natural, así que el problema de empaquetamiento se vuelve aún más complicado. Como podemos ver en la figura de la sección 3.4.2, los únicos valores conocidos de $\text{máx}_3(r,q)$, para $q \neq 2$ y $r \geq 5$ son $\text{máx}_3(5,3) = 20$ y $\text{máx}_3(6,3) = 56$.
Es fácil construir 20-gorros en $PG(4,3)$ pero es difícil mostrar que 20 es el mayor tamaño posible. En contraste, es más difícil aún describir un 56-gorro en $PG(5,3)$, pero demostrar la maximalidad de 56 es sencillo y lo hicieron Bruen y Hirschfeld. Acabamos diciendo que

$$112 \leq \underset{3}{\text{máx}}(7,3) \leq 163$$

lo que sugiere que el problema de encontrar gorros optimales en $PG(6,3)$ está lejos de ser resuelto.

Este apartado ha sido escrito en base a la información extraida de las siguientes fuentes: [2], [3], [4], [6], [7], [9], [10], [11], [12], [13], [18], [21], [20], [23], [25], [28], [29], [30], [31] y [34].

Capítulo 4

Conclusión

Este trabajo pretende proporcionar al lector conocimientos básicos de la teoría de los códigos correctores de errores. A lo largo de los diferentes apartados hemos indicado de modo sucinto su historia, así como su uso actual y, por tanto, las ventajas que estos aportan. Se han explicado algunos conceptos elementales como los de: código, distancia de Hamming, código perfecto o síndrome. Y se han enunciado algunos de los teoremas más simples y fundamentales en esta teoría.

El trabajo se ha centrado en dos familias principales, los códigos correctores de errores y los códigos correctores de errores lineales. Para el estudio de estos últimos, se han introducido conceptos como los de: clase, líder de clase, matriz estándar, matriz generadora o matriz de control de paridad. Con la información anterior, se puede concluir que los códigos correctores de errores tienen un papel esencial en la transmisión de información, especialmente en canales con mucho ruido. Esto se debe a que permiten tener la seguridad de que el mensaje va a llegar correctamente al receptor. El hecho de que la tecnología esté avanzando implica que estos códigos seguirán teniendo un papel indispensable debido, por un lado, a que el número de interconexiones a través de canales con ruido continuará aumentando y, por otro, a su interés en el campo de los ordenadores cuánticos.

Bibliografía

[1] Anderson, I. (1974). A first course in combinatorial mathemathics. Clarendon Press. Oxford.

[2] Barlotti, A. (1955). Un estensione del teorema di Segre-Kustaanheimo. Bolletino dell'Unione Matematica Italiana 10, 498-506.

[3] Bose, Raj Chandra. (1947). Mathematical theory of the symmetrical factorial design. Sankhya 8, 107-166.

[4] Bruen, A. A. y Hirschfeld, J. W. P. (1978). Application of line geometry over finite fields II. The Hermitian surface. Geometriae Dedicata. 7, 333-353.

[5] Cover, T. M. y Thomas, J. A. (1991). Elements of information theory. New York: John Wiley & Sons.

[6] Fenton, N. E. y Vámos, P. (1982). Matroid interpretation of maximal k-arcs in projective spaces. Rend. Mat. 2, 573-580.

[7] Games, R. A. (1983). The packing problem for projective geometries over GF(3) with dimension greater than five. Journal of Combinatorial Theory 35, 126-144.

[8] Hall, M. (1967). Combinatorial theory. Wiley-Interscience.

[9] Helgert, H. J. y Stinaff, R. D. (1973) Minimum-distance bounds for binary linear codes. IEEE Transactions on Information Theory, vol. 19, 344-356.

[10] Hill, R. (1973). On the largest size of cap in $S_{5,3}$. Atti Accad. Naz. Lincei Rend, 54, 378-384.

[11] Hill, Raymond. (1978). Caps and codes. Discrete Mathematics, 22, 111-137.

[12] Hirschfeld, J. W. P. (1979). Projective geometries over finite fields. Oxford Unversity Press.

[13] Hirschfeld, James. (1983). Maximum sets in finite projective spaces. In Surveys in combinatorics, Society Lecture Note Series. Cambridge University Press, 55-76.

[14] Huffman, W. C. , Kim, J-L. y Solé, P. (2021). Concise Encyclopedia of Coding Theory. Chapman and Hall.

[15] Lidl, R. y Niederreiter, H. (1983). Finite fields. Cambridge University Press.

[16] Lidl, R. y Pilz, G. (1997). Applied abstract algebra. Springer.

[17] Ling, S. y Xing, C. (2004). Coding Theory: A first course. 1st Edition. Cambridge University Press.

[18] van Lint, J. H. (1982). Introduction to coding theory. Springer-Verlag.

[19] Mackenzie, C. y Seberry, J. (1984). Maximal ternary codes and Plotkin's bound. Ars Combinatoria, 17A, 251-270.

[20] MacWilliams, F.J. y Sloane, N.J.A. (1997). The theory of error-correcting codes(North-Holland Mathematical Library, Volume 16). North Holland Publishing.

[21] McEliece, R. y Truss, J. K. (1977). The theory of information and coding. Addison-Wesley.

[22] Nordstrom, A. W. y Robinson, J. P. (1967). An optimum non-linear code. Inf. Control, 11, 613-616.

[23] Pellegrino, G. (1970). Sul massimo ordine delle calotte in $S_{4,3}$. Matematiche (Catania), 25, 1-9.

[24] Plotkin, M. (1960). Binary codes with specified minimum distance. IEEE Transactions on Information Theory, 6, 445-450.

[25] Qvist, B. (1952). Some remarks concerning curves of the second degree in a finite plane. Annales Academiae scientiarum Fennicae: Mathematica-Physica, no 134.

[26] Roman, S. (1992). Coding and information theory (Graduate Texts in Mathematics GTM, volume 134). Springer-Verlag.

[27] Roth, R. (2006). Introduction to coding theory. Cambridge University Press.

[28] Segre, B. (1954). Sulle ovali nei piani lineari finiti. Rend. dellí Acc. Nazionale dei Lincei, 17, 141-142.

[29] Shannon, C. E. (1948). A mathematical theory of communication. Bell System Technical Journal, 27, 379-423 y 623-656.

[30] Slepian, D. (1960). Some further theory of group codes. Bell System Technical Journal, 39, 1219-1252.

[31] Sloane, N. J. A. (1982). Recent bounds for codes, sphere packings and related problems obtained by linear programing and other methods. Contemporary Mathematics, 9, 153-185.

[32] Tena A., J. G. y Munuera G., C. (1997). Codificación de la información. Ediciones Universidad de Valladolid.

[33] Tietäväinen, A. (1980). Bounds for binary codes just outside the Plotkin range. Info. Control, 47, 85-93.

[34] Verhoeff, T. (1985). An updated table of minimum-distance bounds for binary linear codes. IEEE Transactions on Information Theory, 39, 662-677.