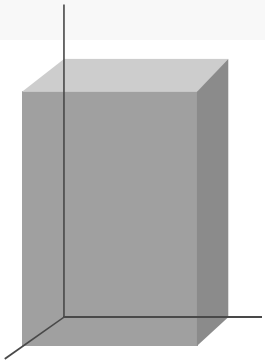
CAPÍTULO 2



APLICACIONES DE ECUACIONES LINEALES Y MATRICES (OPCIONAL)

2.1 INTRODUCCIÓN A LA TEORÍA DE CÓDIGOS

Requisito. El material sobre sistemas binarios que se analizó en el capítulo 1.

En la actual sociedad global, la comunicación abunda en el comercio, el gobierno, la investigación y la educación. Los datos se transmiten de un punto a otro o se registran en formas diversas para representar imágenes de vídeo, sonido, o combinaciones de éstas. Sin importar la distancia que deba recorrer la transmisión, el proceso básico es el mismo. La información debe enviarse y recibirse, y cabe la posibilidad de que ocurra una distorsión. Los datos recibidos deben verificarse de alguna manera para (en el mejor de los casos) detectar errores en la transmisión.

La codificación es una rama de la teoría de la información y la comunicación, que ha desarrollado técnicas para contribuir a detectar y, en algunos casos, corregir errores. Esta disciplina se apoya en diversos campos de las matemáticas, incluyendo álgebra lineal y abstracta, teoría de números, probabilidad y estadística, y combinatoria. En esta sección se presentará una breve introducción a la codificación, en la que utilizaremos el álgebra lineal.

El aspecto clave de la transmisión de datos radica en que se lleve a cabo de manera rápida y barata. Con esto en mente, es razonable considerar un proceso "abreviado" (por ejemplo, omitir ciertas letras de las palabras). Desafortunadamente, cualquier ahorro de tiempo que se derive de un procedimiento de ese tipo, se compensa con un aumento de la posibilidad de que los datos se interpreten de manera incorrecta. Casi todas las técnicas de codificación funcionan a la inversa de una abreviación. Esto es, se envían más datos de los normales como una forma de detectar posibles errores en la transmisión. Esencialmente, la teoría de códigos se basa en una cuidadosa selección de qué debe incluirse en la codificación y cómo hacerlo.

CODIFICACIÓN DE INFORMACIÓN BINARIA Y DETECCIÓN DE ERRORES

Un **mensaje** es una secuencia finita de caracteres de un alfabeto. Elegiremos como nuestro alfabeto el conjunto $B = \{0, 1\}$. Todo carácter, número o símbolo que necesitemos transmitir se representará con un m-vector binario. Esto es, cada carácter, número o símbolo se representará en forma binaria. De acuerdo con lo anterior, los mensajes consistirán de un conjunto de **palabras**, cada una de las cuales será un m-vector binario. El conjunto de todos los m-vectores binarios se denota mediante B^m.

Como vimos en el capítulo 1, los vectores binarios y las matrices binarias comparten las mismas propiedades que los vectores y matrices reales (de base 10), salvo que para los cálculos relacionados con aquellos utilizamos aritmética de base 2. (Vea las tablas 1.1 y 1.2 de la sección 1.2.) Un m-vector binario tiene la forma $[b_1,\ b_2\ \cdots\ b_m]$ o $[b_1\ b_2\ \cdots\ b_m]^T$, donde cada b_j es 0 o 1. Al codificar suele omitirse la notación matricial, por lo que el m-vector binario se escribe como una cadena de bits en la forma $b_1b_2\cdots b_m$. Cuando se utiliza el álgebra matricial, las expresiones se escriben en la forma matricial estándar.

La figura 2.1 muestra los procesos básicos de envío de una palabra, de un punto a otro de un canal de transmisión. Un vector $\mathbf{x}$ en B^m se envía a través de un canal de transmisión, y se recibe como el vector $\mathbf{x}_t$ en B^m. En la práctica, el envío puede verse afectado por perturbaciones —que por lo general se denominan **ruido**— durante su trayecto por el canal de transmisión. Tal problema puede deberse a problemas eléctricos, electricidad estática, interferencia climática, etcétera. Cualquiera de estas condiciones puede causar que un 0 sea recibido como 1, o viceversa. La transmisión errónea de bits en el mensaje enviado da lugar a que la palabra recibida sea diferente de la original; esto es, $\mathbf{x} \neq \mathbf{x}_t$. De presentarse este tipo de errores, $\mathbf{x}_t$ podría ser cualquier vector en B^m.

Figura 2.1 ▶

En la transmisión de información, la tarea básica consiste en reducir la probabilidad de recibir palabras que difieran de la que se envió. Esto se puede lograr de la manera siguiente. Primero elegimos un entero $n > m$ y una función inyectiva e de B^m a B^n, esto es, cualesquiera sean $\mathbf{x}$ y $\mathbf{y}$ en B^m, $\mathbf{x} \neq \mathbf{y}$ implica que $e(\mathbf{x}) \neq e(\mathbf{y})$. De esta manera, a palabras diferentes en B^m corresponden n-vectores diferentes en B^n. La función e se denomina **función de codificación**.

EJEMPLO 1 Sean $m = 2$, $n = 3$ y $e(b_1b_2) = b_1b_2b_3$, donde b_3 se define como 0 ($b_3 \equiv 0$). Tenemos

$$e(00) = 000, \quad e(01) = 010, \quad e(10) = 100, \quad e(11)\ 110,$$

y concluimos que la función e es inyectiva. La función e puede calcularse mediante una multiplicación por la matriz.

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

Así, e es una transformación matricial de B^2 a B^3, dada por

$$e(b_1b_2) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ 0 \end{bmatrix}.$$

■

EJEMPLO 2 Sean $m = 2$, $n = 3$ y $e(b_1b_2) = b_1b_2b_3$, donde b_3 se define como $b_1 + b_2$ ($b_3 \equiv b_1 + b_2$). Tenemos

$$e(00) = 000, \quad e(01) = 011, \quad e(10) = 101, \quad e(11)\ 110,$$

y concluimos que la función e es inyectiva. La función e puede calcularse mediante una multiplicación por la matriz

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

De esta manera, e es una transformación matricial de B^2 a B^3 dada por

$$e(b_1b_2) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_1 + b_2 \end{bmatrix}.$$

■

EJEMPLO 3 Sean $m = 2$, $n = 3$ y $e(b_1b_2) = b_100$. Tenemos

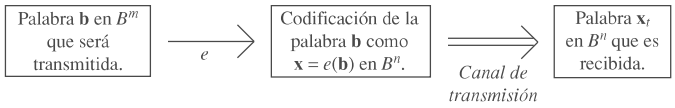
$$e(00) = 000, \quad e(01) = 000, \quad e(10) = 100, \quad e(11) = 100,$$

y concluimos que la función e no es inyectiva. Esta función e es una transformación matricial, ya que

$$e(b_1b_2) = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ 0 \\ 0 \end{bmatrix}.$$

La función inyectiva e de B^m a B^n, $n > m$, se denomina **función de codificación (m, n)** y puede considerarse como un medio para representar cada palabra en B^m como una palabra única en B^n. En el caso de una palabra $\mathbf{b}$ en B^m, $e(\mathbf{b})$ se llama **palabra codificada** o **palabra de código** que representa a $\mathbf{b}$. Los n-m bits adicionales del código pueden utilizarse para detectar errores en la transmisión y, algo más sorprendente, también para ayudar a corregirlos. La figura 2.2 ilustra los dos pasos utilizados para la transmisión: primero se codifica la palabra original con la función e, y luego se transmite la palabra código. Si el canal de transmisión carece de ruido, $\mathbf{x}_t = \mathbf{x}$ para toda $\mathbf{x}$ en B^n. Dado que la función de codificación e es conocida, se puede determinar la palabra original $\mathbf{b}$.

Figura 2.2 ►



En general, los errores de transmisión no pueden evitarse. Diremos que la palabra código $\mathbf{x} = e(\mathbf{b})$ se ha transmitido **con k o menos errores** si $\mathbf{x}$ y $\mathbf{x}_t$ difieren en al menos 1 pero no más de k bits.

Sea e una función de codificación (m, n). Decimos que e **detecta k o menos errores** si cada vez que $\mathbf{x} = e(\mathbf{b})$ se transmite con k o menos errores, $\mathbf{x}_t$ no es una palabra código (así, $\mathbf{x}_t$ no puede ser $\mathbf{x}$ y, por lo tanto, $\mathbf{x}$ no se ha transmitido de manera correcta).

Observación Incluso si la función de codificación e está diseñada para incorporar capacidad de detección de errores, éstos pueden ocurrir.

EJEMPLO 4 Suponga que estamos interesados en transmitir un solo bit. Esto es, transmitiremos 0 o 1. Una manera de protegerse contra errores en la transmisión, consiste en emitir el mensaje más de una vez. Por ejemplo, podríamos utilizar la función de codificación e (1, 3),

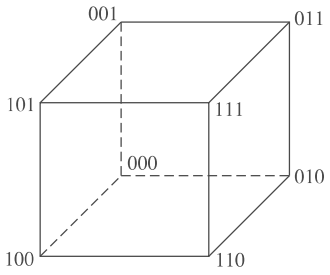
de modo que 0 se codifique como 000 y 1 como 111. En términos de una transformación matricial tendríamos, para un solo bit b,

$$e(b) = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} [b] = \begin{bmatrix} b \\ b \\ b \end{bmatrix}.$$

En consecuencia, sólo hay dos palabras de código válidas, 000 y 111. Si $\mathbf{x} = bbb$ se transmite de modo que la palabra recibida es $\mathbf{x}_t = 001$, esto significa que ocurrió por lo menos un error. Asimismo, si recibiéramos 001, 110 o 010, podríamos concluir que se presentaron errores de transmisión, ya que éstas son palabras código no válidas. Los únicos casos en que resulta imposible detectar errores, ocurren cuando $\mathbf{x} = 000$ pero $\mathbf{x}_t = 111$, o viceversa. Como la función de codificación detecta 2 errores o menos, decimos que e es una función de codificación con capacidad para **detectar un doble error**.

Suponga que además de detectar errores queremos corregirlos. Si $\mathbf{x}_t = 010$, sabemos que ha ocurrido un error en la transmisión, pero ignoramos si éste fue un solo error o un error doble. Si $\mathbf{x} = 000$, esto significa que ocurrió un error, pero si $\mathbf{x} = 111$, sabemos que ocurrieron dos. Una estrategia de corrección en este caso parte de suponer que la ocurrencia de un error es más probable que la ocurrencia de dos. En consecuencia, $\mathbf{x}_t = 010$ se "corrige" como 000. La figura 2.3 ilustra este procedimiento de corrección. Si $\mathbf{x}_t = b_1b_2b_3$, se decodifica —de acuerdo con la figura 2.3— como 000 si podemos movernos del vértice $b_1b_2b_3$ a 000 a lo largo de una sola arista; de otra forma se decodifica como 111. Con esta estrategia, por lo tanto, tenemos un **código de corrección de un solo error**. Pero observe que si $\mathbf{x} = 000$ y $\mathbf{x}_t = 011$, con dicha estrategia decodificaríamos de manera incorrecta $\mathbf{x}_t$ como 111. ■

Figura 2.3 ►



Al procedimiento del ejemplo 4 suele denominársele código de repetición triple. Vea también el ejercicio T.4.

DEFINICIÓN Dado un n-vector $\mathbf{x}$, el número de unos (1) en $\mathbf{x}$ se denomina peso de $\mathbf{x}$, y se denota mediante $|\mathbf{x}|$.

EJEMPLO 5 Determine el peso de cada una de las palabras siguientes en B^6.

(a) $\mathbf{x} = 011000$; $|\mathbf{x}| = 2$
(b) $\mathbf{x} = 000001$; $|\mathbf{x}| = 1$
(c) $\mathbf{x} = 000000$; $|\mathbf{x}| = 0$
(d) $\mathbf{x} = 101010$; $|\mathbf{x}| = 3$ ■

DEFINICIÓN La función de codificación e, de B^m a B^{m+1}, dada por

$$e(\mathbf{b}) = e(b_1b_2 \cdots b_m) = b_1b_2 \cdots b_mb_{m+1} = \mathbf{b}_t,$$

donde

$$b_{m+1} = \begin{cases} 0, & \text{si } |\mathbf{b}| \text{ es par} \\ 1, & \text{si } |\mathbf{b}| \text{ es impar} \end{cases}$$

se denomina **función de codificación de paridad $(m, m + 1)$** o **código de verificación de paridad $(m, m + 1)$**. Si $b_{m+1} = 1$, decimos que $\mathbf{b}_t$ tiene paridad impar, y si $\mathbf{b}_{m+1} = 0$, decimos que $\mathbf{b}_t$ tiene paridad par.

EJEMPLO 6 Si $m = 3$, el código de verificación (3, 4) produce las palabras codificadas,

$$e(000) = 0000, \quad e(001) = 0011, \quad e(010) = 0101, \quad e(100) = 1001,$$
$$e(101) = 1010, \quad e(110) = 1100, \quad e(011) = 0110, \quad e(111) = 1111.$$

Si el canal de transmisión transmite $\mathbf{x} = 101$ de B^3 como $\mathbf{x}_t = 1011$, el peso de $\mathbf{x}_t$ es $|\mathbf{x}_t| = 3$. Sin embargo, como $|101| = 2$ y $\mathbf{x}_t$ tiene paridad impar, sabemos que ocurrió un número impar de errores (al menos uno). Si la palabra recibida hubiera sido $\mathbf{x}_t = 1010$, $|\mathbf{x}_t| = 2$ y $\mathbf{x}_t$ tiene paridad par. En este caso, no podemos concluir que la palabra código esté libre de errores. ■

Observación El código de verificación de paridad $(m, m + 1)$ detecta un número impar de errores, pero no detecta un número par de errores. A pesar de esta limitación, este código se utiliza con mucha frecuencia.

Términos clave

Mensaje
Palabras
Ruido
Función de codificación
Función de codificación (m, n)
Palabra de código (o palabra codificada)
Detección de k o menos errores
Función de codificación para detectar doble error
Función de codificación para detectar un solo error
Función de codificación de paridad $(m, m + 1)$ [o código de verificación de paridad $(m, m + 1)$]

2.1 Ejercicios

Todas las operaciones aritméticas de esta sección deben realizarse por medio de aritmética binaria.

1. Sea e la función de B^3 a B^4 dada por

$$e(b_1b_2b_3) = b_1b_2b_3b_4,$$

donde $b_4 = b_1 + b_3$.

(a) ¿La función e es inyectiva? Si no lo es, determine dos vectores diferentes $\mathbf{b}$ y $\mathbf{c}$ en B^3, tales que $e(\mathbf{b}) = e(\mathbf{c})$.

(b) Determine la matriz A de manera que e pueda escribirse como una transformación matricial en la forma

$$e(b_1b_2b_3) = A\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}.$$

2. Sea e la función de B^3 a B^4 dada por

$$e(b_1b_2b_3) = b_1b_2b_3b_4,$$

donde $b_4 = 0$.

(a) ¿La función e es inyectiva? Si no lo es, determine dos vectores diferentes $\mathbf{b}$ y $\mathbf{c}$ en B^3, tales que $e(\mathbf{b}) = e(\mathbf{c})$.

(b) Determine la matriz A de manera que e pueda escribirse como una transformación matricial en la forma

$$e(b_1b_2b_3) = A\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ 0 \end{bmatrix}.$$

3. Sea e la función de B^3 a B^2 dada por

$$e(b_1b_2b_3) = b_1b_2.$$

(a) ¿La función e es inyectiva? Si no lo es, determine dos vectores diferentes $\mathbf{b}$ y $\mathbf{c}$ en B^3, tales que $e(\mathbf{b}) = e(\mathbf{c})$.

(b) Determine la matriz A de manera que e pueda escribirse como una transformación matricial en la forma

$$e(b_1b_2b_3) = A\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}.$$

4. Sea e la función de B^2 a B^4 dada por

$$e(b_1b_2) = b_1b_2b_3,$$

donde $b_3 = b_1 \times b_2$.

(a) ¿La función e es inyectiva? Si no lo es, determine dos vectores diferentes **b** y **c** en B^2, tales que $e(\mathbf{b}) = e(\mathbf{c})$.

(b) Determine, si existe, la matriz A de manera que e pueda escribirse como una transformación matricial en la forma

$$e(b_1b_2) = A\begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_1 \times b_2 \end{bmatrix}.$$

5. Determine el peso de cada una de las palabras siguientes.

(a) 01110 (b) 10101 (c) 11000 (d) 00010

6. Determine el peso de cada una de las palabras dadas.

(a) 101 (b) 111 (c) 011 (d) 010

7. Determine la paridad de cada una de las palabras siguientes en B^4.

(a) 1101 (b) 0011 (c) 0100 (d) 0000

8. Determine la paridad de cada una de las palabras siguientes en B^5.

(a) 01101 (b) 00011 (c) 00010 (d) 11111

9. Se utiliza un código de verificación de paridad (4, 5) y se reciben las palabras siguientes. Determine si se detectaría un error.

(a) 10100 (b) 01101 (c) 11110 (d) 10000

10. Se utiliza un código de verificación de paridad (5, 6) y se reciben las palabras siguientes. Determine si se detectaría un error.

(a) 001101 (b) 101110 (c) 110000 (d) 111010

11. (a) Determine las palabras código para el código de verificación de paridad (2, 3).

(b) Determine si se detectará un error al recibir cada una de las palabras siguientes.

(i) 011 (ii) 111 (iii) 010 (iv) 001

Ejercicios teóricos

T.1. Determine el número de palabras con peso cero en B^2; con peso uno; con peso dos.

T.2. Determine el número de palabras con peso cero en B^3; con peso uno; con peso dos; con peso tres.

T.3. Determine el número de palabras con peso uno y con peso dos en B^n.

T.4. Sea e una función de codificación (m, n) que detecta k o menos errores. Decimos que e produce un **código de corrección de error**. Un código de corrección de error es **lineal** si la suma (o diferencia) de cualesquiera dos palabras código es también una palabra código.

(a) Demuestre que el código de corrección de error del ejemplo 4 es lineal.

(b) Demuestre que el código de corrección de error del ejercicio 11 es lineal.

T.5. Sea e la función de B^2 a B^4 dada por la transformación matricial siguiente:

$$e(b_1b_2) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} b_1 \\ b_2 \end{bmatrix}.$$

(a) Determine todas las palabras código.

(b) ¿Este código es lineal?

(c) Como todas las palabras código tienen la misma paridad, si utilizáramos una verificación de paridad sobre la palabra que se recibe, ¿esta verificación detectaría todos los errores posibles? Explique su respuesta.

Ejercicios con MATLAB

El ejercicio ML.1 tiene que ver con matrices binarias y con los comandos adicionales que se describen en la sección 12.9.

ML.1. Por medio de las instrucciones siguientes, desarrolle las palabras código para el código de verificación de paridad (4, 5).

(a) Utilice el comando $M =$ **bingen**(**0**, **15**, **4**) para generar una matriz cuyas columnas sean todos los vectores en B^4.

(b) Utilice el comando $s =$ **sum**(**M**) para calcular un vector cuyas entradas sean los pesos de las columnas de la matriz M.

(c) Construya un vector binario, **w**, de 1×16, cuyas entradas sean la paridad de las columnas de la matriz M.

(d) Construya las palabras código del código de verificación de paridad (4, 5) mostrando la matriz $C = [\mathbf{M}; \mathbf{w}]$.