Error-Detecting and Error-Correcting Codes

Text Reference: Section 4.6, p. 259

In this set of exercises, we examine how we can construct a method for detecting and correcting errors made in the transmission of encoded messages. It will turn out that abstract vector spaces and the concepts of null space, rank, and dimension are needed for this construction.

When a message is transmitted, it has the potential to get scrambled by noise. This is certainly true of voice messages, and is also true of the **digital** messages that are sent to and from computers. Now even sound and video are being transmitted in this manner. By a digital message, we mean a sequence of 0's and 1's which encodes a given message. What we will seek to do is to add more data to a given binary message that will help to detect if an error has been made in the transmission of the message; adding such data is called an **error-detecting code**. We will also try to add data to the original message so that we can detect if errors were made in transmission, and also to figure out what the original message was from the possibly corrupt message that we received. This type of code is an **error-correcting code**.

A common type of error-detecting code is called a **parity check**. For example, consider the message 1101. We add a 0 or 1 to the end of this message so that the resulting message has an even number of 1's. We would thus encode 1101 as 11011. If the original message were 1001, we would encode that as 10010, since the original message already had an even number of 1's. Now consider receiving the message 10101. Since the number of 1's in this message is odd, we know that an error has been made in transmission. However, we do not know how many errors happened in transmission or which digit(s) were effected. Thus a parity check scheme detects errors, but does not locate them for correction.

Example: The United States Postal Service uses a code to express the zip code on a letter as a series of long and short bars. The digits are coded as follows:

Zip codes are encoded and placed on the envelope. A long bar begins and ends each code. An additional parity check digit is encoded. This digit, when added to those in the five-digit zip code, produces a number which is a multiple of ten. If the six encoded digits do not add to a multiple of ten, then an error in transmission must have occurred. Thus the zip codes 29733 and 28209 become

29733: 

28209: 

Since $2+9+7+3+3=24$, and $24+6=30$, a 6 was added to the code for 29733; likewise a 9 was added to the code for 28209, since $2+8+2+0+9+9=30$.

In order to discuss error-correcting codes, we will restrict our attention to digital sequences: messages of 0's and 1's. We define the set $\mathbb{Z}_2$ to be the set $\{0,1\}$. It will first be useful to do arithmetic on $\mathbb{Z}_2$. We will add and multiply 0 and 1 as given in the following tables:

$+$	0	1
0	0	1
1	1	0

$\cdot$	0	1
0	0	0
1	0	1

One may check that these operations have the familiar properties of addition and multiplication of real numbers. One peculiarity is the fact that since $1+1=0$, $1=-1$. That is, 1 is its own additive inverse, and thus subtraction is exactly the same as addition in $\mathbb{Z}_2$.

We will now express messages as column vectors of elements of $\mathbb{Z}_2$. The messages 1001 and 1101 would be expressed as

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \text{ and } \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

We will assume that each message is n digits long; we will call the set of all possible messages of length n digits $\mathbb{Z}_2^n$. In other words, $\mathbb{Z}_2^n$ is the set of all vectors with n elements taken from $\mathbb{Z}_2$. The set $\mathbb{Z}_2^4$ contains the following sixteen vectors:

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

We can add these vectors just as we do in $\mathbb{R}^n$; we can also multiply these vectors by scalars taken from $\mathbb{Z}_2$.

Examples:

$$\begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \qquad 1 \cdot \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

In fact, if we use $\mathbb{Z}_2$ as scalars, and use the operations of vector addition and scalar multiplication as given in the last examples, then $\mathbb{Z}_2^n$ is a vector space. We say that $\mathbb{Z}_2^n$ **is a vector space over** $\mathbb{Z}_2$ to emphasize that the scalars we use are taken from $\mathbb{Z}_2$. The material in Sections 4.2 to 4.6 on matrices of real numbers also applies to matrices whose entries are taken from $\mathbb{Z}_2$, except that all arithmetic is done in $\mathbb{Z}_2$.

Example: To find a basis for the column space, a basis for the null space, and the rank of

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

we first row reduce A using $\mathbb{Z}_2$ arithmetic (remember that $1 + 1 = 0$):

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

A basis for $\operatorname{Col} A$ is the pivot columns in A:

$$\left\{ \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \right\}$$

Thus $\operatorname{rank} A = 2$. To find a basis for $\operatorname{Nul} A$, solve $A\mathbf{x} = \mathbf{0}$ and get the equations

$$x_1 = -1x_3 - 1x_4 \text{ and } x_2 = -1x_3 - 1x_4.$$

Since $-1 = 1$, we may write these as

$$x_1 = 1x_3 + 1x_4 \text{ and } x_2 = 1x_3 + 1x_4,$$

so a basis for $\operatorname{Nul} A$ would be

$$\left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} \right\}$$

Notice that these results differ from those we would get if we treated A as a matrix of real numbers; you may confirm that in that case $\operatorname{rank} A = 3$.

We can list all of the members of $\operatorname{Nul} A$:

$$\operatorname{Nul} A = \left\{ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \right\}$$

and note that the number of vectors in $\operatorname{Nul} A$ is $4 = 2^2$, which is 2 raised to the dimension of $\operatorname{Nul} A$. This is true for any subspace of $\mathbb{Z}_2^n$:

Fact: If W is a subspace of $\mathbb{Z}_2^n$ with $\dim W = k$, then the number of vectors in W is equal to 2^k.

Let us assume that our messages are each 4 digits long. We will now describe how to to create a self-correcting code for these messages. We want to do a more sophisticated version of the parity check; we will add three numbers to the end of each 4 digit message. Thus the encoded messages will be elements of $\mathbb{Z}_2^7$. To begin, consider the matrix

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

Notice that the columns in H, which we will call $\mathbf{h}_1$, $\mathbf{h}_2$, $\ldots \mathbf{h}_7$, happen to be all of the non-zero members of $\mathbb{Z}_2^3$. As above we can find a basis for the null space of H:

$$\left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \right\}$$

Since the dimension of $\operatorname{Nul} H$ is 4, by our earlier fact $\operatorname{Nul} H$ contains 16 vectors. Of course, $\mathbb{Z}_2^4$ also contains 16 vectors, so we can encode each vector in $\mathbb{Z}_2^4$ using a different vector in $\operatorname{Nul} H$. For that reason we will call the null space of H the **Hamming (7,4) code**. To encode the vectors in $\mathbb{Z}_2^4$, we form a matrix A whose columns are the basis elements for $\operatorname{Nul} H$; the matrix A will be our encoding matrix.

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

Example: To encode the message 1101, we compute

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

Notice that since the first four rows of A are the identity matrix, multiplication by A merely adds three digits to the original message.

The matrix H was chosen because its nullspace has some very interesting properties which allow us to detect and correct single errors in transmitted messages. We assume at this point that any transmitted message has at most one error in transmission. If the probability of an error in transmission is small, then this is a reasonable assumption. We consider the standard basis vectors $\mathbf{e}_1$, $\mathbf{e}_2$, $\ldots \mathbf{e}_7$ in $\mathbb{Z}_2^7$:

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \ldots \mathbf{e}_7 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Notice that adding one of these vectors to an encoded message vector $\mathbf{x}$ is equivalent to making a single error in the transmission of $\mathbf{x}$. Notice also that the vectors $\mathbf{e}_1$, $\mathbf{e_2}$, $\ldots \mathbf{e_7}$ are not in the nullspace of H, for $H\mathbf{e}_i = \mathbf{h}_i \neq \mathbf{0}$. In fact, we have the following theorem.

Theorem 1 *If H is the matrix given above, and if $\mathbf{x}$ is in $\operatorname{Nul} H$, then $\mathbf{x} + \mathbf{e_i}$ is not in $\operatorname{Nul} H$.*

Proof: Since $\mathbf{x}$ is in $\operatorname{Nul} H$, $H\mathbf{x} = \mathbf{0}$. By the above note, we know that $H\mathbf{e}_i = \mathbf{h}_i \neq \mathbf{0}$. Thus

$$H(\mathbf{x} + \mathbf{e_i}) = H\mathbf{x} + H\mathbf{e_i} = \mathbf{0} + \mathbf{h}_i = \mathbf{h_i} \neq \mathbf{0},$$

and $\mathbf{x} + \mathbf{e_i}$ is not in $\operatorname{Nul} H$.

This result means that if a single error is made in the transmission of a message $\mathbf{x}$, then we can detect that error by checking to see whether the received message lies in $\operatorname{Nul} H$.

Example: If we received the message 0100101, we can check that

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

Since our message vector is in $\operatorname{Nul} H$ we know that no single transmission error has happened. If a single error had happened, the theorem tells us that the resulting message vector would not be in $\operatorname{Nul} H$.

Example: If we received the message 0111001, we can check that

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

Thus (assuming that at most one error in transmission has been made) we know that a single transmission error has happened.

So the Hamming (7,4) code is an error-detecting code. The following theorem will show us that it is also an error-correcting code.

Theorem 2 *If H is the matrix given above, and if $H\mathbf{x} = \mathbf{h}_i$, then $\mathbf{x} + \mathbf{e_i}$ is in* Nul H*, and $\mathbf{x} + \mathbf{e_j}$ is not in* Nul H *for $j \neq i$.*

Proof: Suppose that $H\mathbf{x} = \mathbf{h}_i$. Then

$$H(\mathbf{x} + \mathbf{e_i}) = H\mathbf{x} + H\mathbf{e_i} = \mathbf{h}_i + \mathbf{h}_i = \mathbf{0}.$$

Likewise if $i \neq j$,

$$H(\mathbf{x} + \mathbf{e_j}) = H\mathbf{x} + H\mathbf{e_j} = \mathbf{h}_i + \mathbf{h}_j \neq \mathbf{0}.$$

Suppose we receive a message $\mathbf{x}$ that has had a single error happen in transmission. By Theorem 1, $H\mathbf{x} \neq \mathbf{0}$, so $H\mathbf{x} = \mathbf{h}_i$ for some i. The result in Theorem 2 implies that the single error in transmission must have occurred to the i^{th} digit; changing this digit (by adding $\mathbf{e}_i$ to $\mathbf{x}$) will give us a vector in Nul H, and thus a properly encoded vector. Changing any other digit in $\mathbf{x}$ will not give us a vector in Nul H.

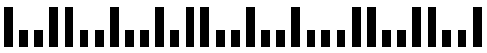
Example: The message 0111001 was in error by a previous example. In fact, we found that

$$H\begin{bmatrix}0\\1\\1\\1\\0\\0\\1\end{bmatrix} = \begin{bmatrix}0&0&0&1&1&1&1\\0&1&1&0&0&1&1\\1&0&1&0&1&0&1\end{bmatrix}\begin{bmatrix}0\\1\\1\\1\\0\\0\\1\end{bmatrix} = \begin{bmatrix}0\\1\\0\end{bmatrix} = \mathbf{h}_2.$$

By Theorem 2, the single error in transmission must have occurred at the second digit. Thus the true message which was sent is 0011001.

Questions:

1. The following United States Postal Service codes were found on envelopes; determine whether an error was made in transmission.

   a) 

   b) 

   c) 

2. Consider the following vectors.

$$\mathbf{a} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \text{ and } \mathbf{c} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

   Compute the following.

   a) $\mathbf{a} + \mathbf{b}$

   b) $\mathbf{c} - \mathbf{b} + \mathbf{a}$

3. Let $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$ be as in Question 2. Is the set $\{\mathbf{a}, \mathbf{b}, \mathbf{c}\}$ linearly independent or linearly dependent?

4. Find a basis for the column space, a basis for the null space, and the rank of

$$B = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

5. Encode the following messages using the Hamming (7,4) code.

   a) 1001

   b) 0011

   c) 0101

6. Each of the following messages has been received, and each had been encoded using the Hamming (7,4) code. During transmission at most one element in the vector was changed. Either determine that no error was made in transmission, or find the error made in transmission and correct it.

   a) 0101101

   b) 1000011

   c) 0010111

   d) 0101010

   e) 0111100

   f) 1001101

   g) 1010010

   h) 1110111