You can also view this case study in the following formats:

Mathematica Maple

Error-Detecting and Error-Correcting Codes

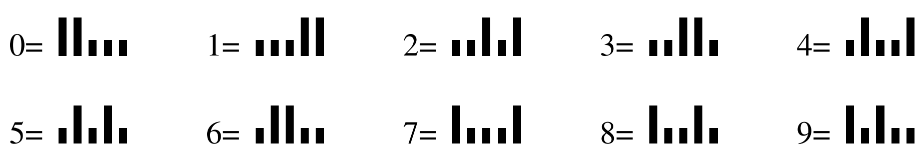
Text Reference: Section 4.6, p. 265

In this set of exercises, a method for detecting and correcting errors made in the transmission of encoded messages is constructed. 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. A digital message is a sequence of 0's and 1's which encodes a given message. More data will be added 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**. More data may also be added to the original message so that errors made in transmission may be detected, and also to figure out what the original message was from the possibly corrupt message that was 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. Add a 0 or 1 to the end of this message so that the resulting message has an even number of 1's. The message 1101 would thus be encoded as 11011. If the original message were 1001, it would be encoded 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, an error has been made in transmission. However, it is not known 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: $\underbrace{}_{2}\;\underbrace{}_{9}\;\underbrace{}_{7}\;\underbrace{}_{3}\;\underbrace{}_{3}\;\underbrace{}_{6}$

28209: $\underbrace{}_{2}\;\underbrace{}_{8}\;\underbrace{}_{2}\;\underbrace{}_{0}\;\underbrace{}_{9}\;\underbrace{}_{9}$

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, attention will be restricted to digital sequences: messages of 0's and 1's. The set $\mathbb{Z}_2$ to be the set $\{0,1\}$. It will first be useful to do arithmetic on $\mathbb{Z}_2$. Addition and multiplication for 0 and 1 are 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 $\mathbb{Z}_2$.

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

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

Assume that each message is n digits long; 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:

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

These vectors can be added just as in $\mathbb{R}^n$; these vectors may also be multiplied 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 $\mathbb{Z}_2$ are the scalars, and the operations of vector addition and scalar multiplication as given in the last examples are used, then $\mathbb{Z}_2^n$ is a vector space: to make clear that $\mathbb{Z}_2$ are the scalars, $\mathbb{Z}_2^n$ is called a **vector space over** $\mathbb{Z}_2$. 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

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

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 $-1 = 1$,

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

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 which would be calculated if A were treated as a matrix of real numbers; you may confirm that in that case $\operatorname{rank} A = 3$.

Here are all of the members of $\operatorname{Nul} A$:

This should be 1

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

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.

Assume that the messages are each 4 digits long. A self-correcting code for these messages will now be created. A more sophisticated version of the parity check is done; three numbers will be added 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 will be called $\mathbf{h}_1, \mathbf{h}_2, \ldots \mathbf{h}_7$, happen to be all of the non-zero members of $\mathbb{Z}_2^3$. A basis for the null space of H may be found as above:

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

For reasons which will be made clear later, it will be better to have a different basis for Nul H. This new basis will be created by making a matrix whose rows are the vectors in the old basis, row reducing this matrix, then using the non-zero rows of the resulting matrix as a basis for $\operatorname{Nul} H$. This process is allowable by Theorem 13 in Section 4.6.

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

Thus the following set of vectors is also a basis for $\operatorname{Nul} H$.

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

Since the dimension of $\operatorname{Nul} H$ is 4, by the earlier fact $\operatorname{Nul} H$ contains 16 vectors. Of course, $\mathbb{Z}_2^4$ also contains 16 vectors, so each vector in $\mathbb{Z}_2^4$ can be encoded using a different vector in $\operatorname{Nul} H$. For that reason the null space of H is called the **Hamming (7,4) code**. To encode the vectors in $\mathbb{Z}_2^4$, form a matrix A whose columns are the basis elements for $\operatorname{Nul} H$; the matrix A will be the 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, 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 for the detection and correction of single errors in transmitted messages. 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. 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, there is 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, $H\mathbf{e}_i = \mathbf{h}_i \neq \mathbf{0}$. Thus

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

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 that error may be detected by checking to see whether the received message lies in $\operatorname{Nul} H$.

Example: If the message 0100101 is received, 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 the message vector is in $\operatorname{Nul} H$, no single transmission error has happened. If a single error had happened, the theorem says that the resulting message vector would not be in $\operatorname{Nul} H$.

Example: If the message 0111001 is received, 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) a single transmission error has happened.

So the Hamming (7,4) code is an error-detecting code. The following theorem will show 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 $\operatorname{Nul} H$, and $\mathbf{x} + \mathbf{e_j}$ is not in $\operatorname{Nul} H$ 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 a message $\mathbf{x}$ is received 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 produce a vector in $\text{Nul}\, H$, and thus a properly encoded vector. Changing any other digit in $\mathbf{x}$ will not produce a vector in $\text{Nul}\, H$.

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

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

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