**Due: Fr. Apr. 24** **NAME(S):** ______________________

______________________

______________________

______________________

# Math 215 Project 3 (25 pts): Error-Detecting and Correcting Codes - Hamming (7,4) code

## 1 Error-Detecting and Correcting Codes

In this project, 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 the concepts learned on vector spaces, null spaces, rank, and dimensions are needed for this construction. When a message is transmitted, it has a potential to get scrambled by noise. This is true for all digital messages (e.g. email, texts, sound, video) that are sent to and from computers and mobile devices. This is also true of store scanners (bar code). By a digital message, we mean a sequence of 0's and 1's which encode a given message. Digital errors are often in the form of a 0 being received as a 1 or vice versa. What we will seek to do is to add more data to a given binary message that will help 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 the transmission, and to figure out what the original message was from the possible corrupt message that we received. This type of code is an **error-correcting code.**

## 2 Vector Space of 0's and 1's

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 $\{0, 1\}$. Addition and multiplication on $\mathbb{Z}_2$ are define 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. Also, notice that since $1 + 1 = 0$ then $1 = -1$ in this setting. That is, 1 is its own additive inverse, and thus subtraction is exactly the same as addition in $\mathbb{Z}_2$.

We now express messages as column vectors of elements of $\mathbb{Z}_2$. The message 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 $\mathbb{Z}_2$. We will focus on $n = 4$, i.e. $\mathbb{Z}_2^4$. Since there are two choices $\{0, 1\}$ for each position in the vectors of length 4, there are $2^4 = 16$ different vectors. The set $\mathbb{Z}_2^4$ contains the following 16 different 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 multiple these vectors by scalars taken from $\mathbb{Z}_2$.

**Example 1:**

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

In fact, if we use $\mathbb{Z}_2$ as scalars, and use the operations of vector addition and scalar multiplication as given in Example 1, then $\mathbb{Z}_2^n$ is a vector space. We say $\mathbb{Z}_2^4$ is a vector space over $\mathbb{Z}_2$ to emphasize that the scalars we use are taken from $\mathbb{Z}_2$.

**Example 2:**
Find a basis for the column space, a basis for the null space, and the rank of the matrix,

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

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

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

A basis for the column space of $A$ is the pivot columns in $A$:

$$\text{Col(A)} = \left\{\begin{bmatrix}1\\1\\0\end{bmatrix},\begin{bmatrix}1\\0\\1\end{bmatrix}\right\}$$

Thus the rank of $A$ is 2. To find a basis for the null space of $A$, solve $Ax = 0$ and after row reductions get:

$$\left[\begin{array}{cccc|c}1&0&1&1&0\\0&1&1&1&0\\0&0&0&0&0\end{array}\right]$$

This yields the equations,

$$\begin{aligned} x_1 &= -1x_3 - 1x_4 \\ x_2 &= -1x_3 - 1x_4 \\ x_3 &= 1x_3 \\ x_4 &= 1x_4 \end{aligned}$$

where $x_3$ and $x_4$ are free variables. Since $-1 = 1$, we rewrite the equations as,

$$\begin{aligned} x_1 &= 1x_3 + 1x_4 \\ x_2 &= 1x_3 + 1x_4 \\ x_3 &= 1x_3 \\ x_4 &= 1x_4 \end{aligned}$$

A basis for the null space of $A$ would be

$$\text{Nul(A)} = \left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} \right\}$$

We can list all of members of null space of $A$

$$\text{Nul(A)} = \left\{ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \right\}$$

and note that the number of vectors in Nul(A) is $4 = 2^2$ which is 2 raised to the dimension of the 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 dimW = k, then the number of vectors in $W$ is equal to $2^k$.

# 3 Hamming (7,4) code

Let us assume that our messages are 4 digits long. We will now describe the Hamming (7,4) code for detecting and correcting errors. Let the 7 columns $h_1, h_2, \ldots, h_7$ of the matrix $H$ represent all of the non-zero vectors in $\mathbb{Z}_2^3$,

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

As above, we can find a basis for the null space of $H$:

$$\text{Nul(H)} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \right\}$$

Since the dimension of Nul(H) is 4, by our earlier fact Nul(H) contains 16 vectors. Notice that $\mathbb{Z}_2^4$ also contains 16 vectors, so we can encode each vector in $\mathbb{Z}_2^4$ using a different vector in 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 use a matrix $A$ whose columns are the basis elements for 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 3:**
To encode the message 1101, we compute

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

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 allows 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 $e_1, e_2, \ldots, e_7$ in $\mathbb{Z}_2^7$:

$$e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, e_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, e_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \ldots, e_7 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Notice that adding one of these vectors to an encoded message vector $x$ is equivalent to making a single error in the transmission of $x$. Notice also that the vectors $e_1, e_2, \ldots, e_7$ are not in the nullspace of $H$, since $He_i = h_i \neq 0$. In fact, we have the following Theorem.

**Theorem 1:** If $H$ is the matrix given above and if $x$ is in Nul(H), then $x + e_i$ is not in Nul(H).

**Example 4:**
If we received the message 0100101, we can check that

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

Since our message vector is in Nul(H) we know that no single transmission error has happened. If a single error had happened, then the theorem tells us that the resulting message vector would not be in Nul(H).

**Example 5a:**
If we received the message 0111001, we can check that

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

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 $Hx = h_i$, then $x + e_i$ is in Nul(H).

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

**Example 5b:**
The message 0111001 was in error in Example 5a. In fact, we found that

$$H \cdot \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = h_2$$

By Theorem 2, the single error in transmission must have occurred at the second digit. Thus, the true message which was sent was 0**0**11001.

# 4 Exercises

**Exercise 1 (4pts) :**
Prove Theorems 1 and 2, i.e. show that theorems are true.

**Exercise 2 (3pts) :**
Encode the following messages
a) 1001
b) 0011
c) 0101

**Exercise 3 (6pts) :**
Use Octave/Matlab for this problem. We assume only one error. Hint: modular arithmetic command in Octave/Matlab can be done with the command mod, i.e. `mod(x,2)` where `x` can be a number, vector, or even a product of a matrix with a vector (i.e. you can use `mod(H*x,2)`). 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. Determine whether an error was made in the transmission, and if so correct it.
a) 0101110
b) 1000011
c) 0010111
d) 0101010
e) 0111100

**Exercise 4 (2pts):**
The Hamming (7,4) code is used for a channel prone to erasures, but not to errors. If ??11001 is received what was the transmitted message?

**Exercise 5 (10pts):**
Given the following table:

| Message | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| 4-bit string | 0000 | 0001 | 1110 | 1011 | 0100 | 0101 | 0111 | 0110 |
| Message | 8 | 9 | + | - | x | ÷ | space | |
| 4-bit string | 1000 | 1111 | 1010 | 0011 | 1001 | 1101 | 1100 | |

a) Encode the questions: $23 + 19$(space)$279 \div 6$ using the table above. Append error detect string to each message by using the techniques given in this project (similar to exercise 2).
b) Decode the string given below. You will need to correct errors using techniques in this project, similar to exercise 3). Separate the strings into 7 digit messages.

000111000000100011100100011111000010001110
0100011110010010110101110001100100111010101111111