0.1. Linear Codes

In elementary physics, we learn of the principal of superposition while studying oscillation theory; we learn how energy passing through a medium in the form of a wave can *interfere* with other such waves to create a new wave, the sum of the waves. This persists until these bursts of energy have passed.

The modern theories of commumincation and information seak to pass information via electrical signals (waves) passing through some channel (media). As information in the form of signals passes through the channel, it will invariably come into contact with other signals, whether latent or otherwise. The signal will then change, and there is no guaruntee that the signal received will be the signal sent; whence, one of the fundamental problems of digital communication is exposed. It becomes necessary, then, to attempt a solution. **?** provides the following instructive example.

Imagine there are two possible messages that we would like to send: `YES` and `NO`. And suppose further that `YES` is encoded as `0` and `NO` as `1`. If we had sent the message `0`, there is the possibility of the single bit being flipped, i.e. `1` is received instead of `0`, the initial message. The receiver would then have an incorrect message. It isn't difficult to envisage senarios in which such an error would have a terrible impact.

Instead, we will encode `YES` as `00000` and `NO` as `11111`. If we sent `00000` through the channel, and if the receiver got `01000`, then it would be reasonable to assumenearest-neighbour that the message intended was `YES`, a correct assumption. Since a single bit was all that was necessary to convey our minimal lexicon {`YES, NO`}, we see that the remaining 4 bits are *redundant*; however, we also see that these redundancies allowed us to correct the message in the case that any single bit was flipped. Similarly, if any two bits had been flipped, then we also would have correctly interpreted the message. If, however, we sent `00000` and received `00111`, then we would have interpreted this message as `11111`, clearly incorrect. So we see that there is a limitation of our ability to correct errors in transmition, but this is hardely unexpected. In any event, the ability to detect and correct errors is a remarkable property whose importance cannot be overstated in today's digital world.

The process of appending to `0` and `1` the redundant strings of bits `0000` and `1111`, respectively, is called *encoding*, and we now introduce such a method.

Suppose we had the binary string $x = x_1x_2\cdots x_n$, where the first k bits $x_1x_2\cdots x_k$ are the message we desire to preserve upon sending through some channel. We choose the remaining *check bits* $x_{k+1}x_{k+2}\cdots x_n$ (the redundancies) in such a way that

$$Hx^t = 0 \qquad (\text{mod } 2),$$

where H is the binary $(n-k)\times n$ matrix called the *parity check matrix* of the code. Moreover, by our assumptions on x, H can be assumed to have the form

$$H = (A \mid I_{n-k}),$$

for some binary matrix A.

More generally, we may take x and H to be over any field GF(q). We have the following definition.

0.1 Definition. Let $H = (A \mid I_{n-k})$ be an $(n-k) \times n$ matrix over GF(q). The *linear code* $\mathscr{C}$ with parity check matrix H is given by $\mathscr{C} = \text{Null}(H) = \{x \in \text{GF}(q^n) : Hx^t = 0\}$, where the extension GF$(q^n)$ is interpreted here as a linear space over GF(q). We say that $\mathscr{C}$ is a linear $[n,k]_q$-code, where clearly $\dim(\mathscr{C}) = k$.

0.2 Example. The repitition code over GF$(q) = \{0, a_1, \ldots, a_{q-1}\}$ of length n is given by

$$\overbrace{\begin{matrix} 0 & 0 & \cdots & 0 \\ a_1 & a_1 & \cdots & a_1 \\ \vdots & \vdots & \vdots & \\ a_{q-1} & a_{q-1} & \cdots & a_{q-1} \end{matrix}}^{n}$$

and has parity check matrix $H = (\mathbf{1}_{n-1} \mid I_{n-1})$.

We have defined a linear code by its parity check matrix. An equivalent definition is to use the so-called *generator matrix* G. If $H = (A \mid I_{n-k})$ is the parity check matrix of the code, then $G = (I_k \mid -A^t)$. The code is then given by the linear span of the rows of G. Note that by construction, it follows that $GH^t = HG^t = O$.

0.3 Example. The repitition code of the previous example has the generator matrix $\mathbf{1}_n^t$.

The dual of a linear code $\mathscr{C}$ is given in the usual way by $\mathscr{C}^\perp = \{x \in \text{GF}(q^n) : xy^t = 0, \text{ for every } y \in \mathscr{C}\}$. If H and G are the parity check and generator matrices of $\mathscr{C}$, then G and H are the parity check and generator matrices of $\mathscr{C}^\perp$, respectively.

In Definition 0.1, there are two parameters explicitly given of a code, namely, the length n of the codewords and the dimension k of the linear space consisting of the codewords. Two more fundamental parameters for us are the minimum distance and the minimum weight defined thus.

0.4 Definition. Let $\mathscr{C}$ be a linear $[n,k]_q$-code. The *Hamming weight* of a codeword $x = x_1x_2\cdots x_n \in \mathscr{C}$ is given by $\text{wt}(x) = \#\{i : x_i \neq 0\}$. If $y = y_1y_2\cdots y_n \in \mathscr{C}$ is any other codeword, then the *Hamming distance* between x and y is defined as $\text{dist}(x,y) = \#\{i : x_i \neq y_i\}$. We then have:

(0.4.a) The minimum weight of the code is $\text{wt}(\mathscr{C}) = \min_{x \in \mathscr{C}\setminus 0} \text{wt}(x)$, and

(0.4.b) the minimum distance of the code is $\text{dist}(\mathscr{C}) = \min_{\substack{x,y \in \mathscr{C} \\ x \neq y}} \text{dist}(x,y)$.

If we wish to emphasize the distance $d = \text{dist}(\mathscr{C})$ of a code, we write $[n, k, d]_q$-code; and to emphasize the weight $w = \text{wt}(\mathscr{C})$ of a code as well, $[n, k, d, w]_q$-code.

The Hamming distance can easily be seen to form a metricmetric on GF(q^n). The next result is then clear (see **?**, Theorem 1.9).

0.5 Proposition. If $\mathscr{C}$ is any $[n, k, d]_q$-code, then the following hold.

(0.5.a) The code can detect s errors if $\text{dist}(\mathscr{C}) \geq s + 1$;

(0.5.b) the code can correct up to t errors if $\text{dist}(\mathscr{C}) \geq 2t + 1$; and

(0.5.c) for every $x, y \in \mathscr{C}$, it holds that $\text{dist}(x, y) = \text{wt}(x-y)$, hence $\text{dist}(\mathscr{C}) = \text{wt}(\mathscr{C})$.

We note that in the case the code is non-linear (see §2.3.), the condition (0.5.c) fails to hold in general.

We give one final result in this subsection relating the parity check matrix and the minimum distance.

0.6 Proposition. Let $\mathbf{C}$ be a linear $[n, k]_q$-code with parity check matrix H. Then $\mathscr{C}$ has $\text{dist}(\mathscr{C}) \geq d$ iff every $d - 1$ columns of H are linearly independent.

Proof. To show necessity, let $x \in \mathscr{C}$ and $w = \text{wt}(x)$. Since $Hx^t = 0$, H has at least w linearly dependent columns; hence, if if any $d - 1$ columns of H are linearly independent, then, by an application of (0.5.c), $\mathscr{C}$ cannot have a codeword of weight $d - 1$ or less.

Towards sufficiency, assume that H has $d-1$ linearly independent columns. If $c_1, \ldots, c_n$ are the columns of H, then there is a dependence relation $a_1c_1 + \cdots + a_nc_n = 0$ ($a_1, \ldots, a_n \in \text{GF}(q)$) for which at most $d - 1$ of the a_is are nonzero. Then the codeword $x = a_1a_2 \cdots a_n$ has weight less than d, thereupon $\text{dist}(\mathscr{C}) \leq d$. ■

0.2. The Hamming and Simplex Codes

The theory of error-correcting codes began with the seminal paper of Richard W. Hamming (**?**) who introduced redundancy for the purpose of error correction. In this paper, he also constructed a most useful family of codes, aptly called the *Hamming codes*. The codes constructed by Hamming were binary; however, it is a simple matter to extend the construction to an arbitrary finite field.

0.7 Definition. Let H be the $n \times (q^n - 1)/(q-1)$ matrix over GF(q) whose columns are representatives of the nonzero 1-dimensional subspaces of the extension field GF(q^n) over GF(q). Then:

(0.7.a) The linear code $\mathscr{H}_{q,n}$ whose parity check matrix is H is called a *Hamming code*.

(0.7.b) The linear code $\mathscr{S}_{q,n}$ whose generator matrix is H is called a *simplex code*.

0.8 Example. The following is the simplex code corresponding to $q = n = 3$.

$$\mathscr{S}_{3,3} = \begin{pmatrix}
\overline{1}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{0}\,\overline{1}\,\overline{2} \\
\overline{2}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{0}\,\overline{2}\,\overline{1} \\
\overline{0}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{0}\,\overline{0}\,\overline{0} \\
\overline{1}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{2}\,\overline{0}\,\overline{1} \\
\overline{2}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{2}\,\overline{1}\,\overline{0} \\
\overline{0}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{2}\,\overline{2}\,\overline{2} \\
\overline{1}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{0} \\
\overline{2}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{2} \\
\overline{0}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{1} \\
\overline{1}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{1}\,\overline{2}\,\overline{0} \\
\overline{2}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{1}\,\overline{0}\,\overline{2} \\
\overline{0}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{1}\,\overline{1}\,\overline{1} \\
\overline{1}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{0}\,\overline{1}\,\overline{2} \\
\overline{2}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{0}\,\overline{2}\,\overline{1} \\
\overline{0}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{0}\,\overline{0}\,\overline{0} \\
\overline{1}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{1} \\
\overline{2}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{0} \\
\overline{0}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{2} \\
\overline{1}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{2}\,\overline{0}\,\overline{1} \\
\overline{2}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{2}\,\overline{1}\,\overline{0} \\
\overline{0}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{1}\,\overline{1}\,\overline{1}\,\overline{2}\,\overline{2}\,\overline{2} \\
\overline{1}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{1}\,\overline{2}\,\overline{0} \\
\overline{2}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{1}\,\overline{0}\,\overline{2} \\
\overline{0}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{2}\,\overline{2}\,\overline{2}\,\overline{1}\,\overline{1}\,\overline{1} \\
\overline{1}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{2}\,\overline{0}\,\overline{1}\,\overline{2} \\
\overline{2}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{1}\,\overline{0}\,\overline{2}\,\overline{1} \\
\overline{0}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{0}\,\overline{0}
\end{pmatrix}.$$

We do not display

0.3. Nonlinear Codes

0.4. Bounds on Codes

Bibliography