Fundementals of Coding Theory

Kyle Dituro

Updated December 22, 2023

Abstract

In this paper, we aim to provide a cursory introduction to the theory of linear codes. We will begin with a brief exposition of some bounds and restrictions on codes to contextualize the field, and will follow with an exploration of linear codes, including encoding, decoding, the dual code, and the Hamming code.

1 Motivating Problem and Introduction

Suppose you wanted to send a message over a wire (which we will henceforth call a *channel* to be defined precisely later) with some degree of unreliability.

> **The Fundamental Problem of ECC** – Find a way to communicate over our channel with reasonable information content and reasonable error handling ability.

Obviously, for an arbitrarily bad channel, this is impossible. if we assume that whatever message we send will be distorted completely randomly or completely erased, then there is nothing that we can do. Thus we have a

> **Reasonable Assumption** – Most errors that occur are not severe.

If errors are severe, then the issue is not one of coding, but rather an issue of engineering.

We can break this problem down into a number of steps:

1. The sender encodes the message
2. The channel transports the message (possibly unreliable)
3. The recipient decodes the message

It is fairly easy to come up with some naive ideas for encoding schemes (which we will call "codes")

> **Example 1.1** (The n-repetition code)**.** By simply repeating your message $n \in \mathbb{Z}$ times, you can guarantee a recovery from $\left\lfloor \frac{n}{2} \right\rfloor$ errors by decoding via consensus.
>
> **Example 1.2** (The parity check code)**.** Take the sum of all digits in a code-word of length $n-1$ and set the n^{th} digit to verify the parity of the other digits.

Example 1.3 (The Venn code)**.** One particular code which we will return to regularly is the $[7, 4]$ binary Hamming code (also sometimes called the Venn code). We quote from Shannon's paper[1]:

> "An efficient code, allowing complete correction of [single] errors and transmitting at the rate $C[= 4/7]$, is the following (found by a method due to R. Hamming):
>
> Let a block of seven symbols be $X_1, X_2, \ldots, X_7$. Of these X_3, X_5, X_6, and X_7 are message symbols and chosen arbitrarily by the source. The other three are redundant and calculated as follows:
>
> $$X_4 \text{ is chosen to make } \alpha = X_4 + X_5 + X_6 + X_7 \text{ even}$$
> $$X_2 \text{ is chosen to make } \beta = X_2 + X_3 + X_6 + X_7 \text{ even}$$
> $$X_1 \text{ is chosen to make } \gamma = X_1 + X_3 + X_5 + X_7 \text{ even}$$
>
> When a block of seven is received, α, β, and γ are computed and if even called zero, if odd called one. The binary number $\alpha\beta\gamma$ then gives the subscript of the X_i that is incorrect (if 0 then there is no decoding error)."

So for example, to send the message 1011, we would have $X_4 = 0$, $X_2 = 1$, $X_1 = 0$, making our codeword 0110011, and if we received the codeword.

Now separately suppose that the codeword 1000111 is received. Then we'd compute that $\alpha = 1$, $\beta = 0$, and $\gamma = 1$, giving us that position $(101)_2 = 5$ is incorrect. So the code word that was *actually* sent was 1000011.

2 Important Definitions

Definition 2.1. A *code* C over an alphabet A is a nonempty subset $C \subset A^n$ of elements called "*codewords*"

Definition 2.2. For sets $I \subseteq A^n \subseteq O$ (calling I the input set and O the output set), a *channel* $H : I \to O$ is a map of sets represented by a matrix which is column stochastic, where $H_{x,y} = \Pr(y \mid x)$. The matrix H is oft called the "transition matrix".

We often require a channel to be "discrete" and "memoryless", meaning that A is finite, and the channels transition probabilities are independent respectively.

Example 2.3. The m-ary symmetric channel $m\mathrm{SC}(p)$ is the channel supporting an alphabet of size m where $\Pr(y \mid x) = p$, (and thus $\Pr(x \mid x) = 1 - (m-1)p$).

Example 2.4. The m-ary erasure channel $m\mathrm{EC}(p)$ is the DMC over a size m alphabet with output alphabet $A \cup \{?\}$, with $\Pr(x \mid x) = 1 - p$, and $\Pr(? \mid x) = p$, where "?" is a character representing an erasure.

Since they are matrices, channels can be thought of as weighted bipartite graphs between the input and output languages weighted by the transition probabilities.

3 Hamming Distance and Sphere Packing

One particularly useful notion is how 'different' two different words are. In this case, the simplest idea is the best.

> **Definition 3.1.** The *Hamming Distance* between $\mathbf{x}$ and $\mathbf{y}$ is defined to be the number of places in which $\mathbf{x}$ and $\mathbf{y}$ differ.

It's worth noting that the Hamming distance is a metric.

As such, denote $S_H(\mathbf{x}, \rho)$ as the sphere centered at $\mathbf{x}$ with radius ρ

This leads to a pretty natural decoding algorithm: Fix a radius ρ, and consider the spheres or radius ρ centered at the codewords $\mathbf{c} \in C$. Then, when receiving a message, check if it lies in a unique sphere. If it does, decode to the center of the sphere. If it does not, declare a decision default.

This is called the **Radius ρ Sphere Shrinking Algorithm**, denoted $\mathbf{SS}_\rho$.

> **Lemma 3.2.** *The following are equiv. for the code C in A^n for some integer $e \leq n$.*
>
> 1. *For the code C, the algorithm $\mathbf{SS}_e$ recovers from all occurrences of e or fewer errors*
> 2. *For all distinct $\mathbf{x}, \mathbf{y}$ in C, we have that $S_H(\mathbf{x}, e) \cap S_H(\mathbf{y}, e) = \emptyset$*
> 3. *the minimum distance of C, denoted $\mathrm{d}_{\min}(C)$ is at least $2e + 1$*

And this lemma leads to the following theorem

> **Theorem 3.3** (Sphere Packing Bound)**.** *Assume that $|S_H(\mathbf{x}, e)|$ is independent of $\mathbf{x} \in A^n$. Now let C be a code within A^n. If $\mathbf{SS}_e$ recovers from all of e errors of fewer, then*
>
> $$|C| \cdot |S_H(*, e)| \leq |A|^n$$

Proof. If we have distinct $c, d \in C$, have $y \in S_H(c, e) \cap S_H(d, e)$, then $\mathbf{SS}_e$ does not recover from either event $c \sim y$ or $c \sim d$. Thus the spheres $S_H(*, e)$ must be pairwise disjoint in A^n, so the inequality follows. □

> **Lemma 3.4.** *In A^n with $|A| = m$, a Hamming sphere of integral radius e has size*
>
> $$|S_H(*, e)| = \sum_{i=0}^{e} \binom{n}{i} (m-1)^i$$
>
> *and thus contains as many words in the wordspace.*

Proof. This is a simple counting of the number of ways to change up to e characters in a word to any of the $m-1$ other possible (incorrect) letters. □

Corollary 3.5 (The Hamming Bound)**.** *Let C be an e-error correcting code in A^n with $|A| = m$. Then*

$$|C| \leq \frac{m^n}{\sum_{i=0}^{e} \binom{n}{i} (m-1)^i}.$$

4 Shannon's Theorem

We now present one of the most important an surprising results of coding theory, due to Shannon. The crux of the theorem – called the Channel Coding Theorem – is that in most circumstances, nearly error-free communication can be achieved. Of course, no guarantees are made about how good the code is.

We introduce a way of interpreting a code's "optimal performance":

Let $\mathrm{P}(C)$ be the minimum average decision failure rate for the code for any decoding algorithm. In other words, for every possible decoding algorithm for the code, compute the average chance of a decision failure when using the algorithm, and take the algorithm with the minimum such average.

Definition 4.1. A *Shannon Family* $\mathcal{F}$ is a collection of codes with the property that for every $\varepsilon > 0$, There is a code $C \in \mathcal{F}$ with $P(C) < \varepsilon$

Example 4.2. The set of all binary repetition codes is a Shannon family for the binary symmetric channel $\mathrm{BSC}(p)$ where $0 \leq p < 1/2$. To see this, realize that we can just take more repetitions of the codeword until the code is good enough.

For the discrete memoryless channel $\mathrm{DCM}(M)$ and $0 \leq \kappa \leq 1$, denote $\mathrm{DMC}(M)_\kappa$ as being the family of all codes $C \subseteq A^n$ (for $|A| = m$) for the channel having *rate*

$$\kappa(C) = \frac{\log_m |C|}{n} \geq \kappa.$$

Intuitively, we think of the rate of a code as a rough estimate of the number of symbols which contain information as opposed to redundancy.

Theorem 4.3 (Shannon's Channel Coding Theorem, 1948)**.** *For every* $\mathrm{DMC}(M)$*, there is a constant* $\mathrm{Cap}(M)$ *(called the capacity of the DMC) such that*
$\mathrm{DMC}(M)_\kappa$ *is a Shannon family when* $\kappa < \mathrm{Cap}(M)$
but
DMC$(M)_\kappa$ *is not a Shannon family when* $\kappa > \mathrm{Cap}(M)$.

We note that the capacity for most channels is rather difficult to compute, but some example computations can be found in [2].

In the language of sphere packing, consider the sphere packing bound over an alphabet of size m:

$$|C| \cdot |S_H(*, e)| \leq |A|^n$$

Then taking a logarithm and dividing by n we get that

$$\frac{\log_m |C|}{n} + \frac{\log_m |S_H(*, e)|}{n} \leq 1$$
$$\kappa(C) + \rho(C) \leq 1$$

(where $\rho(C)$ is the relative size of the error spheres in C). Essentially, this means that we cannot succeed if we are too greedy with either our information content κ or our error handling capability ρ.

It's worth noting that Shannon's Theorem is quite powerful. For example, per [2]:

$$\text{Cap}(\text{BSC}(2\%)) \sim .8586,$$

meaning that even with a nearly 2 percent chance of error on the binary symmetric channel, we can still communicate error-free with 85% information content.

5 Linear Codes

Up to this point, the codes which we have discussed have been almost completely bereft of structure. Because of this, or reasoning about efficient encoding and decoding algorithms is quite limited. In order to remedy this, we will begin discussing linear codes.

> **Definition 5.1.** A *linear code* of length n over a field F is a vector subspace of F^n.

In this context, we are encouraged to think about our codewords now as vectors, and moreover as being generated as the row space of some matrix G with linearly independent rows. We call such a matrix a *generator matrix*, and if G is $k \times n$, then we say that the code C is a $[n, k]$-linear code over F.

In other words, we think about a linear code C as being a k-dimensional vector subspace of F^n. Practically, we can think of this as having k 'useful' bits of information and $n - k$ redundant bits of information in the codevector. Thus the rate of information transmission is $R = \frac{k}{n}$

> **Example 5.2.** In Example 1.3 we looked at the Venn code defined by the equations
>
> $$X_4 \text{ is chosen to make } \alpha = X_4 + X_5 + X_6 + X_7 \text{ even}$$
> $$X_2 \text{ is chosen to make } \beta = X_2 + X_3 + X_6 + X_7 \text{ even}$$
> $$X_1 \text{ is chosen to make } \gamma = X_1 + X_3 + X_5 + X_7 \text{ even}$$
>
> Where (X_3, X_5, X_6, X_7) were message bits and the others were redundant bits.

Thus we expect our code to be of dimension 4 as a subspace of $\mathbb{F}_2^7$. Then by translating the messages $(1,0,0,0)$, $(0,1,0,0)$, $(0,0,1,0)$, and $(0,0,0,1)$ into their corresponding codewords, we get the generator matrix

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

Notice that this also retroactively justifies our naming of the Venn code as the $[7,4]$ binary Hamming code.

5.1 Encoding

This also gives us a sensible way of encoding an arbitrary k-tuple $\mathbf{m}$ with elements from F as a codeword. If C is a $[n,k]$ linear code with generator matrix G, then we can realize $\mathbf{m}G$ is a linear combination of the rows of G, and is thus a codeword.

From this construction, we see that the information from $\mathbf{m}$ will be maintained in the bits of our codeword corresponding to the columns of the generator matrix which correspond to the identity. We call these coordinates an *information set* for the code C, and we say that the generator matrix is *systematic* on those columns. Put explicitly:

Definition 5.3. A $k \times n$ matrix

$$G = \left[\begin{array}{c|c|c} & & \\ \mathbf{g}_1 & \cdots & \mathbf{g}_n \\ & & \end{array}\right]$$

is called *systematic* on positions $i_1, \ldots, i_k$ if

$$\left[\begin{array}{c|c|c} & & \\ \mathbf{g}_{i_1} & \cdots & \mathbf{g}_{i_k} \\ & & \end{array}\right] = I_{k\times k}$$

The following fact then follows naturally:

Fact. *If C is an $[n,k]$ linear code over F with codevectors $\mathbf{c} = \begin{bmatrix} c_1 & \ldots & c_n \end{bmatrix}$, then $\{c_{i_1}, \ldots, c_{i_k}\}$ is an information set for C if there exists a generator matrix G that is systematic on the columns $i_1, \ldots, i_k$.*

5.2 The Dual Code

Following from our characterization of linear codes as the row space of some generator matrix, we can naturally define the idea of the dual code.

Definition 5.4. Let C be a code in F^n. The *dual code* of C, denoted $C^\perp$ is defined

to be the code.

$$C^\perp = \{\mathbf{x} \in F^n \mid \mathbf{x} \cdot \mathbf{c} = 0,\ \forall \mathbf{c} \in C\}$$

It is important to note that in our definition, C is not necessarily a linear code; and yet $C^\perp$ is still a linear code despite this.

Lemma 5.5. *For any code C in F^n, $C^\perp$ is a linear code.*

Proof. Follows from the fact that $C^\perp$ is clearly a vector subspace. □

This leads us to naturally ask about the dual of the dual $C^{\perp\perp}$. Obviously, since the dual of a code is linear, if C is not linear, then $C^{\perp\perp} \neq C$. So the question is begged: what if C is linear?

Lemma 5.6. *If C is an $[n, k]$ linear code over F, then $C^\perp$ is an $[n, n-k]$ linear code over F, and moreover $C = C^{\perp\perp}$*

Proof. Notice that if C is linear with generator G, we get that $C^\perp = \text{null}(G)$, and so by the rank-nullity theorem, $\dim \text{null}(G) = n - k$. So applying rank-nullity again, this time on null $\text{null}(G)$, we get back again that $\dim C^{\perp\perp} = k$.

Now, notice that $C \subseteq C^{\perp\perp}$, since if $\mathbf{x} \in C^\perp$, then (by definition) $\mathbf{c} \cdot \mathbf{x} = 0$ for all $\mathbf{c} \in C$.

Therefore, since $C \subseteq C^{\perp\perp}$ and $\dim C = \dim C^{\perp\perp}$, $C = C^{\perp\perp}$. □

Now suppose that the generator matrix for the dual code $C^\perp$ is H. Then we call H a *check matrix* for C.

This nomenclature comes from the fact that we can think of a vector in the null space as specifying the coefficients for a parity check equation on the vectors in our code. In other words, a vector $\mathbf{x} \in C^\perp$ checks a subset of the coordinates of a codevector $\mathbf{c} \in C$. So we can specify a code only using the check matrix by realizing that

$$C = \left\{\mathbf{x} \;\middle|\; H\mathbf{x}^T = \mathbf{0}\right\}.$$

Example 5.7. Returning again to the $[7, 4]$ Hamming code, we can compute the check matrix of our code to be

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

Notice that this corresponds directly to the equations given in our original treatment of the $[7, 4]$ Hamming code that we gave in Example 1.3. This demonstrates the above-mentioned idea that we can define a code completely by the check matrix.

5.3 Decoding

Now that we have developed the idea of the dual code, we can discuss decoding algorithms for our linear codes.

We begin by thinking of any errors applied by our wire as being added to our message vector as noise. In other words, if the codeword $\mathbf{c}$ is sent, and the word $\mathbf{w}$ is received, then we define the error to be

$$\mathbf{e} := \mathbf{w} - \mathbf{c}.$$

Now consider the partitioning of F^n as cosets of our code C. Then from the above, we realize that $\mathbf{c} + \mathbf{e} = \mathbf{w}$, so $\mathbf{c} + \mathbf{e} + C = \mathbf{w} + C$, and thus $\mathbf{e} + C = \mathbf{w} + C$. In other words, our error and the received word lie in the same coset of our space.

Now, in each coset, choose the vector with the smallest Hamming weight, and call it the *coset leader* of the coset. We will denote the coset leader by $\widehat{\mathbf{e}}$. Then, by our reasonable assumption at the beginning, we can guess that the error that occurred in transmission is $\widehat{\mathbf{e}}$, so we can decode the message $\widehat{\mathbf{c}} = \mathbf{w} - \widehat{\mathbf{e}}$.

One way to accomplish this practically is by constructing a dictionary matrix of dimension $K = |C|$, $R = |F^n|/|C|$ and constructing the dictionary matrix

$$\begin{array}{c} \\ \widehat{\mathbf{e}_1} \\ \vdots \\ \widehat{\mathbf{e}_R} \end{array} \begin{array}{c} \begin{array}{cccc} \mathbf{c}_1 & \cdots & \cdots & \mathbf{c}_K \end{array} \\ \begin{bmatrix} \mathbf{c}_1 + \widehat{\mathbf{e}_1} & \cdots & \cdots & \mathbf{c}_K + \widehat{\mathbf{e}_1} \\ \vdots & & & \vdots \\ \mathbf{c}_1 + \widehat{\mathbf{e}_R} & \cdots & \cdots & \mathbf{c}_K + \widehat{\mathbf{e}_R} \end{bmatrix} \end{array}$$

of all potential received codewords from expected errors given by the coset leaders. Then all possible elements of F^n are enumerated, so we can decode by lookup.

This is – obviously – horribly space-inefficient.

Instead, we look to an idea called **Syndrome decoding**. We mentioned before that the check matrix H for a code C can be thought of as a parity check equation for our matrix, and in fact we can realize that

$$H\mathbf{x}^T = H(\mathbf{c} + \mathbf{e})^T = 0 + H\mathbf{e}^T = H\mathbf{e}^T,$$

and so multiplication of a vector by the check matrix is consistent on the cosets of our space. So call the possible results when multiplying a vector $\mathbf{x}^T$ by H the *syndromes* $\mathbf{s}_i$ of our code, and to each syndrome associate the coset leader $\mathbf{e}_i$.

Then, when we receive a vector $\mathbf{x}$, compute $H\mathbf{x}^T = \mathbf{s}$. Then look up the error $\widehat{\mathbf{e}}$ associated to the syndrome $\mathbf{s}$, and decode to the codeword $\widehat{\mathbf{c}} = \mathbf{x} - \widehat{\mathbf{e}}$.

Example 5.8. Recall the check matrix for the $[7, 4]$ Hamming code which we stated before to be

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

Computing the syndromes from the possible errors, we get

Error	Syndrome
$\mathbf{0}$	000
$\mathbf{e}_7^T$	111
$\mathbf{e}_6^T$	011
$\mathbf{e}_5^T$	101
$\mathbf{e}_4^T$	100
$\mathbf{e}_3^T$	011
$\mathbf{e}_2^T$	010
$\mathbf{e}_1^T$	001

Now assume that we received the vector $\mathbf{x} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$. Then we can compute

$$H\mathbf{x}^T = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix},$$

Which tells us that we should look up syndrome 101.

Then the error we decode from should be $\mathbf{e}_5^T$, and so we decode to

$$\widehat{\mathbf{c}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

5.4 The Singleton Bound

Notice that $H\mathbf{x}^T = \sum_i \mathbf{h}_i x_i$, where $\mathbf{h}_i$ are the column vectors of H, and likewise x_i is the i^{th} entry of $\mathbf{x}$. Then we can see quickly that $d_{\min}(C) = d$ if d is the smallest number of linearly dependent columns of H (by taking the distance to the code word $\mathbf{0}$, since the Hamming distance is translation invariant).

This realization proves a major result in the theory of linear codes:

Theorem 5.9 (The Singleton Bound)**.** *If C is an $[n, k]$ linear code over F, then*

$$d_{\min}(C) \leq n - k + 1$$

Proof. Consider the check matrix for C, $H \in F^{n-k \times n}$. Notice that any $(n-k) \times (n-k+1)$ submatrix of H can have rank at most $n-k$, so necessarily there is a linear dependence in the set of $n-k+1$ columns of our submatrix. Therefore, there is a codeword of Hamming weight $n-k+1$, if not fewer. □

Example 5.10. In the $[7, 4]$ Hamming code, taking any submatrix of size 3×4, for example:

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

(the first four columns), we know that we must have a linear dependence. In fact, this submatrix beats the singleton bound, noticing that the first three columns sum to 0. Not every submatrix, however, is guaranteed to give you the minimum distance. In fact, consider the submatrix

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

Notice that this submatrix does not have a subset of 3 columns which are linearly dependent. By checking over every such submatrix, we can find the true minimum distance of our code to be 3.

6 Further Reading and Open Problems

This paper provides – at best – a conversational understanding of the theory of error correcting codes, and gives a flyover view of some of its most important and fundamental results. For a more compete treatment, there is a book currently being authored at the University at Buffalo [3] which is available for free online.

Furthermore, the field of coding theory is still very much active. For those interested in pursuing research in coding theory, we present a number of open problems (per [4]) which are understandable given the treatment in this paper.

First are some questions about size:

Question 6.1. For a fixed n, d, q, find the largest M such that there exists a code $C \subset \mathbb{F}_q^n$ with $|C| = M$ and minimum weight d.

And likewise, we can rephrase this problem for linear codes:

Question 6.2. For a fixed n, d and $\mathbb{F}_q$ find the largest integer $k \leq n$ such that there exists a linear code $C \subseteq \mathbb{F}_q^n$ with $\dim(C) = k$ and minimum weight d.

An important class of codes is those that meet the singleton bound with equality. These codes are called *maximum distance separable*, or MDS for short.

Question 6.3. Find and classify all MDS codes over various classes of alphabets.

Question 6.4. Find an efficient decoding algorithm for a family of self-dual codes (or all self dual codes). (A code is self-dual if $C = C^{\perp}$)

Question 6.5. find the best linear 2-error-correcting code of length n.

References

[1] C. E. Shannon, "A mathematical theory of communication," The Bell system technical journal, vol. 27, no. 3, pp. 379–423, 1948.

[2] J. I. Hall, Notes on Coding Theory. Michigan State University, 2017, these notes are not intended for broad distribution.

[3] V. Guruswami, A. Rudra, and M. Sudan, Essential Coding Theory. SUNY University at Buffalo, 2023. [Online]. Available: https://cse.buffalo.edu/faculty/atri/courses/coding-theory/book/web-coding-book.pdf

[4] S. Dougherty, J.-L. Kim, and P. Solé, "Open problems in coding theory," in Non Commutative Rings and Applications, 07 2013.