Recap: Block Codes

Each message and codeword is of fixed size

\[ \Sigma = \text{codeword alphabet} \]

\[ k = |m| \quad n = |c| \quad q = |\Sigma| \]

\( C = \text{“code”} = \text{set of codewords} \)

\( C \subseteq \Sigma^n \) (codewords)

\[ \Delta(x,y) = \text{number of positions s.t. } x_i \neq y_i \]

\[ d = \min\{\Delta(x,y) : x,y \in C, x \neq y\} \]

Code described as: \((n,k,d)_q\)
Recap: Role of Minimum Distance

Theorem:
A code C with minimum distance “d” can:
1. detect any \((d-1)\) errors
2. recover any \((d-1)\) erasures
3. correct any \(\left\lfloor \frac{d-1}{2} \right\rfloor\) errors

Stated another way:
For \(s\)-bit error detection \(d \geq s + 1\)
For \(s\)-bit error correction \(d \geq 2s + 1\)
To correct \(a\) erasures and \(b\) errors if \(d \geq a + 2b + 1\)
Desired Properties

We look for codes with the following properties:

1. Good rate: \( k/n \) should be high (low overhead)
2. Good distance: \( d \) should be large (good error correction)
3. Small block size \( k \) (helps with latency)
4. Fast encoding and decoding
5. Others: want to handle bursty/random errors, local decodability, ...
Q:
If no structure in the code, how would one perform encoding?

Gigantic lookup table!

If no structure in the code, encoding is highly inefficient.

A common kind of structure added is linearity
Linear Codes

If $\Sigma$ is a finite field, then $\Sigma^n$ is a vector space

**Definition:** $C$ is a linear code if it is a linear subspace of $\Sigma^n$ of dimension $k$.

This means that there is a set of $k$ independent vectors $v_i \in \Sigma^n$ ($1 \leq i \leq k$) that span the subspace. i.e. every codeword can be written as:

$$c = a_1 \, v_1 + a_2 \, v_2 + \ldots + a_k \, v_k$$

where $a_i \in \Sigma$

“Basis (or spanning) Vectors”
Some Properties of Linear Codes

1. Linear combination of two codewords is a codeword.

2. Minimum distance \( (d) = \) weight of least weight (non-zero) codewords
(Weight of a vector refers to the Hamming weight of a vector, which is equal to the number of non-zero symbols in the vector)

\[
\begin{align*}
    d &= \min_{c_i, c_j \in C, \ i \neq j} |c_i - c_j| \\
    &= \min_{c \in C, \ c \neq 0} |c|
\end{align*}
\]
Generator and Parity Check Matrices

3. Every linear code has two matrices associated with it.

1. **Generator Matrix:**
   A $k \times n$ matrix $G$ such that: $C = \{ xG \mid x \in \Sigma^k \}$
   (Note: Here vectors are “row vectors”.)
   Made from stacking the spanning vectors

\[ \begin{array}{c}
\text{msg} \\
G \\
codeword
\end{array} \]
Generator and Parity Check Matrices

2. Parity Check Matrix:
An \((n - k) \times n\) matrix \(H\) such that: \(C = \{y \in \Sigma^n \mid Hy^T = 0\}\)
(Codewords are the null space of \(H\).)

\[
\begin{align*}
\text{received vector} & \quad \text{syndrome} \\
\uparrow & \quad \uparrow \\
\begin{array}{c}
\text{n-k} \\
\downarrow \\
H
\end{array} & \quad \begin{array}{c}
n-k \\
\downarrow \\
\end{array}
\end{align*}
\]

if syndrome = 0, received vector = codeword
else have to use syndrome to get back codeword (“decode”)
Advantages of Linear Codes

• Encoding is efficient (vector-matrix multiply)

• Error detection is efficient (vector-matrix multiply)

• Syndrome \((HY^T)\) has error information

• How to decode? In general, have \(q^{n-k}\) sized table for decoding (one for each syndrome). Useful if \(n-k\) is small, else want (and there exist) other more efficient decoding algorithms.
The d of linear codes

**Theorem:** Linear codes have distance $d$ if every set of $(d-1)$ columns of $H$ are linearly independent, but there is a set of $d$ columns that are linearly dependent.

Proof sketch: Ideas?

For linear codes, distance equals least weight of non-zero codeword. Each codeword gives some collection of columns that must sum to zero.
Example and “Standard Form”

“Standard form” of $G$ for systematic codes: $[I_k \ A]$.

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

$(7,4,3)$ Hamming code
Relationship of G and H

Theorem: For binary codes, if \(G\) is in standard form \([I_k \ A]\) then \(H = [-A^T \ I_{n-k}]\)

Example of (7,4,3) Hamming code:
Relationship of G and H

Proof:

Two parts to prove: (exercise)

1. Suppose that $x$ is a message. Then $H(xG)^T = 0$.

2. Conversely, suppose that $Hy^T = 0$. Then $y$ is a codeword.
Theorem: For every \((n, k, d)_q\) code, \(n \geq (k + d - 1)\)

Another way to look at this: \(d \leq (n - k + 1)\)

(We will not go into the proof of this theorem in this course due to limited time on this topic.)

Codes that meet Singleton bound with equality are called **Maximum Distance Separable (MDS)**
Maximum Distance Separable (MDS)

Only two binary MDS codes!

Q: What are they?

1. Repetition codes (k = 1)
2. Single-parity check codes (n-k = 1)

Need to go beyond the binary alphabet.
Finite fields!