Error-Correcting Codes [Hamming / Shannon]

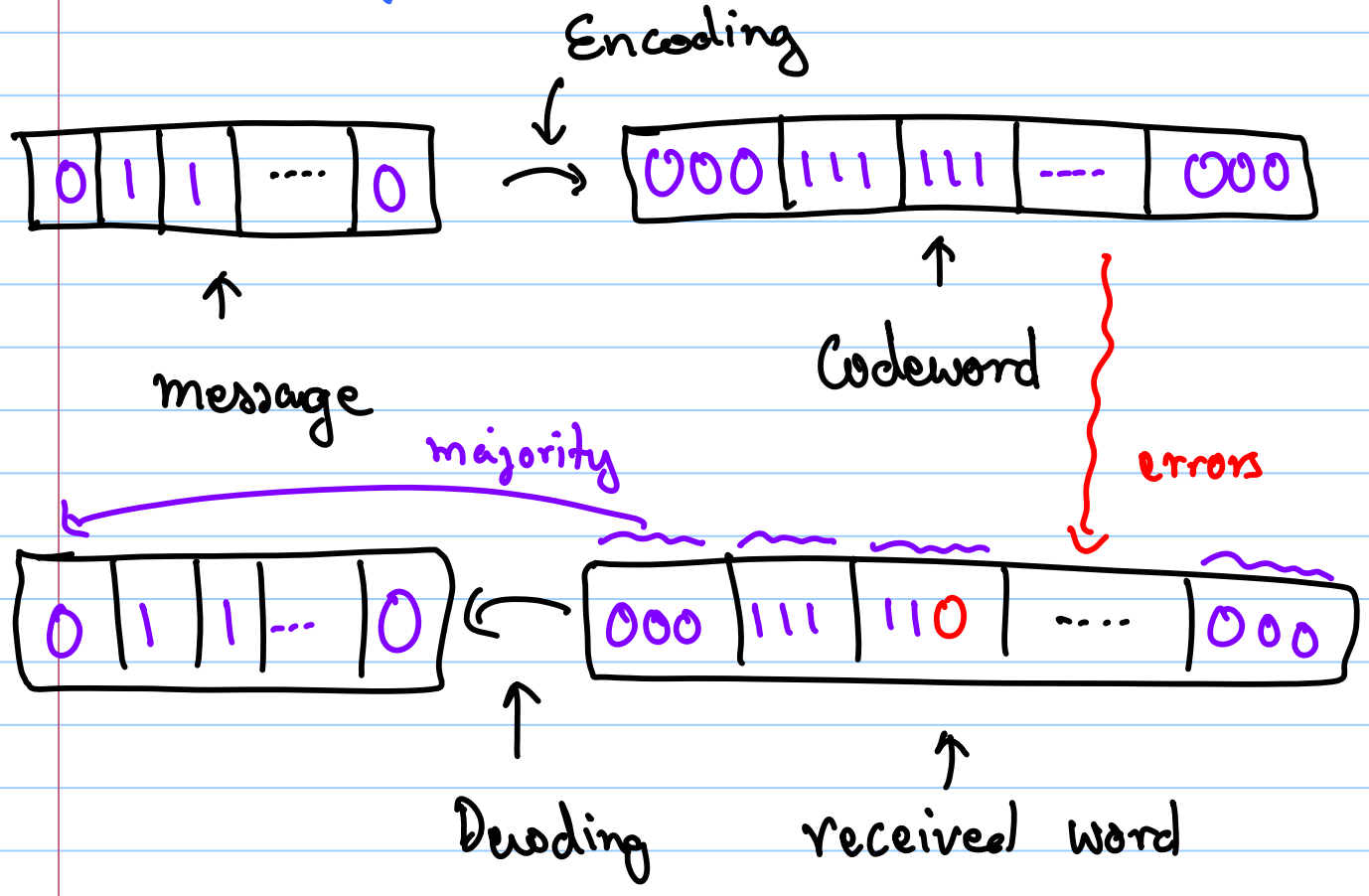
Hamming's Problem (roughly)

- Want to store bits on magnetic storage device
- Bits get corrupted. $0 \to 1$ or $1 \to 0$ but rarely. Say one in every block of 63* bits.
- How can you store information so that it is not lost?

* (Why 63?: Will see....)

Naive Solution

Repeat every bit 3 times



Good news:

- Can encode 21 bit message as 63 bit codeword

- Encoding / Decoding simple (polytime computable)

Bad News

- Rate $\triangleq$ $\frac{\text{message length}}{\text{codeword length}} = \frac{21}{63} = \frac{1}{3}$

Not so great!

Can we do better?

How much better?

Will encoding/decoding be easy?

Hamming Solution - 1

- Break message into 4 bit chunks
- Encode each chunk as follows:

$$m \longrightarrow m \cdot G$$

where $m = \square\square\square\square \in \{0,1\}^4$

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

- Surprising Property:

$\forall\ m_1 \neq m_2$, m_1G & m_2G differ in ≥ 3 bits [Will Prove Later]

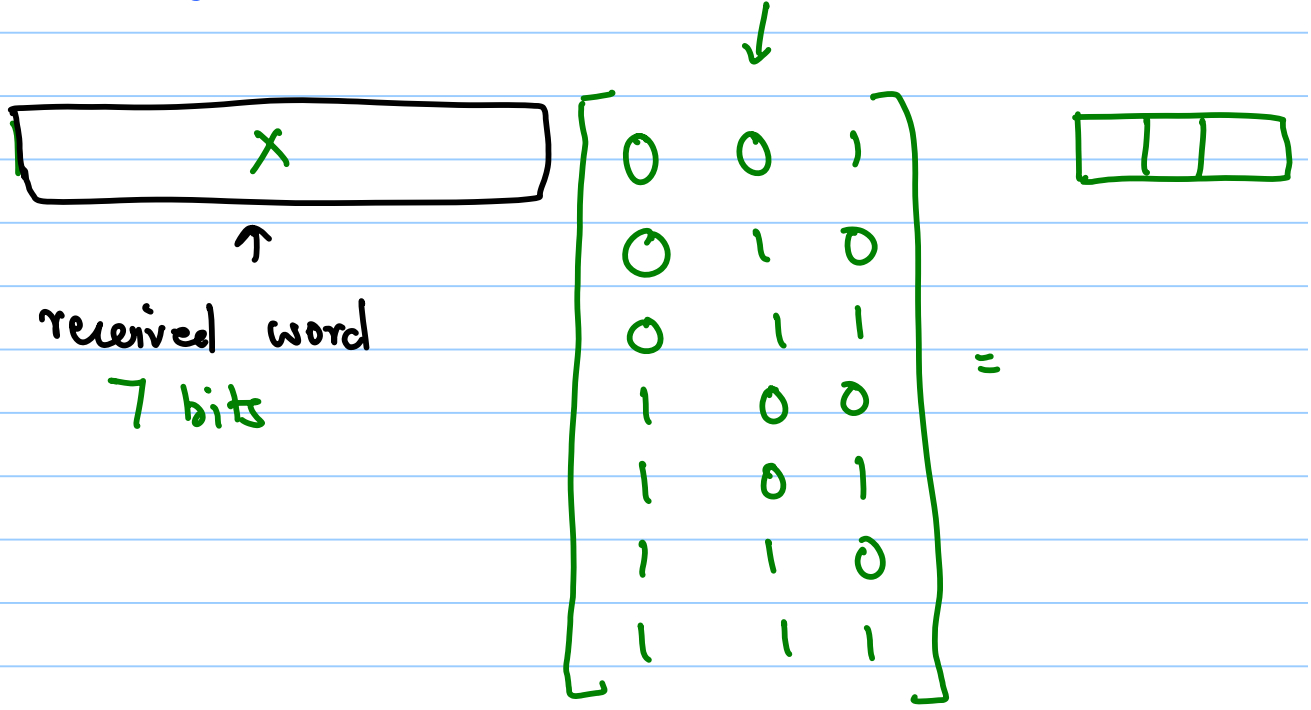
- Rate: 4 bits → 7 bits

  36 bits → 63 bits

  Rate $\approx \frac{4}{7}$ (Will do better)

- Encoding ~ simple

Decoding? More Magic

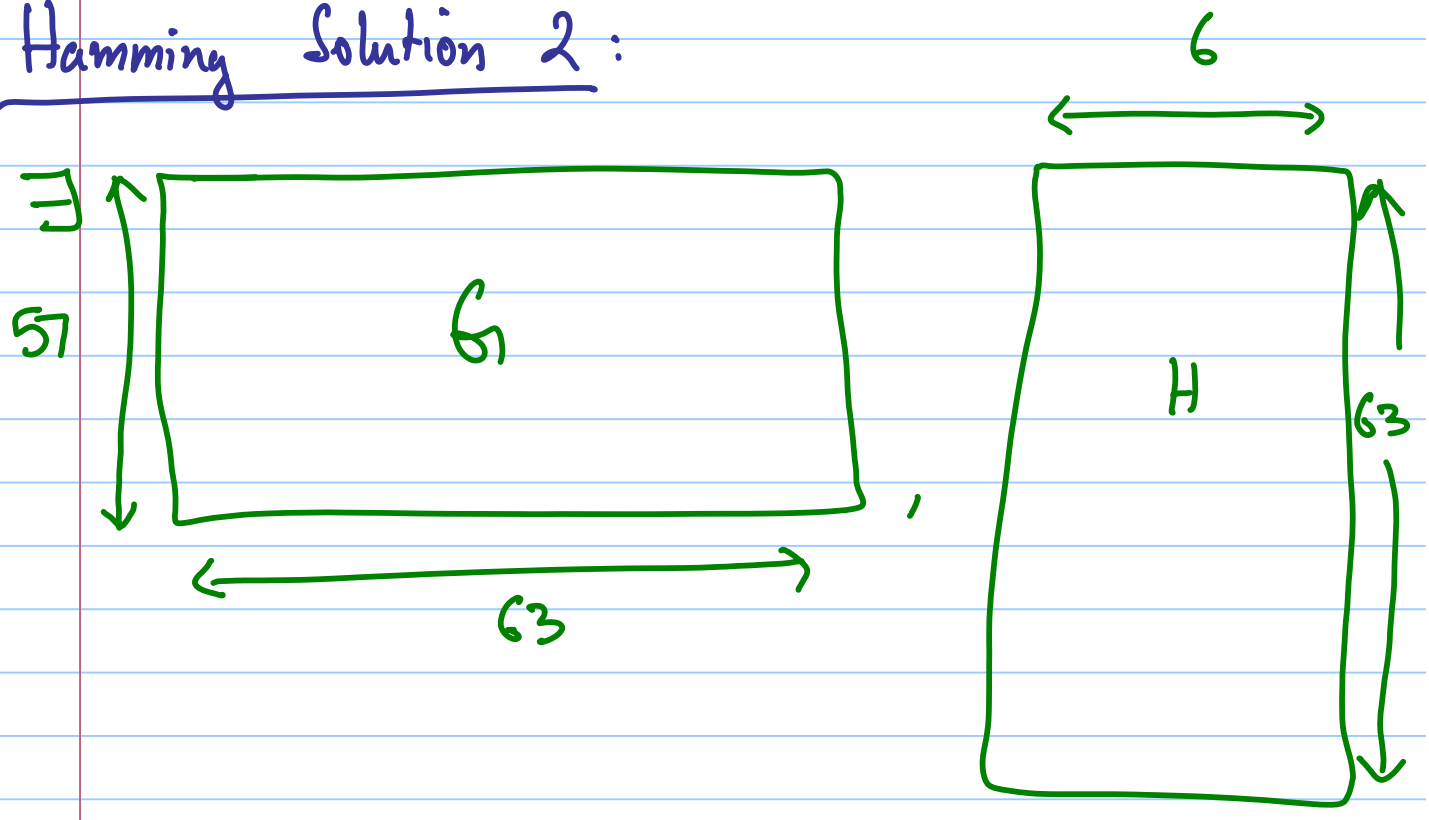


X H = index of flipped bit in binary!

Voila!!

- Is this the best one can do?
- No!

Hamming Solution 2:



Properties:

1. $\forall\ m_1 \neq m_2 \in \{0,1\}^{57}$

   $m_1 G$ & $m_2 G$ differ in ≥ 3 coordinates

2. If x received word with ≤ 1 bit flip, then xH = index of flipped coordinate!

Conclude

- Can achieve rate $= \frac{57}{63}$
- Is this best possible?

  [Hamming]: YES! No Encoding/Decoding scheme does better!

Summary of Hamming's Work

- Construction of "Error-correcting Code"
- Method for Encoding / Decoding
- Proof / Investigation of Optimality / Limits.

  [Modelling critical to prove "optimality"]

Note: [Shannon] has all the same features with slightly different model / emphasis. Will see next week.

Hamming's Notions:

(1) Hamming Distance:

- Let Σ be some finite set
  [Alphabet]
  [$\Sigma = \{0,1\}$ in earlier example;
  $\Sigma = \{0,1\}^8$ (bytes) in CDs]

- Let Σ^n be set of n-letter words over Σ [Ambient Space]

- For $x, y \in \Sigma^n$

$$\Delta(x,y) = \#\text{ coordinates where } x, y \text{ differ}$$

$$= |\{i \mid x_i \neq y_i\}|$$

Δ = Hamming Distance

$\delta(x,y) = \frac{\Delta(x,y)}{n}$. [relativized distance]

- **Fact:** Hamming distance is a metric

  (1) $\Delta(x,y) = 0 \iff x = y$

  (2) $\Delta(x,y) = \Delta(y,x)$

  (3) $\Delta(x,y) + \Delta(y,z) \geq \Delta(x,z)$

- **Conclude:** Can bring in geometric intuition to think about Hamming distance.

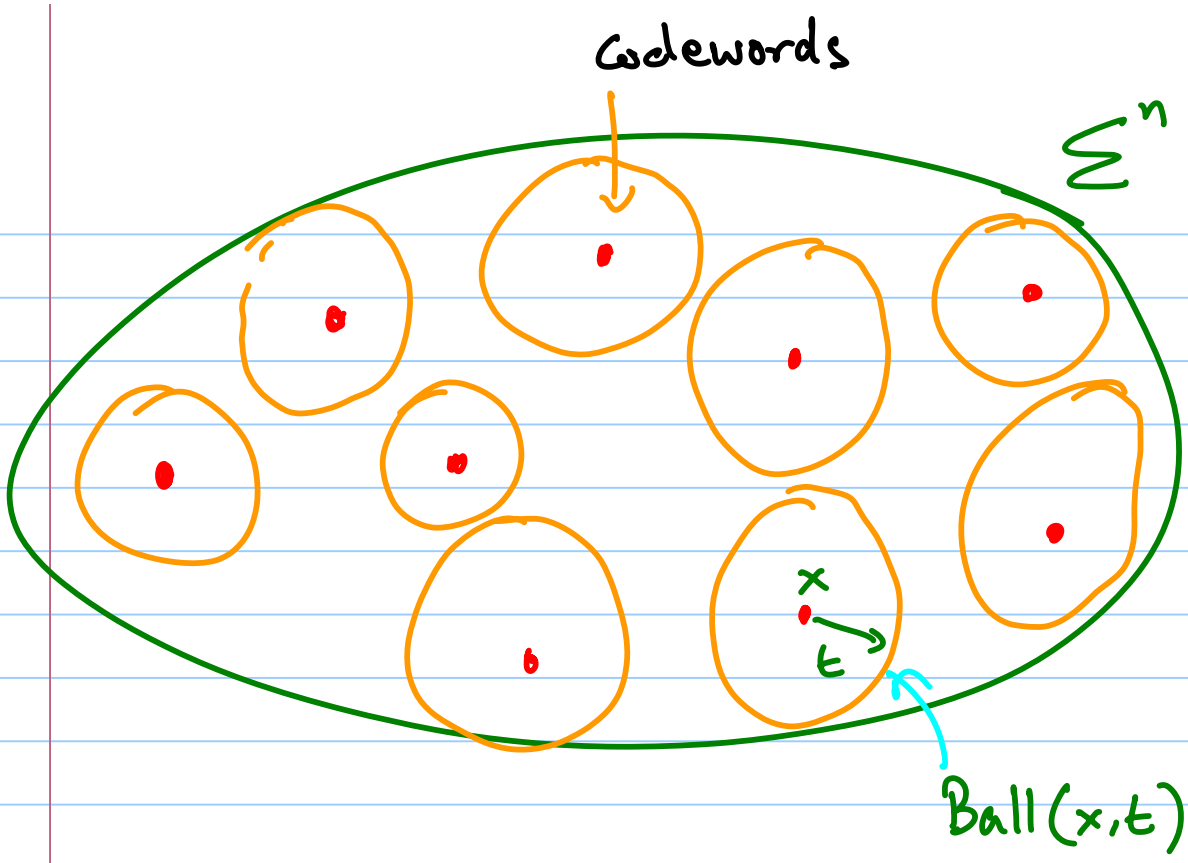
Hamming Notions (Contd.) : Codes

$C \subseteq \Sigma^n$ [set of codewords under some encoding]

- C is t-error correcting if any pattern of upto t errors can be corrected [by some, possibly inefficient, decoding method]

- **Formally :**
  - $\text{Ball}(x,t) = \{ y \in \Sigma^n \mid \Delta(x,y) \leq t \}$
  - C is t-error correcting if $x \neq y \in C$

    $\text{Ball}(x,t) \cap \text{Ball}(y,t) = \phi$.



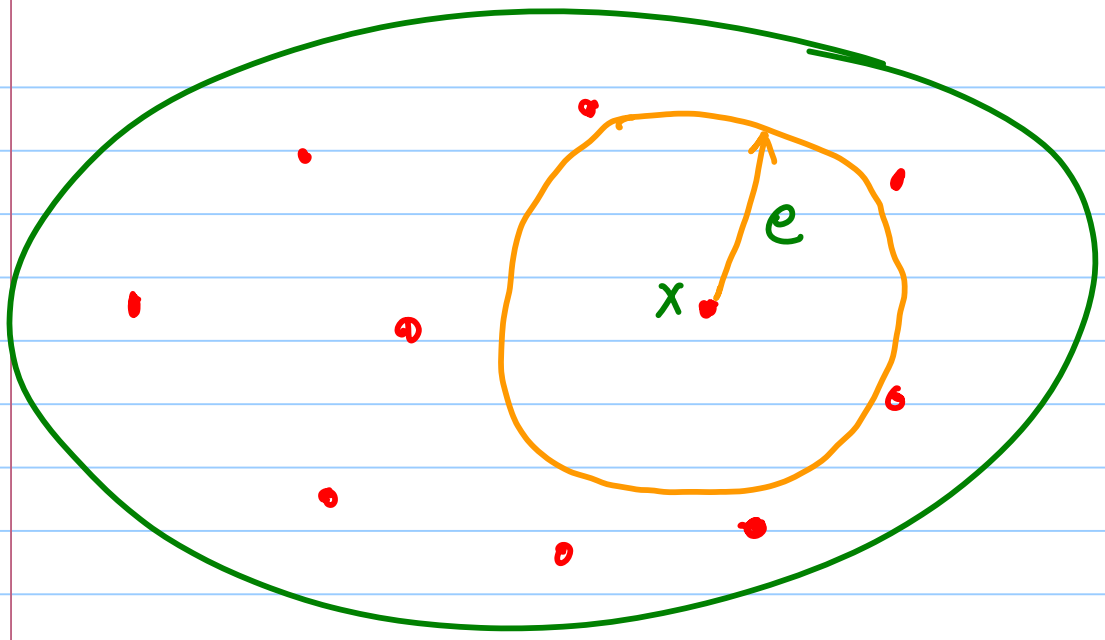
- C is e- error detecting if whenever

$$1 \leq \#\text{ symbol errors} \leq e,$$

it can be detected that errors have occurred.

Formally: $\forall x \in C$

$$\text{Ball}(x,e) \cap C = \{x\}.$$



- $\Delta(C)$ (Distance of Code)

$$= \min_{\substack{x \neq y \\ x,y \in C}} \{\Delta(x,y)\}$$

[Hamming]

Proposition: C is t error-correcting

$\Leftrightarrow$ C is $2t$ error-detecting

$\Leftrightarrow$ Distance of C is $\geq 2t+1$.

Proposition: for $\Sigma = \{0,1\}$ & $x \in \Sigma^n$

$$|\text{Ball}(x,t)| = \sum_{i=0}^{t} \binom{n}{i} \triangleq \text{Vol}(n,t)$$

Proposition: $\Sigma = \{0,1\}$:

if C is t error-correcting then

$$|C| \leq \frac{2^n}{\text{Vol}(n,t)}$$

Can use above to conclude

Rate $= \frac{57}{63}$ is optimal in our example

Rest of this Course:

Follow Hamming's Plan

- Construct Codes (correcting more errors)
- Show Limitations
- Construct decoding algorithms

+

(Non-Hamming Part)

- See how codes are generally useful (in Math. & CS)

Some Claims & Disclaimers

- Motivation is more mathematical & less Engineering; Not a substitute for communication / coding course, but complementary
- It is a graduate class. Some math. maturity is required.
- E.g. Lots of texts & we will follow none!
- Hopefully, will have fun.