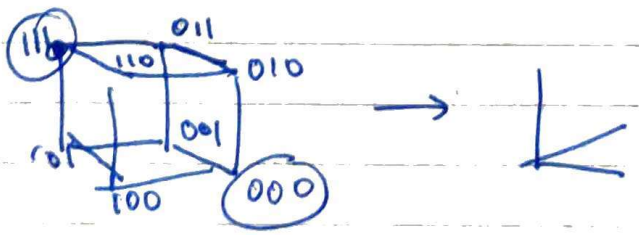
Code based cryptography

→ Meaning error correction codes
→ Digital media is exposed to memory corruption
→ In general k bits are stored in n bits
↳ n-k redundancy
→ Check if the redundant bits are compatible with the rest of the data
↳ Parity Check
↳ If we have good codes then we can correct multiple errors
↳ garuntee of a certain number of errors it can correct

Linear codes

Linear codes
→ Binary code → C → length n and dimension k
→ Generator Matrix → Valid ~~Code~~ words and invalid words
↳ If we have a valid word and there is a 1 letter error → Its easier to correct if the surrounding 1 letter words are mostly invalid rather than valid
→ Similarly with a binary code → 01 → ~~repeating~~ the valid bits are very closed together → Impossible to correct for but if we did 000 → 0 and 111 → 1 then. 101 or 001 are all invalid → now easier to correct



the matrix that takes us from the bit to the higher dimensional stream is called the generator matrix

0[111] = 000
1[111] = 111
↳ generator matrix

→ Minimum distance → determines the error correcting capabilities
↳ Hamming distance between any 2 valid codewords
↳ tells us how many error bits you are guaranteed to be able to fix
↳ distance → 1 → 0 errors detected → 0 corrected
↳ distance → 2 → 1 error detected → 0 corrected

→ Very inefficient
↳ Repitition

3 → 2 → 1
4 → 3 → 1
5 → 3 → 2
6 → 5 → 2

} $d \to d-1 \to \frac{d-1}{2}$

Parity check Matrix
→ if $C^T = 0$ & valid codeword only
→ Syndrom col tells us what errors to fix

Decoding Problem
→ Get a generator matrix and you have a code word c and getting an erroneous vector x → trying to find e or c
→ Find the closest valid word
↳ Goppa codes & leads to efficient decoding
→ If you take a random code then decoding problem is hard
↳ decoding whether the correct code is within t of the closest valid code is hard