CS3236
AY22/23 SEM 1
github/jovyntls

00. INTRODUCTION

data compression

- types of compression
  - **lossless compression** - can recover the contents
  - **lossy compression** - lose some quality - cannot convert back to the higher-quality version
- examples
  - sparse binary string - storing positions of 1s
  - equal number of 0/1s - $L \geq \log_2 \binom{64}{32} \approx 60.7$
  - english text - using relative frequency
  - morse code is NOT binary (contains spaces)
- info theory uses **probabilistic models** (letter frequency, sequence probabilities)
- 2 distinct approaches to compression:
  - **variable length** - map more probable sequences to shorter binary strings
  - **fixed length** - map most probable sequences to strings of a given length
    - insufficient strings for low-probability sequences
    - tradeoff between length/failure probability

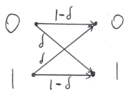
information theory concepts

- speed: **rate** $\to \frac{k}{n}$ (mapping k bits to n bits)
- reliability: $\mathbb{P}[error]$ = $\mathbb{P}[\text{estimated msg} \neq \text{true msg}]$
- **source coding theorem** $\to$ the fundamental compression limit is given by a source-dependent quantity known as the **(Shannon) entropy** H. The (average) storage length can be arbitrarily close to H, but can never be any lower than H.
  - H is a property of the *probability distribution*
- **channel coding theorem** $\to$ there exists a channel-dependent quantity called the **(Shannon) capacity** C such that arbitrarily small error probability can be achieved only for rates $< C$
  - can achieve $\mathbb{P}[error] \leq \epsilon \iff$ rate $< C$

data communication example

- a "transmitter" sends a sequence of 0s and 1s
- a "receiver" sends a sequence *with some corruptions*

channel transition diagram



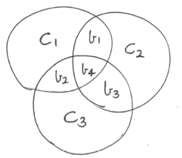
- each bit is flipped independently with probability $\delta \in (0, \frac{1}{2})$

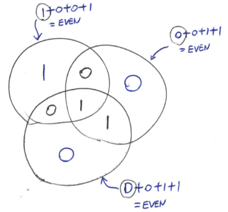
naive

- **uncoded communication** - $\mathbb{P}[correct] = (1-\delta)^N$
- **repetition code** - transmit "000" for "0", "111" for "1"
  - $\mathbb{P}[correct] = [(1-\delta)^3 + 3\delta(1-\delta)^2]^N$
  - more reliable but 3x slower!

Hamming code

- able to correct one bit flip
- maps binary string of length 4 to binary string of length 7
- fill in $b_1b_2b_3b_4$ and assign $c_1c_2c_3$ such that the sum of bits in each circle is even





- $\mathbb{P}[correct] \geq \mathbb{P}[\leq \text{1bit flips}] = (1-\delta)^7 + 7\delta(1-\delta)^6$
- with $\delta = 1$: Shannon capacity $C \approx 0.531$