EC5.102: Information and Communication

(Lec-1)

Source coding-1

(3-March-2025)

Arti D. Yardi

Email address: arti.yardi@iiit.ac.in
Office: A2-204, SPCRC, Vindhya A2, 1st floor

About my teaching style

- I will be using a combination of slides and board.
- Be super interactive.. ask questions..
- Discuss learnings of the class with your friends.
- Refer to the suggested reference books.
- There will be breakout sessions to solve problems (very important).
- There will be self-quizes/surprize-quizes in the class.. :P
- NO LAPTOPS, NO MOBILES in the class!

Calendar

			Mar						Apr		
Mon		3	10	17	24	31		7	14	21	28
Tues		4	11	18	25		1	8	15	22	29
Wed		5	12	19	26		2	9	16	23	30
Thurs		6	13	20	27		3	10	17	24	
Fri		7	14	21	28		4	11	18	25	
Sat	1	8	15	22	29		5	12	19	26	
Sun	2	9	16	23	30		6	13	20	27	

- Note: Quiz-2 on 3-April
- No class on 27-March
- Make-up class: 5-March (Tutorial slot)
- Note: 9 classes before quiz-2, 5 classses after quiz-2
- Quiz-2 on 31-March, Class on 3-April?

References

- Thomas M. Cover and Joy A. Thomas, “Elements of Information Theory”, Wiley India press, Edition 2.
- Raymond Yeung, “A First Course in Information Theory”.
- David J. C. MacKay, “Information Theory, Inference and Learning Algorithms”, Cambridge university press, 2003.

Recap

- Introduction to random variables
  - Random variable (RV), Joint RVs, Conditional RV
  - pmf, pdf, cdf
  - Mean (or expected value) and variance of a RV
- Introduction to information theory
  - Entropy, Joint entropy, Conditional entropy
  - Relative entropy, mutual information
  - Relation between these basic entities

Next agenda

- Block diagram of a digital communication system
  - Analog to digital converter
  - Source coding
  - Channel coding
  - Modulation
  - Communication channel
- Introduction to cryptography
- Introduction to networks

Introduction to source coding (Data compression)

Introduction to data compression

- Suppose you have been given a file with contents:

  a c b a a a d a a b b a a b c d

- To transmit this file, you need to assign a sequence of 0s & 1s to each alphabet.
  - For example, one possible assignment could be

    $a = 0\,0 \quad b = 0\,1 \quad c = 1\,0 \quad d = 1\,1$
  - The file is then given by

    0 0, 1 0, 0 1, 0 0, 0 0, 0 0, 1 1, 0 0, 0 0, 0 1, 0 1, 0 0, 0 0, 0 1, 1 0 1 1
  - The file has 32 bits
- Can you represent this file with fewer number of bits? **Yes!**
- This is **data compression**!
- The process of converting alphabet of a file into bit-sequence is called as **"source encoding"**.

Definition: Source code

- Example of a source code:
  - File: a c b a a a d a a b b a a b c d
  - To transmit this file, we assign: $a = 0\ 0$ $b = 0\ 1$ $c = 1\ 0$ $d = 1\ 1$
  - $0\ 0$ is called as the "codeword" of a.
- **Definition of a source code:**
  - Consider a r.v. X with support set $\mathcal{X}$.
  - Let $\mathcal{D}^*$ be the set of finite-length strings of symbols from a D-ary alphabet. We can assume that the D-ary alphabet $\mathcal{D} = \{0, 1, \ldots, D-1\}$.
  - A source code C is defined as a mapping from $\mathcal{X}$ to $\mathcal{D}^*$.
  - $C(x)$: Codeword of $x \in \mathcal{X}$.
  - $\ell(x)$: Length of $C(x)$
- Encoding and decoding

Expected length of a source code $C(x)$

- Example continued...
  - File: a c b a a a d a a b b a a b c d
  - Source code: a = 0 0   b = 0 1   c = 1 0   d = 1 1
  - What is pmf of X?
- The expected length $L(C)$ a source code $C(x)$ for a r.v. X with pmf $p(x)$ is defined as

$$L(C) = \mathbb{E}_X\Big[\ell(X)\Big] = \sum_{x \in \mathcal{X}} p(x)\ell(x)$$

- Rate = Number of bits after source encoding / Number of symbols = $L(C)$
- **For "good" compression we wish to have $L(C)$ as low as possible!**
  - Can we choose a source code s.t. length of any codeword is say 1?
  - Will it be a "good source code"?
  - How to define a "good"?

How to define a "good" source code?

- Let X be a discrete RV with support set $\mathcal{X}$ and pmf $\{p(x)\}$ where $x \in \mathcal{X}$.
- Consider a source code $C : \mathcal{X} \to \mathcal{D}^*$ with expected length $L(C)$.
- How to define "good" source code?
  - $L(C)$ should be low for good compression.
  - How much low? Hint: We should not loose "information" contained in rv X!!
- A source code is said to be "optimal" if $L(C) = H(X)$.
- Coding efficiency $\eta := H(X) \,/\, L(C)$.
- For "lossless" data compression, $L(C) \geq H(X)$: **Source coding theorem (SCT)** (Note: This is NOT a formal statement!)

Optimal source code for our example

- pmf of X: $\mathbb{P}[X = a] = 1/2$, $\mathbb{P}[X = b] = 1/4$, $\mathbb{P}[X = c] = 1/8$, $\mathbb{P}[X = d] = 1/d$
- Can you construct an optimal source code?
- Consider the following source code:

  a = 0 b = 1 0 c = 1 1 0 d = 1 1 1

  - What is the expected length $L(C) = \sum_{x \in \mathcal{X}} p(x)l(x)$ of this source code?

  $$L(C) = \frac{1}{2} \times 1 + \frac{1}{4} \times 2 + \frac{1}{8} \times 3 + \frac{1}{3} \times 3 = 1.75 \text{ bits}$$

  - What is entropy $H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x)$ this pmf?

  $$H(X) = \frac{1}{2} \times 1 + \frac{1}{4} \times 2 + \frac{1}{8} \times 3 + \frac{1}{3} \times 3 = 1.75 \text{ bits}$$

  - This is an optimal code since $L(C) = H(X)$.
  - Note: This optimal code is a "variable-length" code!
  - Can you construct an optimal "fixed-length" code? **Key idea in SCT!**

Source coding: Questions of interest

- Is there an algorithm to design an optimal code systematically?
- Is optimal code unique?
- What if I don't know the pmf of X?
- What is the intuition behind the result that "for lossless source coding, the minimum value of expected length of source code $L(C)$ is equal to entropy $H(X)$"?
- What will happen if $L(C) < H(X)$?
- Is it desirable to design a source code with $L(C) < H(X)$?
- Can I design a "fixed-length" source code which is optimal?

Self-quiz

- What is a source code? Binary vs D-ary source code?
- How to defind expected length $L(C)$ of source code C?
- When a code C is said to be “optimal”?
- Fixed-length vs variable-length source code
- What is (rough) statement of “source coding theorem”?