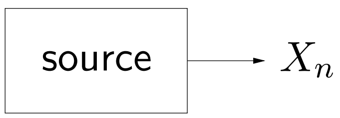
Information Theory

Lecture 2

- Sources and entropy rate: CT4
- Typical sequences: CT3
- Introduction to lossless source coding: CT5.1–5

Information Sources



- *Source data*: a speech signal, an image, a fax, a computer file,. . .
- In practice source data is time-varying and unpredictable.
- Bandlimited continuous-time signals (e.g. speech) can be sampled into discrete time and reproduced without loss.

A *source* $\mathcal{S}$ is defined by a discrete-time *stochastic process* $\{X_n\}$.

- If $X_n \in \mathcal{X}$, $\forall n$, the set $\mathcal{X}$ is the source *alphabet*.
- The source is
  - *stationary* if $\{X_n\}$ is stationary.
  - *ergodic* if $\{X_n\}$ is ergodic.
  - *memoryless* if X_n and X_m are independent for $n \neq m$.
  - *iid* if $\{X_n\}$ is iid (independent and identically distributed).
    - stationary and memoryless $\implies$ iid
  - *continuous* if $\mathcal{X}$ is a continuous set (e.g. the real numbers).
  - *discrete* if $\mathcal{X}$ is a discrete set (e.g. the integers $\{0, 1, 2, \ldots, 9\}$).
  - *binary* if $\mathcal{X} = \{0, 1\}$.

- Consider a source $\mathcal{S}$, described by $\{X_n\}$. Define

$$X_1^N \triangleq (X_1, X_2, \ldots, X_N).$$

- The *entropy rate* of $\mathcal{S}$ is defined as

$$H(\mathcal{S}) \triangleq \lim_{N\to\infty} \frac{1}{N} H(X_1^N)$$

  (when the limit exists).
- $H(X)$ is the entropy of a single random variable X, while entropy rate defines the "entropy per unit time" of the *stochastic process* $\mathcal{S} = \{X_n\}$.

- A *stationary* source $\mathcal{S}$ always has a well-defined entropy rate, and it furthermore holds that

$$H(\mathcal{S}) = \lim_{N\to\infty} \frac{1}{N} H(X_1^N) = \lim_{N\to\infty} H(X_N | X_{N-1}, X_{N-2}, \dots, X_1).$$

  That is, $H(\mathcal{S})$ is a measure of the *information gained when observing a source symbol, given knowledge of the infinite past.*

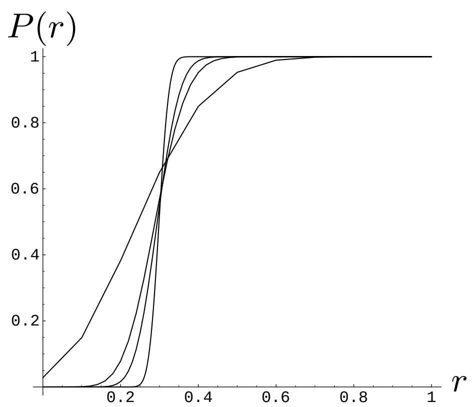
- We note that for iid sources

$$H(\mathcal{S}) = \lim_{N\to\infty} \frac{1}{N} H(X_1^N) = \lim_{N\to\infty} \frac{1}{N} \sum_{m=1}^{N} H(X_m) = H(X_1)$$

- Examples (from CT4): Markov chain, Markov process, Random walk on a weighted graph, hidden Markov models,...

Typical Sequences

- A binary iid source $\{b_n\}$ with $p = \Pr(b_n = 1)$
- Let R be the number of 1:s in a sequence, $b_1, \dots, b_N$, of length $N \implies p(b_1^N) = p^R(1-p)^{N-R}$
- $P(r) \triangleq \Pr(\frac{R}{N} \le r)$ for $N = 10, 50, 100, 500$, with $p = 0.3$,



- As N grows, the probability that a sequence will satisfy $R \approx p \cdot N$ is high $\implies$ given a b_1^N that the source produced, it is likely that

$$p(b_1^N) \approx p^{pN}(1-p)^{(1-p)N}$$

In the sense that the above holds with high probability, the "source will only produce" sequences for which

$$\frac{1}{N}\log p(b_1^N) \approx p\log p + (1-p)\log(1-p) = -H$$

That is, for large N it holds with high probability that

$$p(b_1^N) \approx 2^{-N \cdot H}$$

where H is the entropy (entropy rate) of the source.

- A general discrete source that produces iid symbols X_n, with $X_n \in \mathcal{X}$ and $\Pr(X_n = x) = p(x)$. For all $x_1^N \in \mathcal{X}^N$ we have

$$\log p(x_1^N) = \log p(x_1, \ldots, x_N) = \sum_{m=1}^{N} \log p(x_m).$$

For an arbitrary *random* sequence X_1^N we hence get

$$\lim_{N\to\infty} \frac{1}{N}\log p(X_1^N) = \lim_{N\to\infty} \frac{1}{N}\sum_{m=1}^{N}\log p(X_m) = E\log p(X_1) \quad \text{a.s.}$$

by the (strong) law of large numbers. That is, for large N

$$p(X_1^N) \approx 2^{-N \cdot H(X_1)}$$

holds with high probability.

- The result (the *Shannon–McMillan–Breiman Theorem*) can be extended to (discrete) *stationary* and *ergodic* sources (CT16.8). For a stationary and ergodic source, $\mathcal{S}$, it holds that

$$-\lim_{N\to\infty} \frac{1}{N} \log p(X_1^N) = H(\mathcal{S}) \quad \text{a.s.}$$

where $H(\mathcal{S})$ is the *entropy rate* of the source.

- We note that $p(X_1^N)$ is a *random variable*. However, the right-hand side of

$$p(X_1^N) \approx 2^{-N \cdot H(\mathcal{S})}$$

is a *constant*

$\Longrightarrow$ a *constraint* on the sequences the source "typically" produces!

The Typical Set

- For a given stationary and ergodic source $\mathcal{S}$, the *typical set* $A_\varepsilon^{(N)}$ is the set of sequences $x_1^N \in \mathcal{X}^N$ for which

$$\boxed{2^{-N(H(\mathcal{S})+\varepsilon)} \le p(x_1^N) \le 2^{-N(H(\mathcal{S})-\varepsilon)}}$$

1. $x_1^N \in A_\varepsilon^{(N)} \Rightarrow -N^{-1} \log p(x_1^N) \in [H(\mathcal{S}) - \varepsilon, H(\mathcal{S}) + \varepsilon]$
2. $\Pr(X_1^N \in A_\varepsilon^{(N)}) > 1 - \varepsilon$, for N sufficiently large
3. $|A_\varepsilon^{(N)}| \le 2^{N(H(\mathcal{S})+\varepsilon)}$
4. $|A_\varepsilon^{(N)}| \ge (1-\varepsilon) 2^{N(H(\mathcal{S})-\varepsilon)}$, for N sufficiently large

That is, a large N and a small ε gives

$$\Pr(X_1^N \in A_\varepsilon^{(N)}) \approx 1, \quad |A_\varepsilon^{(N)}| \approx 2^{N\,H(\mathcal{S})}$$
$$p(x_1^N) \approx |A_\varepsilon^{(N)}|^{-1} \approx 2^{-N\,H(\mathcal{S})} \quad \text{for } x_1^N \in A_\varepsilon^{(N)}$$

The Typical Set and Source Coding

1. Fix ε (small) and N (large). Partition $\mathcal{X}^N$ into two subsets: $A = A_\varepsilon^{(N)}$ and $B = \mathcal{X}^N \setminus A$.
2. Observed sequences will "typically" belong to the set A. There are $M = |A| \leq 2^{N(H(\mathcal{S})+\varepsilon)}$ elements in A.
3. Let the different $i \in \{0, \dots, M-1\}$ enumerate the elements of A. An index i can be stored or transmitted spending no more than $\lceil N \cdot (H(\mathcal{S}) + \varepsilon) \rceil$ bits.
4. *Encoding*. For each observed sequence x_1^N
   1. if $x_1^N \in A$ produce the corresponding index i.
   2. if $x_1^N \in B$ let $i = 0$.
5. *Decoding*. Map each index i back into $A \subset \mathcal{X}^M$.

- An error appears with probability $\Pr(X_1^N \in B) \leq \varepsilon$ for large $N \implies$ the probability of error can be made to vanish as $N \to \infty$
- An "almost noiseless" source code that maps x_1^N into an index i, where i can be represented using at most $\lceil N \cdot (H(\mathcal{S}) + \varepsilon) \rceil$ bits. However, since also $M \geq (1-\varepsilon)2^{N(H(\mathcal{S})-\varepsilon)}$, for a large enough N, we need at least $\lfloor \log(1-\varepsilon) + N(H(\mathcal{S}) - \varepsilon) \rfloor$ bits.
- Thus, for large N it is possible to design a source code with rate

$$H(\mathcal{S}) - \varepsilon + \frac{1}{N}\big(\log(1-\varepsilon) - 1\big) < R \leq H(\mathcal{S}) + \varepsilon + \frac{1}{N}$$

bits per source symbol.

$\implies$ "Operational" meaning of entropy rate: *the smallest rate at which a source can be coded with arbitrarily low error probability*.

Data Compression

- For large N it is possible to design a source code with rate

$$H(\mathcal{S}) - \varepsilon + \frac{1}{N}\big(\log(1-\varepsilon) - 1\big) < R \leq H(\mathcal{S}) + \varepsilon + \frac{1}{N}$$

  bits per symbol, having a vanishing probability of error.
    - The above is an *existence result*; it doesn't tell us *how* to design codes.
- For a fixed finite N, the typical-sequence codes discussed are "almost noiseless" fixed-length to fixed-length codes.
- We will now start looking at concrete "zero-error" codes, their performance and how to design them.
    - Price to pay to get zero errors: fixed-length to *variable*-length

Various Classifications

- Source alphabet
    - *Discrete sources*
    - Continuous sources
- Recovery requirement
    - *Lossless* source coding
    - Lossy source coding
- Coding method
    - Fixed-length to fixed-length
    - *Fixed-length to variable-length*
    - Variable-length to fixed-length
    - Variable-length to variable-length

Zero-Error Source Coding

- *Source coding theorem* for symbol codes (today)
  - Symbol codes, code extensions
  - Uniquely decodable and instantaneous (prefix) codes
  - Kraft(-McMillan) inequality
  - Bounds on the optimal codelength
  - Source coding theorem for zero-error prefix codes
- Specific code constructions (next time)
  - Symbol codes: Huffman codes, Shannon-Fano codes
  - Stream codes: arithmetic codes, Lempel-Ziv codes

What Is a Symbol Code?

- D-ary symbol code C for a random variable X

$$C\colon \mathcal{X} \to \{0, 1, \ldots, D-1\}^*$$

  - $\mathcal{A}^*$ = set of finite-length strings of symbols from a finite set $\mathcal{A}$
  - $C(x)$ codeword for $x \in \mathcal{X}$
  - $l(x)$ length of $C(x)$ (i.e. number of D-ary symbols)
- Data compression $\implies$ minimize *expected length*

$$L(C, X) = \sum_{x \in \mathcal{X}} p(x) l(x)$$

- Extension of C is $C^*\colon \mathcal{X}^* \to \{0, 1, \ldots, D-1\}^*$

$$C^*(x_1^n) = C(x_1)C(x_2)\cdots C(x_n), \ \ n = 1, 2, \ldots$$

Example: Encoding Coin Flips

$\mathcal{X}$				Problem
C_0	0	1	10	010
C_u	00	1	10	$10\cdots0$
C_i	00	1	01	–

Uniquely Decodable Codes

- C is *uniquely decodable* if

$$\forall \mathbf{x}, \mathbf{y} \in \mathcal{X}^*, \qquad \mathbf{x} \neq \mathbf{y} \implies C^*(\mathbf{x}) \neq C^*(\mathbf{y})$$

- Any uniquely decodable code must satisfy the Kraft inequality

$$\sum_{x \in \mathcal{X}} D^{-l(x)} \leq 1$$

(McMillan's result, Karush's proof in C&T)

Instantaneous Codes

- C is *instantaneous* (or prefix) if prefix-free
  - no codeword is a prefix of any other codeword
- Instantaneous codes are uniquely decodable

$\Longrightarrow$ prefix codes satisfy the Kraft inequality

- Given a set of codeword lengths that satisfy the Kraft inequality there exists a prefix code with those codeword lengths.

$\Longrightarrow$ there is a prefix code for every set of codeword lengths that allow a uniquely decodable code

$\Longrightarrow$ no loss of generality in studying only prefix codes

Most Compression Possible?

For any uniquely decodable D-ary symbol code C (defining $H_D(X) \triangleq -\sum_x p(x) \log_D p(x)$),

$$
\begin{aligned}
L(C, X) &= \sum_{x \in \mathcal{X}} p(x) \log_D D^{l(x)} \\
&= H_D(X) + \sum_{x \in \mathcal{X}} p(x) \log_D \frac{p(x)}{D^{-l(x)}} \\
&\overset{\text{log-sum}}{\geq} H_D(X) + 1 \cdot \log_D \frac{1}{\sum_{x \in \mathcal{X}} D^{-l(x)}} \\
&\overset{\text{Kraft}}{\geq} H_D(X)
\end{aligned}
$$

with equality iff $p(x) = D^{-l(x)}$, i.e. $l(x) = -\log_D p(x)$.

How Close Can We Get?

- The optimal length $l(x) = \log_D \frac{1}{p(x)}$ need not be an integer
- Use $l(x) = \left\lceil \log_D \frac{1}{p(x)} \right\rceil$
- These codeword lengths satisfy the Kraft inequality

$$\sum_{x \in \mathcal{X}} D^{-\left\lceil \log_D \frac{1}{p(x)} \right\rceil} \leq \sum_{x \in \mathcal{X}} D^{-\log_D \frac{1}{p(x)}} = \sum_{x \in \mathcal{X}} p(x) = 1$$

$\Longrightarrow$ There exists a (uniquely decodable) prefix code with these codeword lengths

- For such a code C,

$$l(x) < -\log_D p(x) + 1 \implies L(C, X) < H_D(X) + 1$$

Source Coding Theorem

Uniquely Decodable Zero-Error Codes

- The best uniquely decodable D-ary symbol code can compress to within 1 symbol of the entropy

$$\min_{C \text{prefix}} L(C, X) \in [H_D(X), H_D(X) + 1)$$

- Coding blocks of source symbols gives

$$\min_{C \text{prefix}} L(C, X_1^n) \in [H_D(X_1^n), H_D(X_1^n) + 1)$$

- The minimum expected codeword length *per symbol* satisfies

$$\min_{C \text{prefix}} \frac{L(C, X_1^N)}{N} \to H_D(\mathcal{S}),$$

where $H_D(\mathcal{S})$ is the *entropy rate* (base D) of the source.

Penalty for the Wrong Code

- $X \sim p(x)$
- C_q: $l(x) = \left\lceil \log \frac{1}{q(x)} \right\rceil$
- Using C_q to code X, the expected codeword length satisfies

$$H(p) + D(p\|q) \le L(C_q, X) \le H(p) + D(p\|q) + 1$$

$\Longrightarrow$ $D(p\|q)$ is the penalty for mismatch

$$L_q \approx \mathrm{E}_p \log \frac{1}{q(X)} = \mathrm{E}_p \log \frac{p(X)}{p(X)q(X)} = \mathrm{E}_p \log \frac{1}{p(X)} + \mathrm{E}_p \log \frac{p(X)}{q(X)}$$