Lecture-04: Data Compression and Transmission

1 Data Compression

Consider a source that generates a sequence of symbols taking values in a finite alphabet $\mathcal{X}$. Such a source is typically modeled by a random process $(X_t : t \in \mathbb{N})$. For the purpose of communication and storage, it is desirable to encode this sequence using as few bits as possible. If exact reconstruction is required, then this is known as **lossless source coding**.

The most natural approach to this problem is to encode length-N source blocks $(X_1, \ldots, X_N)$ into variable-length blocks of bits. Let $\{0,1\}^* = \cup_{n=0}^{\infty} \{0,1\}^n$ denote the set of finite-length binary strings. Then, the source encoder is a function

$$\begin{aligned} w : \mathcal{X}^N &\to \{0,1\}^* \\ x &\mapsto w(x). \end{aligned}$$

If the source sequence consists of the length-N blocks $x^{(1)}, x^{(2)}, \ldots, x^{(r)}$, then the encoded sequence is the concatenation $w(x^{(1)}), w(x^{(2)}), \ldots, w(x^{(r)})$. Since the output sequence does not add markers between blocks, one must choose the w carefully to guarantee decodability. The standard approach is use to **prefix-free** (or **instantaneous**) codes where no codeword is a prefix of another codeword. This allows the decoder to uniquely reconstruct the codeword boundaries.

An important property of a code is its average length. If $l_w(x)$ is the length of $w(x)$ in bits, then the average length of an encoded block

$$L(w) = \sum_{x \in \mathcal{X}^N} P_N(x) l_w(x) \text{ bits.}$$

Any prefix-free code can be represented by a binary tree whose leaf nodes are labeled by codewords. To construct a prefix-free code, one can draw a binary tree and sequentially assign x values to nodes. After each assignment, all children of the assigned node are removed.

Exercise 1.1. Is there a prefix-free code with codeword lengths 1, 2, 3, 3? How about 2, 2, 3, 3, 3, 4, 4, 4? Try constructing a code for each case.

Lemma 1.2 (Kraft Inequality). *A prefix-free source code with length function $l_w(x)$ exists iff $\sum_{x \in \mathcal{X}^N} 2^{-l_w(x)} \leqslant 1$.*

Proof. We will show that if $w : \mathcal{X}^N \to \{0,1\}^*$ is a prefix-free encoding map, then it satisfies the Kraft inequality. Let $l_{\max} = \max_{x \in \mathcal{X}^N} l_w(x)$ and recall that a binary tree has exactly 2^l nodes at depth l. Since w is a prefix-free code, $w(x)$ is one of the nodes in the binary tree at depth $l_w(x)$, and none of its children belong to the code. This node has $2^{l_{\max}} - 2^{l_w(x)}$ leaf nodes at the depth $l_{\max}$.

Conversely, we can show that if there is a sequence of lengths $l_w(x)$ for messages $x \in \mathcal{X}^N$, satisfying the Kraft inequality, then there exists a prefix-free encoding map $w : \mathcal{X}^N \to \{0,1\}^*$ such that $l_w(x)$ is the length of the code $w(x)$. To construct a code, one starts with the complete binary tree of depth $l_{\max}$. Then, for each $x \in \text{supp}(P_N)$ (in order of increasing length), one assigns x to a codeword $w(x)$ of length $l_w(x)$. For an x with length $l_w(x)$, one finds an available node at depth $l_w(x)$, assigns the binary label of that node to $w(x)$, and then removes all children of that node.

Assigning a codeword of length $l_w(x)$ removes exactly $2^{l - l_w(x)}$ nodes at depth l for $l \geqslant l_w(x)$. Thus, this process succeeds up to depth l if and only if

$$\sum_{x : l_w(x) \leqslant l} 2^{l - l_w(x)} \leqslant 2^l.$$

□

Theorem 1.3 (Source Coding Theorem). *For the distribution $P_N(x)$, let L_N^* be average length of an encoded block for the prefix-free code with the minimum average length. Then,*

$$H(X) \leqslant L_N^* \leqslant H(X) + 1.$$

Proof. Let $l_w(x)$ be the length function for a valid prefix-free code and define $Q_N(x) = 2^{-l_w(x)}$. Since $l_w(x)$ must satisfy the Kraft inequality, it follows that $\sum_x Q_N(x) \leqslant 1$. Using non-negativity of KL-divergence (which holds even if $\sum_x q(x) \leqslant 1$), we see that the average code length, L, satisfies

$$L - H(P_N) = \sum_{x\in\mathcal{X}} P_N(x) \left(l_w(x) - \log_2 \frac{1}{P_N(x)} \right) = \sum_{x\in\mathcal{X}} P_N(x) \log_2 \frac{P_N(x)}{Q_N(x)} = D(P_N \| Q_N) \geqslant 0.$$

If we choose $l_w(x)$ to be the length function for a code that achieves the optimal $L = L_N^*$, then this implies that $L_N^* \geqslant H(P_N)$. To achieve the upper bound, we design a code with $l_w(x) = \lceil \log_2 P_N(x) \rceil$ and compute

$$L_N^* \leqslant \sum_{x\in\mathcal{X}} P_N(x) \lceil -\log_2 P_N(x) \rceil \leqslant \sum_{x\in\mathcal{X}} P_N(x) \left(1 + \log_2 \frac{1}{P_N(x)} \right) = H(X) + 1.$$

Together, these complete the proof. □

Remark 1.4. This shows that one operational definition of the entropy is "the minimum average length of any variable length code that can be used to reconstruct X".

Remark 1.5. In theory, one can use source coding theorem to achieve optimal compression rate for i.i.d. sequences. To see this, we observe that $H(P_N) = NH(X_1)$. Thus, by increasing N, the constructive upper bound in the theorem gives a compression rate (i.e., bits per source symbol) of

$$\frac{L_N^*}{N} \leqslant H(X_1) + \frac{1}{N}.$$

Example 1.6. Let us consider what happens if a code is designed for a different distribution, Q_N, and then used with the distribution P_N. From source coding theorem, we see that the average code length, L, must satisfy $L \geqslant H(P_N) + D(P_N \| Q_N)$. On the other hand, if the lengths are chosen to be $l_w(x) = \lceil -\log_2 Q_N(x) \rceil$, then the average length satisfies

$$L = \sum_{x\in\mathcal{X}} P_N(x) \lceil -\log_2 Q_N(x) \rceil \leqslant \sum_{x\in\mathcal{X}} P_N(x) \left(\log_2 \frac{1}{Q_N(x)} + 1 \right) = H(X) + 1 + D(P_N \| Q_N).$$

Remark 1.7. This shows that one operational definition of the divergence $D(P_N \| Q_N)$ is "the increase in average length associated with designing a code for Q_N when the true distribution is P_N".

2 Data Transmission

In engineering, one often wants to communicate information across an unreliable medium. For example, think of a system that modulates the current in a wire (by adjusting the voltage at one end) and measures the current at the other end. Due to thermal fluctuations, the difference between the modulated current at the measured current will always contain some randomness. One can analyze this situation by first discretizing time and then defining a simple mathematical model.

Definition 2.1. A **discrete memoryless channel** (DMC) is defined by a finite input alphabet $\mathcal{X}$, a finite output alphabet $\mathcal{Y}$, and a conditional probability distribution $Q(y|x)$. For $N \in \mathbb{N}$ channel uses, let the channel input vector be a random vector $X = (X_1, \ldots, X_N) \in \mathcal{X}^N$. Then, the channel output vector is a random vector $Y = (Y_1, \ldots, Y_N) \in \mathcal{Y}^N$ where

$$Q_N(y|x) \triangleq P(Y = y | X = x) = \prod_{t=1}^{N} Q(y_i|x_i).$$

Example 2.2. For example, the **binary symmetric channel** (BSC) with error probability ρ has $\mathcal{X} = \mathcal{Y} = \{0,1\}$ and is defined by

$$Q(y|x) = (1-\rho)\mathbb{1}_{\{x=y\}} + \rho\mathbb{1}_{\{x\neq y\}}.$$

Example 2.3. For example, the **binary erasure channel** (BEC) with erasure probability ϵ has $\mathcal{X} = \{0,1\}, \mathcal{Y} = \{0,1,*\}$, and is defined by

$$Q(y|x) = \epsilon\mathbb{1}_{\{y=*\}} + (1-\epsilon)\mathbb{1}_{\{y=x\}}.$$

Channel coding is the process of improving performance by adding redundancy (e.g., by encoding an K bit message into $N > K$ bits).

Definition 2.4. For binary-input channel, a length-N **code** carrying an K-bit message is defined by an encoder that maps $m \in \{0,1\}^K$ to a codeword $x^{(m)} \in \{0,1\}^N$. The ratio $R = K/N$ is called the rate (in information bits per channel use) of the code. A message decoder $x^d : \mathcal{Y}^N \to \{0,1\}^K$ is a mapping from the channel output to one of the possible input messages.

Example 2.5. For a BSC, the simplest approach is to simply repeat each bit N times and decode via majority vote (i.e., $x^{(0)} = 00\ldots00, x^{(1)} = 11\ldots11$, and $x^d(y) = \mathbb{1}_{\{\sum_{i=1}^N y_i > N/2\}}$. In this case, the original bit will be recovered correctly as long as there are no more than $\lfloor N/2 \rfloor$ errors. Thus, one can achieve arbitrary reliability by increasing N. But, increasing N also reduces the rate of communication.

Definition 2.6. For a code/decoder pair, the block error probability of message m is the probability,

$$P_B(m) = \sum_{y\in\mathcal{Y}^N} Q_N(y|x^{(m)})\mathbb{1}_{\{x^d(y)\neq m\}},$$

that decoder does not return m when message m is transmitted. The maximum and average block error probabilities are denoted by $P_B^{\max} \triangleq \max_{m\in\{0,1\}^K} P_B(m)$ and $P_B^{\text{av}} \triangleq \frac{1}{2^M}\sum_{m\in\{0,1\}^K} P_B(m)$ respectively.

Definition 2.7. A code rate R is **achievable** if there exists a sequence of encoder/decoder pairs with rate $R_N \to R$ and block error rate $P_{B,N} \to 0$. The **channel capacity** is the supremum of all achievable code rates.

Remark 2.8. One can get a qualitative feel for achievable rates via the following argument. The key is that, for i.i.d. sequences $(X_1,\ldots,X_N)$ with large N, the probability distribution essentially becomes uniform over a set of $2^{NH(X)}$ "typical" sequences. Thus, for $(X,Y) \sim p(x)Q(y|x)$, the i.i.d. sequence $((X_1,Y_1),...,(X_N,Y_N))$ takes one of $2^{NH(X,Y)}$ different typical values with essentially uniform probability. If we ignore the X values, then the number of $(Y_1,\ldots,Y_N)$ typical sequences is roughly $2^{NH(Y)}$. If we fix the $(X_1,\ldots,X_N)$ sequence to a typical value $(x_1,\ldots,x_N)$, then the number of $((x_1,Y_1),\ldots,(x_N,Y_N))$ typical sequences is roughly $2^{NH(Y|X=x)} = 2^{NH(Y|X)}$. This last set of sequences can be seen as the likely set of output sequences if x is transmitted. Thus, if the likely output sets of each codeword fill the space but do not overlap, then we get $2^{NR}2^{NH(Y|X)} = 2^{NH(Y)}$ or $R = H(Y) - H(Y|X) = I(X;Y)$.

Theorem 2.9 (Channel Coding Theorem). *For a DMC, the channel capacity is given by*

$$C \triangleq \max_{p(x)} I(X;Y) = \max_{p(x)} \sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} p(x)Q(y|x)\log_2\frac{Q(y|x)}{\sum_{x'} p(x')Q(y|x')}.$$

Thus, for any $R \leqslant C$, there exists a sequence of encoder/decoder pairs such that $R_N \to R$ and $P_{B,N} \to 0$. Conversely, if a sequence of encoder/decoder pairs satisfies $R_N \to R$ and $P_{B,N} \to 0$, then $R \leqslant C$.

Proof. Achievability will be shown in a later lecture. The following converse demonstrates the power and simplicity of information theory. For any length-N encoder/decoder pair, let M be a uniform random message, $X = x^{(M)}$ be its encoded codeword, Y be the channel output, and $\hat{M} = x^d(Y)$ be the decoded message. Since $M - X - Y - \hat{M}$ form a Markov chain, we have from the successive applications of $H(M) = K = NR$, definition of $I(M;\hat{M})$, Fano's inequality, $\mathcal{H}(P_B) \leqslant 1$, data processing inequality, and lemma on successive channel use,

$$\begin{aligned} NR = H(M) &= H(M|\hat{M}) + I(M;\hat{M}) \leqslant \mathcal{H}(P_B) + P_B \log_2(2^{NR} - 1) + I(M;\hat{M}) \leqslant 1 + P_B NR + I(M;\hat{M}) \\ &\leqslant 1 + P_B NR + I(X;Y) \leqslant 1 + P_B NR + NC. \end{aligned}$$

Solving for an upper bound on R, we find that $R \leqslant \frac{1}{1-P_B}\left(C + \frac{1}{N}\right)$. Thus, $R_N \to R \leqslant C$ for any sequence where $N \to \infty$ and $P_{B,N} \to 0$. □

Exercise 2.10. Verify that the capacity of the BEC(ϵ) channel is $C = 1 - \epsilon$ and the capacity of the BSC(ρ) channel is $C = 1 - \mathcal{H}(\rho)$. Hint: Use $I(X;Y) = H(Y) - H(Y|X)$.

Lemma 2.11. *For N channel uses on a DMC, $I(X;Y) \leqslant NC$.*

Proof. This follows from successive applications of chain rule of entropy, memorylessness of channel, reduction of entropy due to conditioning, and mutual information upper-bounded by capacity,

$$\begin{aligned} I(X;Y) = H(Y) - H(Y|X) &= \sum_{i=1}^{N} H(Y_i|Y_1,\ldots,Y_{i-1}) - \sum_{i=1}^{N} H(Y_i|Y_1,\ldots,Y_{i-1},X) \\ &= \sum_{i=1}^{N} H(Y_i|Y_1,\ldots,Y_{i-1}) - \sum_{i=1}^{N} H(Y_i|X_i) \leqslant \sum_{i=1}^{N} H(Y_i) - \sum_{i=1}^{N} H(Y_i|X_i) \leqslant \sum_{i=1}^{N} I(X_i;Y_i) \leqslant NC. \end{aligned}$$

□