Queens' College Cambridge

Information Theory

Alistair O'Brien

Department of Computer Science

May 4, 2022

Contents

1 Introduction

1.1 Information and Entropy

- **Motivation**: Need to measure the *information content* of random variables on probability space $(\Omega, \mathcal{F}, P)$.

Definition 1.1.1. (**Shannon Information**) The *Shannon information* of the discrete random variable X on $(\Omega, \mathcal{F}, P)$ is a total function $h : \overrightarrow{X}(\Omega) \to \mathbb{R}$ defined as

$$h(x) = -\log_2 p_X(x)$$

where $h(x)$ is measured in *Shannon bits*.

- Shannon bits $\neq$ encoded bits. Example: Bias coin result with $p_{\text{head}} = 0.25$. Then $h(1) = 2$ bits, but result only requires 1 bit to encode outcome.

Definition 1.1.2. (**Axioms of Information**) Let $h : [0, 1] \to \mathbb{R}$ be the measure of information with a given probability, satisfying the following axioms:

(I) $\forall p \in [0, 1].h(p) \geq 0$.
*Notion of a **negative** number of bits is nonsensical.*

(II) h is monotonically decreasing.
*Intuition of **"surprisal"**. Events w/ high probability = low surprisal $\implies$ low information content, and vice versa.*

(III) $h(1) = 0$.
No information gained if an event is certain.

(IV) $h(p_X \cdot p_Y) = h(p_X) + h(p_Y)$.
*Information is **additive**. Information of 2 independent events is the sum of information from each event.*

Theorem 1.1.1. (**Axiomatic Derviation of Information**) Let $h : [0,1] \to \mathbb{R}$ be a measure of information satisfying (I)–(IV), then I is of the form:

$$h(p) = -k \log p$$

for some $k > 0$.

Proof. Let h be as described. Let us assume that it satisfies (I)-(IV). By (IV) we have:

$$h(p_X \cdot p_Y) = h(p_X) + h(p_Y)$$

Taking derivatives wrt $p_X p_Y$ yields:

$$\begin{aligned} \frac{\partial}{\partial p_X p_Y} h(p_X \cdot p_Y) &= \frac{\partial}{\partial p_X p_Y} h(p_X) + h(p_Y) \\ \iff \frac{\partial}{\partial p_X} h'(p_X \cdot p_Y) \cdot p_X &= \frac{\partial}{\partial p_X} h'(p_Y) \\ \iff h'(p_X \cdot p_Y) + h''(p_X \cdot p_Y) \cdot p_X \cdot p_Y = 0 \end{aligned}$$

Let $p = p_X \cdot p_Y$, so we have the following ODE:

$$h''(p) \cdot p + h'(p) = 0$$

By the inverse product rule, we have

$$\begin{aligned} h'(p) + h''(p) \cdot p = \frac{\mathrm{d}}{\mathrm{d}p} (h'(p) \cdot p) &= 0 \\ \iff h'(p) \cdot p &= k_1 \\ \iff h'(p) &= \frac{k_1}{p} \\ \iff h(p) &= k_1 \log p + k_2 \end{aligned}$$

where k_1, k_2 are constants of integration. By (II) and (III), k_1, k_2 must satisfy

$$\begin{aligned} k_1 \log p + k_2 \geq 0 \\ k_1 \log 1 + k_2 = 0 \end{aligned}$$

giving us

$$k_1 < 0$$
$$k_2 = 0$$

Writing $k_1 = -k$ for some $k > 0$, we have

$$h(p) = -k \log p$$

□

Definition 1.1.3. (**Entropy**) *Entropy* is defined as the expected information content of a discrete random variable X on $(\Omega, \mathcal{F}, P)$:

$$H(X) = \mathbb{E}\left[h(X)\right] = -\sum_{x \in \overrightarrow{X}(\Omega)} p_X(x) \log_2 p_X(x)$$

Definition 1.1.4. (**Joint Entropy**) The entropy of the joint distribution of discrete random variables X, Y on $(\Omega, \mathcal{F}, P)$ is given by:

$$H(X, Y) = \mathbb{E}\left[h(X, Y)\right] = -\sum_{x \in \overrightarrow{X}(\Omega), y \in \overrightarrow{Y}(\Omega)} p_{X,Y}(x, y) \log_2 p_{X,Y}(x, y)$$

Definition 1.1.5. (**Conditional Entropy**) For two discrete random variables X, Y on $(\Omega, \mathcal{F}, P)$, the conditional entropy of X given $Y = y$ is defined as:

$$H(X \mid Y = y) = \mathbb{E}\left[h(X \mid Y = y)\right] = -\sum_{x \in \overrightarrow{X}(\Omega)} p_X(x \mid Y = y) \log_2 p_X(x \mid Y = y)$$

Definition 1.1.6. (**Iterated Conditional Entropy**) The iterated conditional entropy $H(X \mid Y)$, for discrete random variables X, Y on $(\Omega, \mathcal{F}, P)$, is given by:

$$\begin{aligned}
H(X \mid Y) &= \mathbb{E}_Y\left[H(X \mid Y)\right] \\
&= -\sum_{x \in \overrightarrow{X}(\Omega), y \in \overrightarrow{Y}(\Omega)} p_X(x \mid Y = y) p_Y(y) \log_2 p_X(x \mid Y = y) \\
&= -\sum_{x \in \overrightarrow{X}(\Omega), y \in \overrightarrow{Y}(\Omega)} p_{X,Y}(x, y) \log_2 p_X(x \mid Y = y)
\end{aligned}$$

This is the expected uncertainty/information of X given Y, averaged over all *possible values* of X *and* Y.

Theorem 1.1.2. (**Chain Rule of Entropy**) The joint, conditional and marginal entropies of discrete random variables X, Y on $(\Omega, \mathcal{F}, P)$ satisfy

$$H(X,Y) = H(X) + H(Y \mid X) = H(Y) + H(X \mid Y)$$

Proof. Let X, Y be discrete random variables on $(\Omega, \mathcal{F}, P)$. So

$$\begin{aligned}
H(X,Y) &= -\sum_{x \in \overrightarrow{X}(\Omega), y \in \overrightarrow{Y}(\Omega)} p_{X,Y}(x,y) \log_2 p_{X,Y}(x,y) \\
&= -\sum_{x \in \overrightarrow{X}(\Omega), y \in \overrightarrow{Y}(\Omega)} p_{X,Y}(x,y) \log_2 p_X(x) p_Y(y \mid x) \\
&= -\sum_{x \in \overrightarrow{X}(\Omega), y \in \overrightarrow{Y}(\Omega)} p_{X,Y}(x,y) \left[\log_2 p_X(x) + \log_2 p_Y(y \mid x)\right] \\
&= -\sum_{x \in \overrightarrow{X}(\Omega)} \left(\sum_{y \in \overrightarrow{Y}(\Omega)} p_Y(y \mid x)\right) p_X(x) \log_2 p_X(x) \\
&\quad - \sum_{x \in \overrightarrow{X}(\Omega), y \in \overrightarrow{Y}(\Omega)} p_{X,Y}(x,y) \log_2 p_X(x \mid y) \\
&= H(X) + H(Y \mid X)
\end{aligned}$$

Symmetric proof for $H(X,Y) = H(Y) + H(X \mid Y)$. □

Theorem 1.1.3. (**Independence Bound of Entropy**) For the set of discrete random variables $X_1, \ldots, X_n$ on $(\Omega, \mathcal{F}, P)$:

$$H(X_1, \ldots, X_n) \leq \sum_{i=1}^{n} H(X_i)$$

with equality when the random variables $X_1, \ldots, X_n$ are i.i.d.

1.1.1 Principal of Maximal Entropy

- Entropy is *maximized* when all outcomes are *equiprobable*.

Theorem 1.1.4. Let X be a discrete random variable on $(\Omega, \mathcal{F}, P)$. The entropy $H(X)$ satisfies:

$$H(X) \leq \log_2 |\overrightarrow{X}(\Omega)|$$

Proof. Let X be as described.
Proof Idea:

1. Formalize statement as an optimization problem.
2. Use Lagrangian multipliers to find the optimal solution.

Wlog. $\mathcal{X} = \{1, \dots, n\}$ and $p_i = p_X(i)$. We wish to maximize $H(X)$ (varying $\mathbf{p}$) subject to the constraint $\sum_{i=1}^n p_i = 1$. We now solve the optimization problem using Lagrange Multipliers. We have the following Lagrangian:

$$\mathcal{L}(p_1, \dots, p_n, \lambda) = -\sum_{i=1}^n p_i \log_2 p_i + \lambda \left(\sum_{i=1}^n p_i - 1 \right)$$

Computing the partial derivation wrt p_i and equating to 0 yields:

$$\begin{aligned}
\frac{\partial}{\partial p_i} - \sum_{j=1}^n p_j \log_2 p_j + \lambda \left(\sum_{j=1}^n p_j - 1 \right) = 0 \\
\iff -\log_2 p_i - \frac{p_i}{p_i \ln 2} - \lambda = 0 \\
\iff p_i = 2^{-(\lambda + 1/\ln 2)}
\end{aligned}$$

Hence p_i is *constant*. Given that $\sum_{i=1}^n p_i = 1$, we deduce that $p_i = 1/|\mathcal{X}|$. Substituting p_i into $H(X)$ yields

$$\begin{aligned}
H(X) &= -\sum_{i=1}^n \frac{1}{|\mathcal{X}|} \log_2 |\mathcal{X}| \\
&= \log_2 |\mathcal{X}|
\end{aligned}$$

So we conclude that $H(X) \leq \log_2 |\mathcal{X}|$, with equality when $p_X(x) = 1/|\mathcal{X}|$ (when X is uniformly distributed). $\square$

- This theorem is key for many optimization problems: maximal information/entropy gained $\implies$ best algorithm. See Coding Problems.

1.1.2 Mutual Information

- **Motivation**: Measure information that one variable contains about another – useful for inference.

Definition 1.1.7. (**Relative Entropy**) The relative entropy between two distributions p_X and q_X for the discrete random variable X on $(\Omega, \mathcal{F}, P)$ is

$$D(p_X \parallel q_X) = \sum_{x \in \overrightarrow{X}(\Omega)} p_X(x) \log_2 \frac{p_X(x)}{q_X(x)} = \mathbb{E}_{p_X}\left[\log_2 \frac{p_X(X)}{q_X(X)}\right]$$

Theorem 1.1.5. (**Properties of Rel. Entropy**) Relative entropy satisfies the following properties:

(i) $D(p_X \parallel q_X) \geq 0$ for all discrete distributions p_X, q_X.

(ii) $D(p_X \parallel p_X) = 0$.

- *Intuitively*, $D(p_X \parallel q_X)$ quantifies how 'close' q_X is to p_X.
  It is **not** a distance metric (not symmetric, nor does it satisfy the triangle eq.).

Definition 1.1.8. (**Mutual Information**) The mutual information of discrete random variable X, Y on $(\Omega, \mathcal{F}, P)$ is defined as the relative entropy between their joint distribution and the product of their marginal distributions:

$$\begin{aligned} I(X;Y) &= D\left(p_{X,Y} \parallel p_X \cdot p_Y\right) \\ &= \sum_{x \in \overrightarrow{X}(\Omega), y \in \overrightarrow{Y}(\Omega)} p_{X,Y}(x,y) \log_2 \frac{p_{X,Y}(x,y)}{p_X(x)p_Y(y)} \end{aligned}$$

Theorem 1.1.6. The mutual information and marginal and conditional entropies of discrete random variables X, Y on $(\Omega, \mathcal{F}, P)$ satisfies

$$I(X;Y) = H(X) - H(X \mid Y) = H(Y) - H(Y \mid X)$$

Proof. Let X, Y be discrete random variables on $(\Omega, \mathcal{F}, P)$. So

$$\begin{aligned} I(X;Y) &= \sum_{x \in \overrightarrow{X}(\Omega), y \in \overrightarrow{Y}(\Omega)} p_{X,Y}(x,y) \log_2 \frac{p_{X,Y}(x,y)}{p_X(x)p_Y(y)} \\ &= \sum_{x \in \overrightarrow{X}(\Omega), y \in \overrightarrow{Y}(\Omega)} p_{X,Y}(x,y) \log_2 \frac{p_Y(x \mid y)}{p_X(x)} \\ &= \sum_{x \in \overrightarrow{X}(\Omega), y \in \overrightarrow{Y}(\Omega)} p_{X,Y}(x,y) \log_2 p_Y(x \mid y) - \sum_{x \in \overrightarrow{X}(\Omega), y \in \overrightarrow{Y}(\Omega)} p_{X,Y}(x,y) \log_2 p_X(x) \\ &= H(X) - H(X \mid Y) \end{aligned}$$

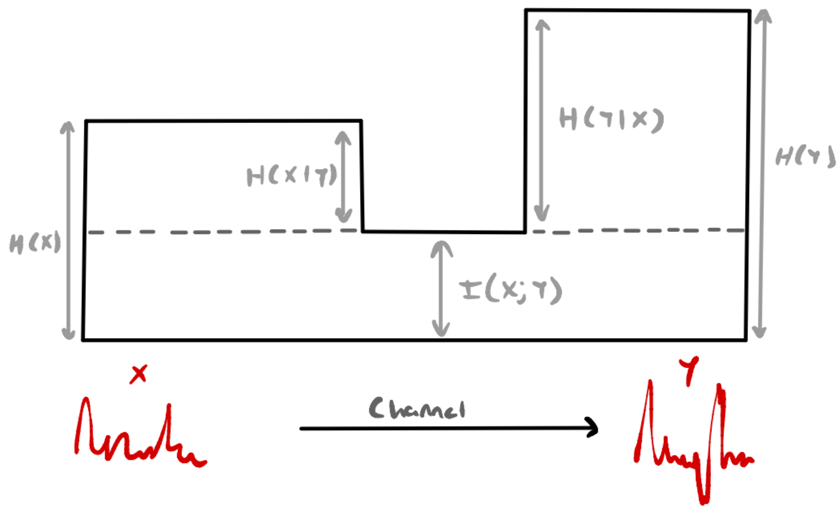


Figure 1.1: Mutual Information Visualization

Symmetric proof for $I(X;Y) = H(Y) - H(Y \mid X)$. $\square$

Corollary 1.1.6.1. $I(X;Y) = H(X) + H(Y) - H(X,Y)$

Definition 1.1.9. (**Conditional Mutual Information**) The conditional mutual information between discrete random variables X, Y, Z on $(\Omega, \mathcal{F}, P)$ is

$$\begin{aligned} I(X;Y \mid Z) &= \mathbb{E}\left[\log_2 \frac{p_{X,Y}(X,Y \mid Z)}{p_X(X \mid Z)p_Y(Y \mid Z)}\right] \\ &= H(X \mid Z) - H(X \mid Y, Z) \end{aligned}$$

Theorem 1.1.7. (**Properties of Mutual Information**) Mutual entropy satisfies:

(i) $I(X;Y) \geq 0$

(ii) Chain rule: $I(X,Y;Z) = I(X;Z) + I(Y;Z \mid X)$

1.2 Continuous Information Measures

- **Idea**: Extend information measures for continuous random variables, required for signal processing + noisy channels.
- **Problem**: Entropy doesn't extend to continuous random variables. Considering the discretization of the random variable $X \sim f_X$ into X_Δ with period Δx is given by:

$$\begin{aligned}
p_i &= \int_{i\Delta x - \Delta x/2}^{i\Delta x + \Delta x/2} f_X(x)\,\mathrm{d}x \approx f(i\Delta x)\Delta x \\
H(X_\Delta) &= -\sum_i p_i \log_2 p_i \\
&\approx -\sum_i f_X(i\Delta x)\Delta x \log_2 f_X(i\Delta x)\Delta x \\
&= -\sum_i f_X(i\Delta x)\Delta x \log_2 f_X(i\Delta x) - \underbrace{\left(\sum_i f_X(i\Delta x)\Delta x\right)}_{1} \log_2 \Delta x \\
&= -\sum_i f_X(i\Delta x)\Delta x \log_2 f_X(i\Delta x) - \log_2 \Delta x
\end{aligned}$$

Considering the limit of $\Delta x \to 0$ yields

$$H(X_\Delta) = -\int_{x \in \overrightarrow{X}(\Omega)} f_X(x) \log_2 f_X(x)\,\mathrm{d}x - \underbrace{\lim_{\Delta x \to 0} \log_2 \Delta x}_{\to \infty}$$

RHS is undefined!

Definition 1.2.1. (**Differential Entropy**) The differential entropy of the continuous random variable X on $(\Omega, \mathcal{F}, P)$ is defined as:

$$\mathrm{d}H(X) = \mathbb{E}\left[-\log_2 f_X(X)\right] = -\int_{x \in \overrightarrow{X}(\Omega)} f_X(x) \log_2 f_X(x)\,\mathrm{d}x$$

- Hence $H(X_\Delta) = \mathrm{d}H(X) - \lim_{\Delta x \to 0} \log_2 \Delta x$.

- *Differential entropy* has no physical meaning (as opposed to discrete entropy), but may be used to compute differences between discretized continuous entropies:

$$\begin{aligned} H(X_\Delta) - H(Y_\Delta) &= \mathrm{d}H(X) - \lim_{\Delta x \to 0} \log_2 \Delta x - (\mathrm{d}H(Y) - \lim_{\Delta y \to 0} \log_2 \Delta y) \\ &= \mathrm{d}H(X) - \mathrm{d}H(Y) \end{aligned}$$

- Differences between entropies and differential entropies:
  (i) $\forall k \in \mathbb{R}.\, \mathrm{d}H(X + k) = \mathrm{d}H(X)$
  (ii) $\forall k \in \mathbb{R}.\, \mathrm{d}H(kX) = \mathrm{d}H(X) + \log_2 k$, for $k \neq 0$.

Definition 1.2.2. (**Relative Entropy**) The *relative entropy* between two continuous distributions f_X and g_X for the continuous random variable X on $(\Omega, \mathcal{F}, P)$ is

$$D(f_X \parallel g_X) = \int_{x \in \overrightarrow{X}(\Omega)} f_X(x) \log_2 \frac{f_X(x)}{g_X(x)} \,\mathrm{d}x = \mathbb{E}_{f_X}\left[\log_2 \frac{f_X(X)}{g_X(X)}\right]$$

- When the integral is undefined, $D(f_X \parallel g_X) = \infty$ by convention.

Definition 1.2.3. (**Mutual Information**) Mutual information for two continuous random variables X, Y is anlogous to the discrete definition:

$$\begin{aligned} I(X;Y) &= D(f_{X,Y} \parallel f_X \times f_Y) \\ &= \iint_{x,y \in \overrightarrow{X}(\Omega) \times \overrightarrow{Y}(\Omega)} f_{X,Y}(x,y) \log_2 \frac{f_{X,Y}(x,y)}{f_X(x) f_Y(y)} \,\mathrm{d}x\,\mathrm{d}y \end{aligned}$$

1.3 Distances

- **Idea**: Relative entropy is the *entropic 'distance'* between two distributions.
- **Problem**: It doesn't satisfy axioms of distance!

Definition 1.3.1. (**Entropic Distance**) The entropic distance between two random variables X, Y on $(\Omega, \mathcal{F}, P)$ is:

$$D(X,Y) = H(X,Y) - I(X;Y)$$

Lemma 1.3.1. (**Properties of Entropic Distance**) Distance satisfies the following properties:

(i) $D(X, Y) \geq 0$

(ii) $D(X, X) = 0$

(iii) $D(X, Y) = D(Y, X)$

(iv) $D(X, Z) \leq D(X, Y) + D(Y, Z)$

1.3.1 Connection to Machine Learning

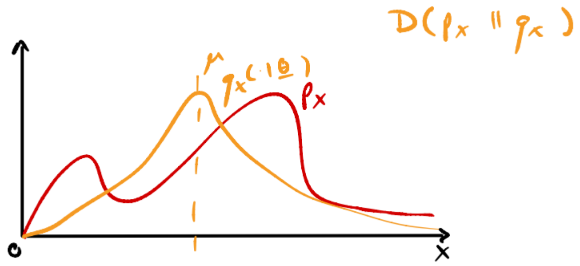
- **Idea**: Relative entropy is the **cost** incurred if q_X is used to encode X when p_X is the *true* distribution.

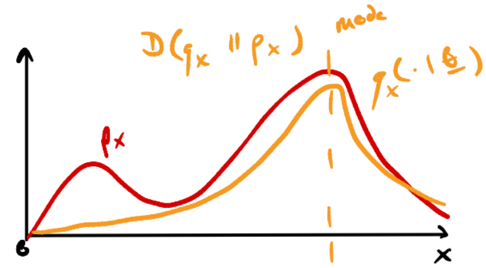
- Suppose we wish to fit a model $q_X(\cdot \mid \boldsymbol{\theta})$ to the distribution p_X minimizing the cost $D(p_X \parallel q_X(\cdot \mid \boldsymbol{\theta}))$:

$$
\begin{aligned}
\hat{\boldsymbol{\theta}} &= \arg\min_{\boldsymbol{\theta}} D(p_X \parallel q_X(\cdot \mid \boldsymbol{\theta})) \\
&= \arg\min_{\boldsymbol{\theta}} \sum_{x \in \overrightarrow{X}(\Omega)} p_X(x) \log_2 \frac{p_X(x)}{q_X(x \mid \boldsymbol{\theta})} \\
&= \arg\min_{\boldsymbol{\theta}} H(X) - \sum_{x \in \overrightarrow{X}(\Omega)} p_X(x) \log_2 q_X(x \mid \boldsymbol{\theta}) \\
&= \arg\max_{\boldsymbol{\theta}} \sum_{x \in \overrightarrow{X}(\Omega)} p_X(x) \log_2 q_X(x \mid \boldsymbol{\theta}) \\
&= \arg\max_{\boldsymbol{\theta}} \mathbb{E}_{p_X} \left[\log_2 q_X(X \mid \boldsymbol{\theta})\right] \\
&= \arg\max_{\boldsymbol{\theta}} \lim_{n \to \infty} \frac{1}{n} \underbrace{\sum_{i=1}^{n} \log_2 q_X(x_i \mid \boldsymbol{\theta})}_{MLE}
\end{aligned}
$$

Hence minimizing relative entropy *is* MLE!

- Similar relations exist for reinforcement learning (on the right):





Definition 1.3.2. (**Cross Entropy**) The cross-entropy between the distributions p_X, q_X for X on $(\Omega, \mathcal{F}, P)$ is defined as:

$$H(p_X, q_X) = -\sum_{x \in \overrightarrow{X}(\Omega)} p_X(x) \log_2 q_X(x)$$

- Minimizing the cross entropy is also equivalent to MLE (see above).

1.3.2 Information and Correlation

- **Motivation**: Mutual information and the correlation coefficient both numerically encode a relationship between random variables X, Y.

Definition 1.3.3. (**Correlation Coefficient**) For random variables X, Y on $(\Omega, \mathcal{F}, P)$, the correlation coefficient is defined by

$$\rho(X, Y) = \frac{\text{Cov}[X, Y]}{\sqrt{\text{Var}[X] \cdot \text{Var}[Y]}}$$

Lemma 1.3.2. (**Properties of Correlation and Mutual Information**) The correlation coefficient $\rho(X, Y)$ and mutual information $I(X; Y)$ satisfy the following properties:

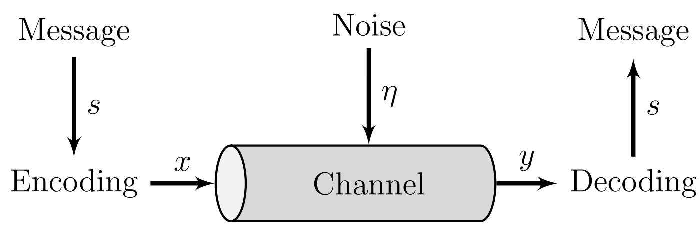
(i) $\rho(X, Y) \neq 0 \implies I(X; Y) > 0$. Correlation implies shared information.

(ii) $\rho(X, Y) = 0 \;\not\Longrightarrow\; I(X; Y) = 0$. No correlation *doesn't necessarily* imply no shared information – since $\rho(X, Y)$ attempts to fit a *linear* relation between random variables (relationship may be non-linear).

2 Coding Problems

- **Idea**: Reducing size of message sent over a *channel* while maximizing information content, this is known as the *coding problem*.
- **Notation**: $\mathcal{X} = \overrightarrow{X}(\Omega)$.

Definition 2.0.1. (**Communication Channel**) A *communication* channel in medium in which a message is encoded before being sent over the channel, potentially adding *noise*. The channel output is decoded, to recover the message:



- **Most** problems in information theory are instantiations of a communication channel problem.

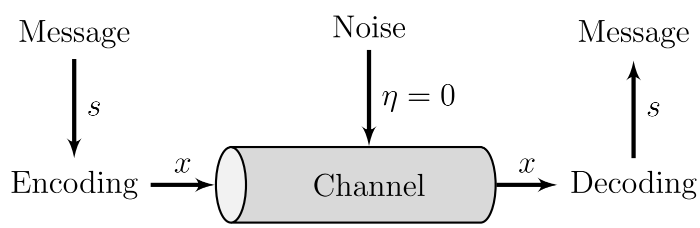
Definition 2.0.2. (**Codes**) A *code* $\mathscr{C}$ with respect to discrete random variable X on $(\Omega, \mathcal{F}, P)$, is a function $C : \mathcal{X} \to \Sigma^*$, where Σ is a finite alphabet.

- We write $\mathscr{C}(x)$ for the codeword of x. We write $l(x) = |C(x)|$.
- We often assume $\Sigma = \{0, 1\}$.
- We extend $\mathscr{C}$ to $\mathscr{C}^+ : \mathcal{X}^+ \to \Sigma^*$, defined by:

$$\mathscr{C}^+(x_1 x_2 \dots x_n) = \mathscr{C}(x_1)\mathscr{C}(x_2)\dots\mathscr{C}(x_n)$$

- Codewords of C is $\mathcal{C} = \overrightarrow{C}(\mathcal{X})$

Definition 2.0.3. (**Coding Problem**) The *coding problem* is defined as the problem of finding a code $\mathscr{C}$ that minimizes codeword length $\mathbb{E}[l(X)]$ (transmitted) via a noiseless channel:



- 2 approaches to the coding problem:

  Lossless Fully recover the message s

  Lossy Cannot fully recover s – formally, due to collisions in encoding with probability δ. If δ is sufficiently small $\implies$ compressor (or coding) is *practical*.

2.1 Shannon's Source Coding Theorem

- **Idea**: Shannon's Source Coding Theorem focuses on theoretical limit of lossy compression with *fixed length encodings*.

2.1.1 Block Codes

- **Motivation**: Encoding of blocks symbols for *fixed length encodings*.

Definition 2.1.1. (**Block**) For a discrete random variable X on $(\Omega, \mathcal{F}, P)$, a block of n, denoted X^n is defined as:

$$X^n = (X_1, \dots, X_n)$$

where $(X_i)_{1 \leq i \leq n}$ are i.i.d random variables distributed by p_X.

- By **additivity**, $H(X^n) = nH(X)$.

Definition 2.1.2. (**Block Code**) A n-block code $\mathscr{C}^n$ wrt. to the block X^n on $(\Omega, \mathcal{F}, P)$ is a function $\mathscr{C}^n : \mathcal{X}^n \to \Sigma^*$.

- Block codes are a formalization for fixed length encodings.
- We can characterise effeciency of a n-block code $\mathscr{C}^n$ via the *expected per-symbol codeword length*:

$$\mathbb{E}\left[\frac{1}{n}l(X^n)\right] = \frac{1}{n}\mathbb{E}\left[l(X^n)\right]$$

2.1.2 Lossy Codes

- **Motivation**: Characterize lossy compression with lossy codes for a given *probability of error* ϵ.

Definition 2.1.3. (**Lossy Code**) A ϵ-lossy code $\mathscr{C}_\epsilon$ wrt. the discrete random variable X on $(\Omega, \mathcal{F}, P)$ is a code $\mathscr{C}_\epsilon : \mathcal{X} \to \Sigma^*$ with a *probability of error* ϵ satisfying:

$$\epsilon \geq P(\mathscr{C}(X) \neq \mathscr{C}^{-1}(X))$$

- Write $p_e(\mathscr{C}) = P(\mathscr{C}(X) \neq \mathscr{C}^{-1}(X))$.
- Course touches on *smallest* ϵ-sufficient sets (not required for our proof).

Definition 2.1.4. (**Smallest ϵ-sufficient Set**) For the discrete random variable X on $(\Omega, \mathcal{F}, P)$, we define the *smallest ϵ-sufficient set* $\mathcal{X}_\epsilon \subseteq \mathcal{X}$ s.t:

$$P(x \in \mathcal{X}_\epsilon) \geq 1 - \epsilon$$

- Algorithm for computing $\mathcal{X}_\epsilon$:

```
let smallest_sufficient_set X ϵ =
  𝒳 ← List.sort 𝒳 ~compare:(reverse order induced by p_X);
  𝒳_ϵ ← [];
  while 𝒳_ϵ |> List.map ~f:p_X |> List.sum < 1 - ϵ do
    𝒳_ϵ ← List.pop 𝒳 :: 𝒳_ϵ
  done;
  𝒳_ϵ
```

- The maximum entropy of $\mathcal{X}_\epsilon$ is $H_\epsilon(X) = \log_2 |\mathcal{X}_\epsilon|$.

2.1.3 Typical Sets

- **Motivation**: Consider a *typical* (expected) set of blocks and its asymptotic properties.

Definition 2.1.5. (**Typical String**) A *typical* string $\mathbf{x} \in \mathcal{X}^n$ satisfies:

$$\forall x_i \in \mathcal{X}. \sum_{x \in \mathbf{x}} I_{x=x_i} = \mathbb{E}\left[\sum_{x \in \mathbf{x}} I_{x=x_i}\right] = p_X(x_i)n$$

- A typical string contains the *expected number of each symbol*
- The probability of a typical string $\mathbf{x}$ is:

$$p(\mathbf{x}) = \prod_{x_i \in \mathcal{X}} p_X(x_i)^{p_X(x_i)n}$$

Hence the *information* of typical string:

$$\begin{aligned} h(\mathbf{x}) &= -\log_2 \prod_{x_i \in \mathcal{X}} p_X(x_i)^{p_X(x_i)n} \\ &= -\sum_{x_i \in \mathcal{X}} \log_2 p_X(x_i)^{p_X(x_i)n} \\ &= -n \sum_{x_i \in \mathcal{X}} p_X(x_i) \log_2 p_X(x_i) \\ &= nH(X) \end{aligned}$$

Hence $p(\mathbf{x}) = 2^{-nH(X)}$.

Definition 2.1.6. (**Typical Set**) A *typical set* $A_\epsilon^n(X)$ with respect to the discrete random variable X is the set of strings $\mathbf{x} \in \mathcal{X}^n$ s.t

$$2^{-n(H(X)+\epsilon)} < p(\mathbf{x}) < 2^{-n(H(X)-\epsilon)}$$

We write $A_\epsilon^n(X)$ as an ϵ-typical set wrt. to X, we have,

$$A_\epsilon^N = \left\{\mathbf{x} \in \mathcal{X}^n : \left|\frac{1}{N}h(\mathbf{x}) - H(X)\right| < \epsilon\right\}$$

Theorem 2.1.1. (**Asymptotic equipartition property**) If $X_1, \ldots, X_n \overset{\text{iid}}{\sim} p_X$, then

$$\forall \epsilon > 0. \lim_{n \to \infty} P\left(\left|\frac{1}{N} h(X^n) - H(X)\right| < \epsilon\right) = 1$$

Proof. Let $\epsilon > 0$ be arbitrary. Recall that the WLL states that

$$\lim_{n \to \infty} P\left(\left|\overline{X_n} - \mu\right| < \epsilon\right) = 1$$

Instating for the random variable $h(X)$ yields

$$\begin{aligned}
\mu &= H(X) \\
\overline{X_n} &= \frac{1}{n} \sum_{i=1}^{N} h(X_i) \\
&= -\frac{1}{n} \sum_{i=1}^{N} \log_2 p_X(X_i) \\
&= -\frac{1}{n} \log_2 \prod_{i=1}^{n} p_X(X_i) \\
&= \frac{1}{n} h(X^n)
\end{aligned}$$

So we are done. □

Lemma 2.1.1. (**Properties of $A_\epsilon^n(X)$**)

- For sufficiently large n,

$$P\left(X^n \in A_\epsilon^n(X)\right) \geq 1 - \epsilon$$

- For sufficiently large n,

$$(1 - \epsilon) 2^{n(H(X) - \epsilon)} < |A_\epsilon^n(X)| < 2^{n(H(X) + \epsilon)}$$

2.1.4 Source Coding Theorem

Theorem 2.1.2. (**Shannon's Source Coding Theorem**) Shannon's source coding theorem states that for a discrete random variable X on $(\Omega, \mathcal{F}, P)$, for all $0 \leq \delta \leq 1$, there exists a δ-lossy block code $\mathscr{C}_\delta^n$ s.t

$$\lim_{n \to \infty} \frac{1}{n} \mathbb{E}[l(X^n)] = H(X)$$

Proof. Let X, δ be as described. By the $\epsilon - \delta$ def. of a limit, we wish to show that

$$\forall \epsilon > 0. \exists n_0. \forall n > n_0. \left| \frac{1}{n} \mathbb{E}[l(X^n)] - H(X) \right| < \epsilon$$

for some δ-lossy block code $\mathscr{C}_\delta^n$.

Let $\epsilon > 0$ be arbitrary. We define n_0 s.t $n_0 > (\epsilon - \delta/2)^{-1}$. Let $n > n_0$ be arbitrary. Let us define the δ-lossy n-block code $\mathscr{C}_\delta^n : \mathcal{X}^n \to \{0,1\}^*$ as

$$\mathscr{C}_\delta^n(\mathbf{x}) = \begin{cases} \text{encoding of } \mathbf{x} & \text{if } \mathbf{x} \in A_{\delta/2}^n(X) \\ 1 & \text{otherwise} \end{cases}$$

where $|\mathcal{C}| = |A_{\delta/2}^n(X)| + 1$. Thus the expected codeword length is

$$\mathbb{E}[l(X^n)] = \log_2 |\mathcal{C}|$$

We verify $\mathscr{C}_\delta^n$ is δ-lossy. By Lemma 2.1.1, we have

$$p_e(\mathscr{C}_\delta^n) = P(X^n \notin A_{\delta/2}^n(X)) < \frac{\delta}{2} \leq \delta$$

By AEP, we have

$$\begin{aligned} |A_{\delta/2}^n(X)| + 1 &< 2^{n(H(X)+\delta/2)} + 1 \\ &\leq 2 \times 2^{n(H(X)+\delta/2)} && n(H(X)+\delta/2) \geq 0 \\ &< 2^{n(H(X)+\epsilon)} && n\epsilon > 1 + n\delta/2 \end{aligned}$$

Hence

$$\begin{aligned} && \mathbb{E}[l(X^n)] &= \log_2(|A_{\delta/2}^n(X)| + 1) \\ && &< n(H(X)+\epsilon) \\ \iff && \frac{1}{n}\mathbb{E}[l(X^n)] - H(X) &< \epsilon \end{aligned}$$

We now show that $-\frac{1}{n}\mathbb{E}[l(X^n)] + H(X) < \epsilon$. By AEP, we have

$$\begin{aligned} |A_{\delta/2}^n(X)| + 1 &> \left(1 - \frac{\delta}{2}\right) 2^{n(H(X)-\delta/2)} + 1 \\ &> \frac{1}{2} 2^{n(H(X)-\delta/2)} && 0 \leq \delta/2 \leq 1/2 \\ &= 2^{n(H(X)-\delta/2)-1} \\ &> 2^{n(H(X)-\epsilon)} && n\epsilon > 1 + n\delta/2 \end{aligned}$$

Hence

$$\begin{aligned} \mathbb{E}[l(X^n)] &= \log_2(|A^n_{\delta/2}(X)| + 1) \\ &> n(H(X) - \epsilon) \\ \iff -\frac{1}{n}\mathbb{E}[l(X^n)] + H(X) &< \epsilon \end{aligned}$$

Completing the proof.

□

- **Intuitition**: $\mathcal{X}^n_\delta$ contains all probability of sequences in $\mathcal{X}^n$ up to an error δ, and typical set $A^n_\epsilon(X)$ contains most of the probability of the sequences in X^n, hence
$$|\mathcal{X}^n_\delta| \approx |A^n_\epsilon(X)| \approx 2^{nH(X)}$$

2.2 Symbol Codes

- **Motivation**: Symbol codes formalize the theoretical limits of lossless compression for *variable length codes*.

Definition 2.2.1. (**Characterization of Variable-Length Codes**) A variable-length (symbol) code $\mathscr{C}$ has the following properties:

Non-Singular Codes $\forall x_1, x_2 \in \mathcal{X}. x_1 \neq x_2 \implies \mathscr{C}(x_1) \neq \mathscr{C}(x_2)$. This property is necessary for lossless and perfectly decodable encodings.

Unique Decodability A code is uniquely decodable if $\mathscr{C}^+$ is non-singular.

2.2.1 Prefix Codes

- **Motivation**: Additional desirable properties for codes include *'easy' decodability* $\implies$ **prefix codes**

Definition 2.2.2. (**Prefix Codes**) A *symbol code* $\mathscr{C}$ is said to be a *prefix code* if the following holds:

$$\forall x_1, x_2 \in \mathcal{X}. x_1 \neq x_2 \implies \mathscr{C}(x_1) \neq \mathsf{prefix}(\mathscr{C}(x_2))$$

where a *prefix* of a string $u \in \Sigma^*$ is a string v s.t

$$\exists w \in \Sigma^*. u = vw$$

- Prefix codes can easily be decoded by traversing a *prefix tree*.
- Prefix codes are *uniquely decodable*!

Theorem 2.2.1. (**Kraft's Inequality**) For a n-ary uniquely decodable code $\mathscr{C}$ wrt. X on $(\Omega, \mathcal{F}, P)$,

$$\sum_{x \in \overrightarrow{X}(\Omega)} \frac{1}{n^{l(x)}} \leq 1$$

Proof. Let X be an arbitrary discrete random variable on $(\Omega, \mathcal{F}, P)$. Let $\mathscr{C}$ be a n-ary ($|\Sigma| = n$) uniquely decodable code. Let us define $S = \sum_{x \in \overrightarrow{X}(\Omega)} n^{-l(x)}$.
Proof Idea:

1. Find upper bound for S^m for all $m \in \mathbb{N}$.
2. Show upper bound only holds if $S \leq 1$

Let $m \in \mathbb{N}$ be arbitrary. We have

$$\begin{aligned} S^m &= \left[\sum_{x \in \overrightarrow{X}(\Omega)} n^{-l(x)} \right]^m \\ &= \sum_{x_1 \in \overrightarrow{X}(\Omega)} \sum_{x_2 \in \overrightarrow{X}(\Omega)} \cdots \sum_{x_m \in \overrightarrow{X}(\Omega)} n^{-\sum_{i=1}^m l(x_i)} \end{aligned}$$

We note that $l(\mathbf{x}) = \sum_{i=1}^m l(x_i)$ for $\mathbf{x} = (x_1, \ldots, x_m) \implies$ each string $\mathbf{x}$ of length $l(\mathbf{x})$ constributes $n^{-l(\mathbf{x})}$ to the sum. So we re-write the summation as:

$$S^m = \sum_{l=1}^{m \cdot l_{\max}} q_l n^{-l}$$

where q_l is the number of codewords with length l. Since $\mathscr{C}$ is uniquely decodable $\implies q_l \leq n^l$. Hence

$$S^m = \sum_{l=1}^{m \cdot l_{\max}} q_l n^{-l} \leq \sum_{l=1}^{m \cdot l_{\max}} 1 = m l_{\max}$$

As a result, we have $\sum_{x\in\overrightarrow{X}(\Omega)} n^{-l(x)} \leq (ml_{\max})^{1/m}$ for any $m \in \mathbb{N}$. Since the lhs doesn't depend on m, the inequality holds in the limit $m \to \infty$, and since

$$\lim_{m\to\infty} (ml_{\max})^{1/m} = 1,$$

we conclude that,

$$\sum_{x\in\overrightarrow{X}(\Omega)} n^{-l(x)} \leq 1.$$

□

- **Cases**:
  - If $<$ $\implies$ redundancy in the code
  - If $=$ $\implies$ the code C is *complete* (often achieved w/ prefix codes with no empty leaves)

Lemma 2.2.1. For a code $\mathscr{C}$ with codeword lengths $(l_i)_{i\geq 1}$, there is a prefix code P with equal codeword lengths, if and only if:

$$\sum_{i=1}^{m} n^{-l_i} \leq 1$$

Proof. Without loss of generality, we have:

$$\begin{array}{ccccccc} x_1 & & x_2 & & \cdots & & x_m \\ l_1 & \leq & l_2 & \leq & \cdots & \leq & l_m \end{array}$$

Proof Idea:

1. Find a constraint on whether prefix code exists
2. Show equivalence to Kraft's inequality

A prefix code $\mathscr{P}$ must satify for all $1 \leq i \leq m$ codeword $\mathscr{P}(x_i)$ for x_i is not a prefix of any codewords $\mathscr{P}(x_j)$, for all $1 \leq j < i$. The set of 'ruled-out' (or forbidden) codewords is given by:

$$\begin{aligned} \mathcal{F}_1 &= \emptyset \\ \mathcal{F}_{i+1} &= \{\mathscr{P}(x_i)u \in \Sigma^* : u \in \Sigma^*, l_i + |u| = l_{i+1}\} \\ &\quad \cup \{cu \in \Sigma^* : u \in \Sigma^*, c \in \mathcal{F}_i, |c| + |u| = l_{i+1}\} \end{aligned}$$

Thus we have the following recursion relation:

$$|\mathcal{F}_1| = 0$$
$$|\mathcal{F}_{i+1}| = (|\mathcal{F}_i| + 1)n^{l_{i+1}-l_i}$$

A prefix code exists iff the number of possible prefix codewords > number of forbidden codewards, that is:

$$\forall 1 \leq i \leq m. \quad n^{l_i} > |\mathcal{F}_i| = \sum_{j=1}^{i-1} n^{l_i - l_j}$$

We have

$$\begin{aligned} & \sum_{j=1}^{i-1} n^{l_i - l_j} < n^{l_i} \\ \iff & 1 + \sum_{j=1}^{i-1} n^{l_i - l_j} \leq n^{l_i} \\ \iff & \sum_{j=1}^{i} n^{l_i - l_j} \leq n^{l_i} \\ \iff & \sum_{j=1}^{i} n^{-l_j} \leq 1 \end{aligned}$$

So we are done.

$\square$

- **Remark**: The above Lemma allows to work with prefix codes under the assumption of unique decodability, due to Kraft's equality.

2.2.2 Source Coding Theorem for Symbol Codes

- **Motivation**: Consider theoretical limit of expected codeword length (compressed size)

Lemma 2.2.2. (**Source Coding Theorem Part I**) For a discrete random variable X on $(\Omega, \mathcal{F}, P)$ and uniquely decodable code $\mathscr{C} : \mathcal{X} \to \Sigma^*$,

$$\mathbb{E}\left[l(X)\right] \geq H(X)$$

Proof. Let X be an arbitrary discrete random variable on $(\Omega, \mathcal{F}, P)$. This is an optimization problem *subject to* Kraft's inequality:

$$\min_{\text{u.d } C:\overrightarrow{X}(\Omega)\to\Sigma^*} \sum_{x\in\overrightarrow{X}(\Omega)} l(x)p_X(x)$$
$$\text{subject to} \sum_{x\in\overrightarrow{X}(\Omega)} |\Sigma|^{-l(x)} \leq 1$$

Proof Idea:

1. Relax the optimization problem to use Lagrange Multipliers.
2. Solve.

We write $l_i = l(x_i)$ and $p_i = p_X(x_i)$. Thus the problem is:

$$\min_{(l_i)\in\mathbb{N}} \sum_i l_i p_i \quad \text{subject to} \quad \sum_i |\Sigma|^{-l_i} \leq 1$$

Given we're interested in a *lower bound*, we relax our feasible region from $\mathbb{N}$ to $\mathbb{R}$. We now assert the following (both proved by contradictions):

- Kraft's inequality $\implies l_i > 0$.
- Optimiality is only achieved when $\sum_i |\Sigma|^{-l_i} = 1$.

As a result, our optimization problem is now given by:

$$\min_{(l_i)\in\mathbb{R}} \sum_i l_i p_i \quad \text{subject to} \quad \sum_i |\Sigma|^{-l_i} = 1$$

We now change variables, resulting in a simpler problem definition. Let us define $q_i = |\Sigma|^{-l_i}$, so we have $l_i = -\log_{|\Sigma|} q_i$. Giving the following optimization problem:

$$\min_{(q_i)\in\mathbb{R}} -\sum_i p_i \log_{|\Sigma|} q_i \quad \text{subject to} \quad \sum_i q_i = 1$$

We now solve this optimization problem using Lagrange Multipliers. To do so, we form the Lagrangian:

$$\mathcal{L}(q_1, \ldots, q_m, \lambda) = -\sum_{i=1}^{m} p_i \log_{|\Sigma|} q_i + \lambda \left(\sum_{i=1}^{m} q_i - 1 \right)$$

Computing the partial derivatives wrt to q_i and equating to 0 yields:

$$\begin{aligned}
\frac{\partial}{\partial q_i} - \sum_{j=1}^{m} p_j \log_{|\Sigma|} q_j + \lambda \left(\sum_{j=1}^{m} q_j - 1 \right) &= 0 \\
\iff -\frac{p_i}{q_i \ln |\Sigma|} + \lambda &= 0 \\
\iff q_i &= \frac{p_i}{\lambda \ln |\Sigma|}
\end{aligned}$$

Substituting q_i into the constraint $\sum_{i=1}^{m} q_i = 1$ gives us:

$$\lambda = \frac{1}{\ln |\Sigma|}$$

Hence $q_i = p_i$. Thus $\mathbb{E}[l(X)] = -\sum_i p_i \log_{|\Sigma|} p_i = H(X)$ (in $|\Sigma|$-shannon bits). ☐

- **Remarks**: $l_i = -\log_{|\Sigma|} p_i$ may be an optimal codeword length, but its not neccessarily a *feasible* length (e.g. could be fractional).

Lemma 2.2.3. (**Source Coding Theorem Part II**) For an arbitrary discrete random variable X on $(\Omega, \mathcal{F}, P)$, there exists a prefix code $\mathscr{C}$ s.t

$$\mathbb{E}[l(X)] < H(X) + 1$$

Proof. Let X be an arbitrary discrete random variable on $(\Omega, \mathcal{F}, P)$.
Proof Idea:

- Determine lengths (l_i) that satisfy inequality.
- Show (l_i) satisfy Kraft's inequality $\implies$ existence of prefix code with specified lengths.

Let us define $(l_i)_{1 \leq i \leq m}$ for $m = |\mathcal{X}|$, as

$$l_i = \lceil -\log_2 p_i \rceil$$

We have

$$\begin{aligned}
\mathbb{E}[l(X)] &= \sum_{i=1}^{m} p_i \lceil -\log_2 p_i \rceil \\
&< \sum_{i=1}^{m} p_i \left(-\log_2 p_i + 1\right) \\
&= -\sum_{i} p_i \log_2 p_i + 1 \\
&= H(X) + 1
\end{aligned}$$

We now show there exists a prefix code with lengths $(l_i)_{1\leq i\leq m}$. By Lemma 2.2.1, it is sufficient to show that

$$\sum_{i=1}^{m} 2^{-l_i} \leq 1$$

We have

$$\begin{aligned}
\sum_{i=1}^{m} 2^{-l_i} &= \sum_{i=1}^{m} 2^{-\lceil -\log_2 p_i \rceil} \\
&\leq \sum_{i=1}^{m} 2^{\log_2 p_i} \\
&= \sum_{i=1}^{m} p_i = 1
\end{aligned}$$

Thus completing the proof. □

Theorem 2.2.2. (**Source Coding Theorem for Symbol Codes**) For a discrete random variable X on $(\Omega, \mathcal{F}, P)$, there exists a prefix code $C : \mathcal{X} \to \Sigma^*$ such that

$$H(X) \leq \mathbb{E}[l(X)] < H(X) + 1$$

2.2.3 Huffman Codes

- **Idea**: Huffman codes are a realization of an optimal symbol code according to the source coding theorem.

Definition 2.2.3. (**Huffman Coding Algorithm**) A *Huffman code* $\mathscr{C} : \mathcal{X} \to \{0,1\}^*$ for the discrete random variable X on $(\Omega, \mathcal{F}, P)$, defined by the algorithm:

```
let rec huffman (p_1, ..., p_m) =
  if m = 2 then
    C s.t C(x_1) = 0, C(x_2) = 1
  else
    List.sort p_1 ≥ p_2 ≥ ... ≥ p_m;
    let C' = huffman (p_1, ..., p_{m-2}, p_{m-1} + p_m) in
    C s.t ∀i ≤ m - 2.C(x_i) = C'(x_i)
            ∧ C(x_{m-1}) = C'(x_{m-1}) · 0
            ∧ C(x_m) = C'(x_m) · 1
```

- Time complexity: $O(|\mathcal{X}|)$ (using bucket sort for sorting)
- **Problems**:
  - The additional bit in $H(X) + 1$ can be significiant if $H(X) < 1$.
    Solution: Encode blocks of size $N \implies \frac{1}{N}$ (at most) additional bits.
    Problem: Blocks result in exponential increase in $|\mathcal{X}|$.
  - Distribution of X must be *known* and *fixed*.
    Solution: Estimate distribution from compressed data and transmit in compressed file.
    Problem: Distirbution may be large, and may change on each compression (e.g. videos), not efficient!
  - Extension is not i.i.d.
    Solution: Blocks
    Adaptive coding schemes must process blocks in a top-down manner (as opposed to Huffman's bottom up approach).

Optimality

Let $X \sim \mathbf{p}$ be an arbitrary discrete random variable. Wlog. $\mathcal{X} = \{1, 2, \ldots, m\}$ and $p_1 \geq p_2 \geq \cdots \geq p_m$. Let us define X_{m-1} to be the random variable over

$\mathcal{X}_{m-1} = \{1, 2, \ldots, m-1\}$ and

$$P(X_{m-1} = i) = \begin{cases} p(i) & \text{if } 1 \leq i \leq m-2 \\ p(m-1) + p(m) & \text{if } i = m-1 \end{cases}$$

We define the *huffman split* of prefix code $\mathscr{C}_{m-1}$ as a prefix code for X given by:

$$\mathscr{C}(i) = \begin{cases} \mathscr{C}_{m-1}(i) & \text{if } 1 \leq i \leq m-2 \\ \mathscr{C}_{m-1}(i-1) \cdot 0 & \text{if } i = m-1 \\ \mathscr{C}_{m-1}(i-1) \cdot 1 & \text{if } i = m-2 \end{cases}$$

Lemma 2.2.4. Let $\mathscr{C}^{opt}_{m-1}$ be an optimal prefix code for X_{m-1}. Let $\mathscr{C}$ the huffman split of $\mathscr{C}^{opt}_{m-1}$. Then $\mathscr{C}$ is an optimal prefix code for X.

Proof. We note the following properties of an optimal prefix code $\mathscr{C}$:

(i) If $p(x) > p(y) \implies l(x) \leq l(y)$. Assume there exists $p(x) > p(y)$ s.t $l(x) > l(y)$, then $p(x)l(x) > p(y)l(y)$. Supposing we swapped the codewords of x and y, yielding code $\mathscr{C}'$. Then we have

$$\begin{aligned} \mathbb{E}_{\mathscr{C}'}[l(X)] - \mathbb{E}_{\mathscr{C}}[l(X)] &= p(x)l(y) + p(y)l(x) - (p(x)l(x) + p(y)l(y)) \\ &= p(x)(l(y) - l(x)) - p(y)(l(y) - l(x)) \\ &= \underbrace{(p(x) - p(y))}_{>0}\underbrace{(l(y) - l(x))}_{<0} \\ &< 0 \end{aligned}$$

Contradicting the assumption that $\mathscr{C}$ is optimal!

(ii) $l(m-1) = l(m) = l_{\max}$. Assume $l(m-1) < l(m) = l_{\max}$. Since the prefix property of C holds $\implies$ we can truncate codeword of m to $l(m-1)$, preserving prefix property and reducing $\mathbb{E}[l(X)]$. A contradiction!

(iii) $\mathscr{C}(m-1)$ and $\mathscr{C}(m)$ differ by last bit. Follows from the above property.

Properties (ii) and (iii) imply there is an optimal prefix code that is a result of a huffman split. We have the following expected length for a Huffman split:

$$
\begin{aligned}
\mathbb{E}[l(X)] &= \sum_{i=1}^{m} p(i)l(i) \\
&= \sum_{i=1}^{m-2} p(i)l(i) + p(m-1)l(m-1) + p(m)l(m) \\
&= \sum_{i=1}^{m-2} p(i)l(i) + (p(m-1) + p(m))(l_{m-1}(m-1) + 1) \\
&= \sum_{i=1}^{m-2} p_{m-1}(i)l(i) + p_{m-1}(m-1)(l_{m-1}(m-1) + 1) \\
&= \mathbb{E}[l_{m-1}(X_{m-1})] + p(m-1) + p(m)
\end{aligned}
$$

If $\mathbb{E}[l_{m-1}(X_{m-1})]$ is optimal, then it follows $\mathbb{E}[l(X)]$ is optimal for some fixed distribution $\mathbf{p}$. $\square$

2.2.4 Arithmetic Codes

- **Idea**: Adaptive compression using dependence between symbols. Requires top-down encoding for variable-length blocks (strings).

Definition 2.2.4. (**Segment Code**) A *segment code* $\mathscr{S}$ for the discrete random variable X on $(\Omega, \mathcal{F}, P)$, is a mapping from strings $\mathbf{x} \in \mathcal{X}^+$ to *segments* (or intervals) $\mathscr{S}(\mathbf{x}) = [l_{\mathbf{x}}, h_{\mathbf{x}})$ satisfying:

(i) $h_{\mathbf{x}} - l_{\mathbf{x}} = p(\mathbf{x})$

(ii) $\forall \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}^n . \mathbf{x}_1 \neq \mathbf{x}_2 \implies \mathscr{S}(\mathbf{x}_1) \cap \mathscr{S}(\mathbf{x}_2) = \emptyset$

(iii) $\bigcup_{\mathbf{x} \in \mathcal{X}^n} \mathscr{S}(\mathbf{x}) = [0, 1)$

(iv) $\mathbf{x}_1 = \mathsf{prefix}(\mathbf{x}_2) \implies \mathscr{S}(\mathbf{x}_1) \subseteq \mathscr{S}(\mathbf{x}_2)$

- A segment code is a prefix code by property (ii) and (iv).
  Properties (i) and (iii) are required for optimality.

- **Examples**:

- **Non-adaptive segment codes**: A non-adaptive (static) segment code $\mathscr{S}$ for X where $\mathcal{X} = \{x_1, x_2, \dots, x_m\}$ is defined by:

  $$\mathscr{S}(\mathbf{x}x_i) = [l_{\mathbf{x}} + p(\mathbf{x})F(x_{i-1}), l_{\mathbf{x}} + p(\mathbf{x})F(x_i))$$

  where F is the cdf:

  $$F(x_i) = \sum_{j=1}^{i} p_X(x_j)$$

- **Adaptive segment codes**: An adaptive segment code $\mathscr{S}$ for X where $\mathcal{X} = \{x_1, x_2, \dots, x_m\}$ is defined by:

  $$\mathscr{S}(\mathbf{x}x_i) = [l_{\mathbf{x}} + p(\mathbf{x})F(x_{i-1} \mid \mathbf{x}), l_{\mathbf{x}} + p(\mathbf{x})F(x_i \mid \mathbf{x}))$$

  where the conditional cdf F is:

  $$F(x_i \mid \mathbf{x}) = \sum_{j=1}^{i} p_X(x_j \mid \mathbf{x})$$

- **Idea**: **Arithmetic coding** is a segment code with a binary encoder that represents the segment using the minimal number of bits.

Definition 2.2.5. (**Arithmetic Code**) An *arithmetic code* $\mathscr{A} : \mathcal{X}^+ \to \{0,1\}^*$ is defined as a tuple $(\mathscr{S}, \mathscr{I})$ where $\mathscr{S} : \mathcal{X}^+ \to [0,1)$ is a segment code and $\mathscr{I} : [0,1) \to \{0,1\}^*$ is a (binary) interval encoder, such that

$$\mathscr{A}(\mathbf{x}) = \mathscr{I}(\mathscr{S}(\mathbf{x}))$$

where the interval encode $\mathscr{I}$ returns the binary representation of n for the interval $\mathcal{I}$ s.t the binary interval $\mathcal{I}_b = [n/2^k, (n+1)/2^k)$ is the largest interval satisfying $\mathcal{I}_b \subseteq \mathcal{I}$.

Analysis of Encoder

- **Problem**: Selecting binary interval $\mathcal{I}_{n,k} = \left[\frac{n}{2^k}, \frac{n+1}{2^k}\right)$ of length $\frac{1}{2^k}$ s.t $\mathcal{I}_{n,k} \subseteq [a, b)$.
- Maximizing length $1/2^k$ subject to $\frac{1}{2^k} \leq b - a$ yields $k = \lceil -\log_2(b-a) \rceil$.

- Also require constraint $a \leq n/2^k$ hence $n_k = \lceil 2^k a \rceil$.
- Remaining constraint: $\frac{n_k+1}{2^k} \leq b$.
  Cases:
  - If $\frac{n_k+1}{2^k} \leq b$. Return n_k
  - If $\frac{n_k+1}{2^k} > b$. Then $I_{k+1} \subseteq [a, b)$, as

    $$n_{k+1} - 1 = \lceil 2^{k+1} a \rceil - 1 < 2^{k+1} a$$

    hence

    $$\frac{n_{k+1}+1}{2^{k+1}} < a + \frac{2}{2^{k+1}} = a + \frac{1}{k} \leq a + (b-a) = b$$

    Return n_{k+1}

Lemma 2.2.5. For a non-adaptive arithmetic encoding $\mathscr{A}$,

$$H(X^n) \leq \mathbb{E}[l(X^n)] \leq H(X^n) + 2$$

Proof. Analysis of encoder yields

$$l(\mathbf{x}) \leq k + 1 = \lceil -\log_2 p(\mathbf{x}) \rceil + 1 \leq -\log_2 p(\mathbf{x}) + 2$$

Hence

$$\mathbb{E}[l(X^n)] \leq H(X^n) + 2$$

The lower bound follows from Theorem 2.2.2 □

- Given $H(X^n) = nH(X)$, then

  $$H(X) \leq \mathbb{E}[l(X)] \leq H(X) + \frac{2}{n}$$

  So for large n, we achieve optimal encoding (by squeeze theorem)!
- **Remark**: Upper bound holds for *adaptive encoding*
- Algorithm:

```
let 𝒜 x =
  let [l_u, h_u) = 𝒮 x in
  let r = first differing bit of l_u and h_u in
  (* l_u = 0.b_1 b_2 ··· b_{r-1} 0 ··· and l_u = 0.b_1 b_2 ··· b_{r-1} 1 ··· *)
  if 0.b_1 b_2 ··· b_{r-1} 1 < h_u then
    b_1 b_2 ··· b_{r-1} 1
  else
    (* assert: 0.b_1 b_2 ··· b_{r-1} 1 = h_u *)
    match l_u with
    | 0.b_1 b_2 ··· b_{r-1} 0 -> b_1 b_2 ··· b_{r-1}
    | 0.b_1 b_2 ··· b_{r-1} 0x when x = 0 ··· -> b_1 b_2 ··· b_{r-1} 01
    | 0.b_1 b_2 ··· b_{r-1} 0x when x = 1···1 (s times) 0 -> b_1 b_2 ··· b_{r-1} 0 1···1 (s+1 times)
```

- **Problems**: Still requires distribution prior to encoding

2.2.5 Lempel-Ziv Codes

- **Idea**: Replace previously seen substrings with pointers (or keys in a dictionary). Asymptotically optimal (especially for text).

- **Algorithm**:

  - Traverse string $\mathbf{x} = x_1 x_2 \ldots x_m$ emitting substrings that have not previously been seen.

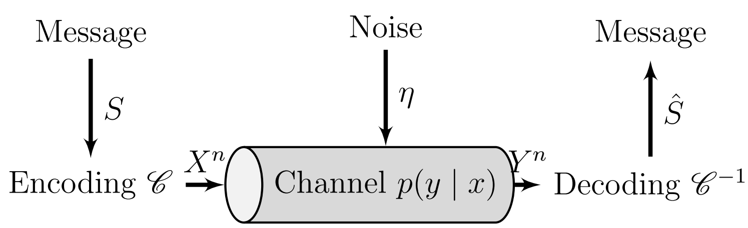
    For example: $1011010100010 \to \square, 1, 0, 11, 01, 010, 00, 10$ where $\square$ is the first (empty) substring.
  - Create a dictionary D mapping substrings to codewords (or pointers) in Σ.
  - Traverse substrings, applying dictionary.

Substrings	$\square$	1	0	11	01	010	00	10
Codeword	0	1	2	3	4	5	6	7
Codeword, Bit	–	–, 1	–, 0	1, 1	2, 1	4, 0	2, 0	1, 0

3 Channel Problems

- **Motiviation**: Study of communication in the presence of *noise*

Definition 3.0.1. (**Discrete Channel**) A discrete channel Q is a tuple $(\mathcal{X}, p_{Y,X}, \mathcal{Y})$ where $\mathcal{X}, \mathcal{Y}$ are the input,output alphabets of the channel and $p_{Y,X}$ is the conditional pmf of discrete random variables X, Y over $\mathcal{X}, \mathcal{Y}$.



- A channel is *memoryless* if the current output *only* depends on the current input:

$$p_{Y_n|X^n}(y_n \mid x_1, \ldots, x_n) = p_{Y_n|X_n}(y_n \mid x_n)$$

- Discrete memoryless channel = DMC

Definition 3.0.2. (**Rate**) The rate R of a channel Q with code $\mathscr{C}$ is defined as the expected information (in bits) transmitted per a symbol:

$$R = \mathbb{E}\left[\frac{h(S)}{l(S)}\right] = \frac{H(S)}{\mathbb{E}[l(S)]}$$

Definition 3.0.3. (**Error Probability**) The error probability of a code $\mathscr{C}$, for source S and channel $Q = (\mathcal{X}, p_{Y|X}, \mathcal{Y})$ is

$$p_e(\mathscr{C}) = P(\hat{S} \neq S) = \sum_{s \in \overrightarrow{X}(\Omega)} \lambda_s p_S(s)$$

where the conditional probability of error λ_s is

$$\lambda_s = P(\hat{S} \neq s \mid S = s) = P(\mathscr{C}^{-1}(Y^n) \neq s \mid X^n = \mathscr{C}(s))$$

Definition 3.0.4. (**Achievable**) A rate R is achievable for the channel Q if there exists a sequence of codes $(\mathscr{C}_i)_{i\geq 1}$ s.t

(i) $R_i < R$ for all codes $i \geq 1$

(ii) $\lim_{n\to\infty} p_e^n = 0$ where $p_e^n = p_e(\mathscr{C}_i)$

3.1 Shannon's Channel Coding Theorem

3.1.1 Definitions

Definition 3.1.1. (**(M, n) codes**) An (M, n) code for the channel $Q = (\mathcal{X}, p_{Y|X}, \mathcal{Y})$ consists of:

(i) A domain of *messages* $M = \{1, 2, \ldots, M\}$ (we use M interchangably for the set and it's cardinality).

(ii) An encoding function $\mathscr{C} : M \to \mathcal{X}^n$. The set of codewords is called the codebook $\mathcal{C} = \{\mathscr{C}(1), \ldots, \mathscr{C}(|M|)\}$.

(iii) A decoding function $\mathscr{C}^{-1} : \mathcal{Y}^n \to M$

- Wlog. we use (M, n) codes where $S \sim U(M)$ **is uniformly distributed**.

Lemma 3.1.1. (**Properties of (M, n) codes**)

(i) Rate of a (M, n) code is $R = \log_2(|\mathcal{C}|)/n$.

(ii) Rate R is achievable if there exists a sequence of $(\lceil 2^{nR} \rceil, n)$ codes s.t $\lim_{n\to\infty} p_e^n = 0$.

Definition 3.1.2. (**Capacity**) The capacity C of a channel Q is defined as:

$$C = \sup\{R : R \text{ is achievable}\}$$

- See Section 3.2 for properties and examples.

3.1.2 Jointly Typical Sets

- **Motivation**: Extend typicality to joint distributions as noisy channel problems deal w/ joint distributions.

Definition 3.1.3. (**Jointly Typical Set**) A *jointly typical set* $A_\epsilon^n(X, Y)$ wrt discrete random variables X, Y is the set of string pairs $(\mathbf{x}, \mathbf{y}) \in \overrightarrow{X^n}(\Omega) \times \overrightarrow{Y^n}(\Omega)$ s.t

$$\begin{aligned} 2^{-n(H(X)+\epsilon)} &< p(\mathbf{x}) < 2^{-n(H(X)-\epsilon)} \\ 2^{-n(H(Y)+\epsilon)} &< p(\mathbf{y}) < 2^{-n(H(Y)-\epsilon)} \\ 2^{-n(H(X,Y)+\epsilon)} &< p(\mathbf{x}, \mathbf{y}) < 2^{-n(H(X,Y)-\epsilon)} \end{aligned}$$

We have

$$\begin{aligned} A_\epsilon^n(X, Y) = \Bigg\{ (\mathbf{x}, \mathbf{y}) \in \overrightarrow{X^n}(\Omega) \times \overrightarrow{Y^n}(\Omega) : \left| \frac{1}{n} h(\mathbf{x}) - H(X) \right| < \epsilon, \left| \frac{1}{n} h(\mathbf{y}) - H(Y) \right| < \epsilon, \\ \left| \frac{1}{n} h(\mathbf{x}, \mathbf{y}) - H(X, Y) \right| < \epsilon \Bigg\} \end{aligned}$$

Theorem 3.1.1. (**Joint asymptotic equipartition property**) Let (X^n, Y^n) be i.i.d sequences of length n distributed by $p_{X^n, Y^n}(\mathbf{x}, \mathbf{y}) = \prod_{i=1}^n p_{X,Y}(x_i, y_i)$. Then

$$\lim_{n \to \infty} P((X^n, Y^n) \in A_\epsilon^n(X, Y)) = 1$$

Proof. Follows directly from Theorem 2.1.1. □

Lemma 3.1.2. (**Properties of** $A_\epsilon^n(X, Y)$)

- For sufficiently large n, and $(X^n, Y^n) \sim p_{X^n} p_{Y^n}$,

$$(1 - \epsilon) 2^{-n(I(X;y)+3\epsilon)} \leq P((X^n, Y^n) \in A_\epsilon^n(X, Y)) \leq 2^{-n(I(X;y)-3\epsilon)}$$

- For sufficiently large n,

$$|A_\epsilon^n(X, Y)| < 2^{n(H(X,Y)+\epsilon)}$$

- For sufficient large n,

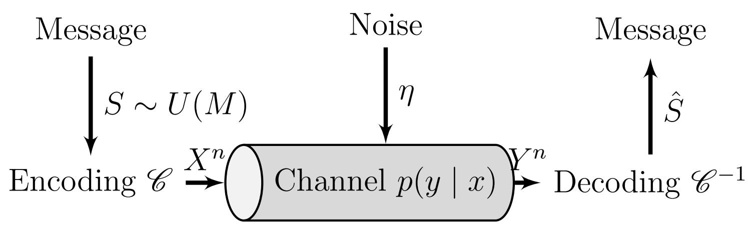
$$P((X^n, Y^n) \in A_\epsilon^n(X, Y)) \geq 1 - \epsilon$$

3.1.3 Channel Coding Theorem

Theorem 3.1.2. (**Channel Coding Theorem**) The capacity of a DMC $(\mathcal{X}, p_{Y|X}, \mathcal{Y})$ is

$$C = \max_{p_X} I(X;Y)$$

where Y is distributed by $p_Y(y) = \sum_{x \in \mathcal{X}} p_{Y|X}(y \mid x) p_X(x)$



- Theorem is proved in 2 parts:
  - (I) $R \leq \max_{p_X} I(X;Y) \implies R$ is achievable.
  - (II) R is achievable $\implies R \leq \max_{p_X} I(X;Y)$

Theorem 3.1.3. (**Channel Coding Theorem Part I**) For the DMC $Q = (\mathcal{X}, p_{Y|X}, \mathcal{Y})$,

$$R \leq \max_{p_X} I(X;Y) \implies R \text{ is achievable (on } Q)$$

Proof. Let R be arbitrary. We assume there exists p_X s.t $R \leq \max_{p_X} I(X;Y)$.
Proof Idea:

1. Construct sequence of $(\lceil 2^{nR} \rceil, n)$ codes using typical sets.
2. Perform error analysis.

Let $M = \lceil 2^{nR} \rceil$. Let $s \in M$ be a message. We exhibit the (M, n) code $\mathscr{C}$ as the matrix:

$$\mathscr{C} = \begin{bmatrix} \mathscr{C}(1) \\ \mathscr{C}(2) \\ \vdots \\ \mathscr{C}(\lceil 2^{nR} \rceil) \end{bmatrix} = \begin{bmatrix} X_1(1) & X_2(1) & \cdots & X_n(1) \\ X_1(2) & X_2(2) & \cdots & X_n(2) \\ \vdots & \vdots & \ddots & \vdots \\ X_1(\lceil 2^{nR} \rceil) & X_2(\lceil 2^{nR} \rceil) & \cdots & X_n(\lceil 2^{nR} \rceil) \end{bmatrix}$$

where the i.i.d random variable X_j on $(M, \mathcal{F}, P)$ is distributed by p_X.

The code $\mathscr{C}$ is known both to the sender and reciever. The encoded message $\mathbf{x}$ has probability $p(\mathbf{x}) = \prod_{i=1}^n p_X(x_i)$. The recieved code $\mathbf{y}$ has probability $p(\mathbf{y} \mid \mathbf{x}) = \prod_{i=1}^n p_{Y|X}(y_i \mid x_i)$.

Typical set decoding. The decoder iterates over $\hat{s} \in M$ decoding $\mathbf{y}$ as $\hat{s}$ if $\hat{s}$ is the *unique message* s.t $(\mathscr{C}(\hat{s}), \mathbf{y}) \in A_\epsilon^n(X, Y)$ Otherwise, set $\hat{s} = 0$ (fail).

We now consider the error analysis. Given that $S \sim U(M)$, we have

$$
\begin{aligned}
p_e^n = P(\hat{S} \neq S) &= \sum_{s \in \overrightarrow{S}(\Omega)} \lambda_s p_S(s) \\
&= \frac{1}{\lceil 2^{nR} \rceil} \sum_{s=1}^{\lceil 2^{nR} \rceil} \lambda_s
\end{aligned}
$$

Let E_s denote the event $(\mathscr{C}(s), Y^n) \in A_\epsilon^n(X, Y)$. Considering λ_s yields

$$
\begin{aligned}
\lambda_s &= P(\hat{S} \neq s \mid S = s) \\
&= P(\mathscr{C}^{-1}(Y^n) \neq s \mid X^n = \mathscr{C}(s)) \\
&= P\left(\overline{E_s} \cup \bigcup_{s' \neq s} E_{s'}\right) \\
&\leq P(\overline{E}_s) + \sum_{s \neq s'} P(E_{s'})
\end{aligned}
$$

By the Joint AEP (Theorem 3.1.1)

$$
\lim_{n \to \infty} P((X^n, Y^n) \in A_\epsilon^n(X, Y)) \geq 1 - \epsilon
$$

Given that $E_s \subseteq (X^n, Y^n) \in A_\epsilon^n(X, Y)$, we have

$$
\begin{aligned}
\lambda_s &\leq \epsilon + \sum_{s' \neq s} 2^{-n(I(X;Y) - 3\epsilon)} \\
&= \epsilon + (\lceil 2^{nR} \rceil - 1) 2^{-n(I(X;Y) - 3\epsilon)} \\
&\leq \epsilon + 2^{-n(I(X;Y) - 3\epsilon - R)}
\end{aligned}
$$

Since $R < I(X; Y) - 3\epsilon$, it follows that $\lim_{n \to \infty} p_e^n = 0$. $\square$

Theorem 3.1.4. (**Fano's Inequality**) Let X, Y be a discrete random variable on $(\Omega, \mathcal{F}, P)$ and $\hat{X} = f(Y)$, where $f : \mathcal{Y} \to \mathcal{X}$. Let $p_e = p(\hat{X} \neq X)$, then $H(X \mid \hat{X}) \leq H_2(p_e) + p_e \log_2 |\overrightarrow{X}(\Omega)|$

Theorem 3.1.5. (**Channel Coding Theorem Part II**) For the DMC $Q = (\mathcal{X}, p_{Y|X}, \mathcal{Y})$,

$$R \text{ is achievable (on } Q) \implies R \leq \max_{p_X} I(X;Y)$$

Proof. Applying Fano's inequality to the channel coding theorem yields

$$\begin{aligned} H(S \mid \hat{S}) &\leq H_2(p_e) + p_e \log_2 |S| \\ &\leq 1 + p_e n R \end{aligned}$$

Given that $S \sim U(\lceil 2^{nR} \rceil)$, we have

$$\begin{aligned} H(S) &= nR \\ &= H(S \mid \hat{S}) - I(S;\hat{S}) \\ &\leq 1 + p_e nR + I(X^n;Y^n) \\ &\leq 1 + p_e nR + n \max_{p_X} I(X;Y) && \text{(Memoryless)} \\ \iff R &\leq \frac{1}{n} + p_e R + \max_{p_X} I(X;Y) \end{aligned}$$

Assuming R is achievable, hence $\lim_{n\to\infty} p_e^n = 0$, then $R \leq \max_{p_X} I(X;Y)$. □

3.2 Capacity

Definition 3.2.1. (**Capacity**) The capacity of a channel is defined as

$$C = \max_{p_X} I(X;Y)$$

- Above defn. follows from Shannon's Coding Theorem.
- We assume the *memoryless property*: $p_{Y^n|X^n}(y^n \mid x^n) = \prod_{i=1}^n p_{Y|X}(y_i \mid x_i)$.
- Transition probabilities may be written as a matrix:

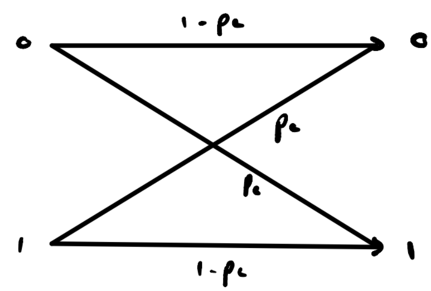
  $$Q_{ij} = p_{Y|X}(y_j \mid x_i)$$

  Hence $\mathbf{p}_Y = \mathbf{Q}\mathbf{p}_X$.

3.2.1 Binary Symmetric Channels

- Let X and Y be discrete random variables s.t $\overrightarrow{X}(\Omega) = \overrightarrow{Y}(\Omega) = \{0, 1\}$, distributed by $X \sim \mathsf{Bern}(p_X)$ and

$$p_{Y|X}(y \mid x) = \begin{cases} p_e & \text{if } x \neq y \\ 1 - p_e & \text{if } x = y \end{cases}$$



- Considering $I(X;Y)$, we have

$$I(X;Y) = H(Y) - H(Y \mid X)$$

Considering the distribution of Y yields:

$$\begin{aligned} p_Y = p_Y(1) &= \sum_{x \in \{0,1\}} p_{Y|X}(y \mid x) p_X(x) \\ &= p_X(1 - p_e) + (1 - p_X)p_e \end{aligned}$$

Hence

$$\begin{aligned} H(Y) &= -\sum_{y \in \{0,1\}} p_Y(y) \log_2 p_Y(y) \\ &= -p_Y \log_2 p_Y - (1 - p_Y) \log_2(1 - p_Y) \\ &= H_2(p_Y) \end{aligned}$$

where $H_2(p) = -p \log_2 p - (1 - p) \log_2(1 - p)$ is the *binary entropy function*. The conditional entropy is given by:

$$H(Y \mid X) = -\sum_{x \in \{0,1\}} p_X(x) \sum_{y \in \{0,1\}} p_{Y|X}(y \mid x) \log_2 p_{Y|X}(y \mid x)$$

Consdering $H(Y \mid X = x)$ for $x \in \{0,1\}$ gives us:

$$-\sum_{y\in\{0,1\}} p_{Y|X}(y \mid 0)\log_2 p_{Y|X}(y \mid 0) = -(1-p_e)\log_2(1-p_e) - p_e\log_2 p_e = H_2(p_e)$$

$$-\sum_{y\in\{0,1\}} p_{Y|X}(y \mid 1)\log_2 p_{Y|X}(y \mid 1) = -p_e\log_2 p_e - (1-p_e)\log_2(1-p_e) = H_2(p_e)$$

So

$$\begin{aligned} H(Y \mid X) &= \sum_{x\in\{0,1\}} p_X(x)H(Y \mid X = x) \\ &= H_2(p_e)\sum_{x\in\{0,1\}} p_X(x) = H_2(p_e) \end{aligned}$$

Thus the mutual information is

$$\begin{aligned} I(X;Y) &= H(Y) - H(Y \mid X) \\ &= H_2(p_X(1-p_e) + (1-p_X)p_e) - H_2(p_e) \end{aligned}$$

- Maximizing $I(X;Y)$ gives the capacity $C = 1 - H_2(p_e)$.

3.2.2 Binary Erasure Channels

- Let X and Y be discrete random variables s.t $\overrightarrow{X}(\Omega) = \{0,1\}$, $\overrightarrow{Y}(\Omega) = \{0,1,?\}$, distributed by $X \sim \mathsf{Bern}(p_X)$ and

$$p_{Y|X}(y \mid x) = \begin{cases} p_e & \text{if } y = ? \\ 1-p_e & \text{if } y = x \end{cases}$$

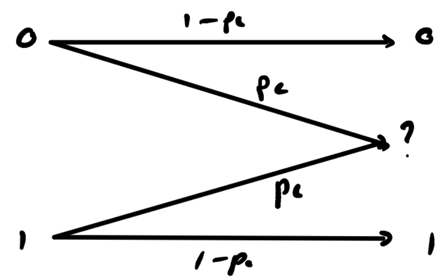
- Considering the mutual information $I(X;Y)$ given by

$$I(X;Y) = H(X) - H(X \mid Y)$$

So we have:

$$\begin{aligned} H(X) &= H_2(p_X) \\ H(X \mid Y) &= \sum_{y\in\{0,1,?\}} p_Y(y)H(X \mid Y = y) \\ &= p_Y(0)H(X \mid Y = 0) + p_Y(1)H(X \mid Y = 1) + p_Y(?)H(X \mid Y = ?) \\ &= 0 + 0 + p_eH_2(p_X) = p_eH_2(p_X) \end{aligned}$$

Thus $I(X;Y) = (1-p_e)H_2(p_X)$.



- Maximizing $I(X;Y)$ gives the capacity $C = 1 - p_e$.

3.2.3 Gaussian Channels

- **Motivation**: Signals are continuous, so is noise. Noise is the sum of many induvidual signals (Fourier series) $\implies$ by CLT, noise is normally distributed.

Definition 3.2.2. (**Gaussian Channel**) A gaussian channel G is a discrete-time channel with input X_t and output Y_t, and noise Z_t at time t s.t

$$Y_t = X_t + Z_t, \qquad Z_t \sim \mathcal{N}(0, \sigma^2)$$

- If $\sigma^2 = 0$ or input is unconstrainted, then $C = \infty$!
- **Power limitation**: Limitation is on the power of the input $\mathbb{E}[X^2] \leq P$ (Physics: amplitude is bounded by power)

Theorem 3.2.1. The capacity of a Gaussian channel G with power constraint P and noise variance σ^2 is

$$C = \max_{f_X : \mathbb{E}[X^2] \leq P} I(X;Y) = \frac{1}{2} \log \left(1 + \frac{P}{\sigma^2}\right)$$

Proof. *(Assuming capacity result for DMCs generalizes to gaussian channels)*

We have

$$\begin{aligned} I(X;Y) &= \mathrm{d}H(Y) - \mathrm{d}H(Y \mid X) \\ &= \mathrm{d}H(Y) - \mathrm{d}H(X + Z \mid X) \\ &= \mathrm{d}H(Y) - \mathrm{d}H(Z) \\ &\leq \mathrm{d}H(\mathcal{N}(0, P+\sigma^2)) - \mathrm{d}H(\mathcal{N}(0,\sigma^2)) \\ &= \frac{1}{2}\log 2\pi e(P+\sigma^2) - \frac{1}{2}\log 2\pi e\sigma^2 \\ &= \frac{1}{2}\log\left(1 + \frac{P}{\sigma^2}\right) \end{aligned}$$

(Assuming X is a Gaussian) Hence capacity is

$$C = \frac{1}{2}\log\left(1 + \frac{P}{\sigma^2}\right)$$

□

- Proof that DMC result generalizes to continuous information is *fiddly*.
- Proof that X is Gaussian relies on Lagrangian Multipliers.

Definition 3.2.3. (**Bandlimited Channel**) A bandlimited channel B is a Gaussian channel G with an impluse response function $h(t)$ of an ideal *bandpass* filter, which cuts off all frequencies $> W$, where

$$Y_t = (X_t + Z_t) \otimes h(t)$$

Definition 3.2.4. (**Nyquist-Shannon Sampling Theorem**) Suppose that $f(t)$ is bandlimited to W. Then the function is completely determined by samples of the function spaced $\frac{1}{2W}$ seconds apart.

- By Nquist-Shannon theorem, power constraint is $P/2W$ (per sample) and noise variance is σ^2 (per sample), hence capacity is

$$\begin{aligned} C &= 2W\frac{1}{2}\log\left(1 + \frac{\frac{P}{2W}}{\sigma^2}\right) \\ &= W\log\left(1 + \frac{P}{2W\sigma^2}\right) \end{aligned}$$

- Minimize power usage by using larger bandwidth W (UWB).

3.3 Error Correcting Codes

Definition 3.3.1. (**Error Correcting Code**) Error correcting codes are codes $\mathscr{C} : \mathcal{X} \to \Sigma^*$ s.t probability of error $p_e(\mathscr{C})$ is minimized (ideally 0) over a noisy channel.

- Primarily split into 2 categories:
  - **Block Codes**: (M, n) block codes which encode M bits using n bits $\implies n - M$ error correction bits.
  - **Convolution Codes**: Similar to streaming codes (See segment codes). Often decoded using the Viterbi algorithm (modelling a sliding window of bits as a Markov Process).

3.3.1 Repitition Codes

Definition 3.3.2. (**Repitition Codes**) A r-repitition code $\mathscr{C}^r : \mathcal{X} \to \Sigma^*$ of a code $\mathscr{C} : \mathcal{X} \to \Sigma^*$ is defined as:

$$\mathscr{C}^r(x) = \underbrace{\mathscr{C}(x)\mathscr{C}(x)\cdots\mathscr{C}(x)}_{r \text{ times}}$$

- **Problem**: Optimal decoding for a DMC $Q = (\mathcal{X}, p_{Y|X}, \mathcal{Y})$.
- Considering $P(S = s \mid Y^r = \mathbf{y})$:

$$\begin{aligned} P(S = s \mid Y^r = \mathbf{y}) &= P(X^r = \mathscr{C}^r(s) \mid Y^r = \mathbf{y}) \\ &= \frac{P(Y^r = \mathbf{y} \mid X^r = \mathscr{C}^r(s))P(X^r = \mathscr{C}^r(s))}{P(Y^r = \mathbf{y})} \end{aligned}$$

  By memorylessness:

$$\begin{aligned} P(Y^r = \mathbf{y}) &= \prod_{i=1}^{r} p_Y(y_i) \\ P(X^r = \mathscr{C}^r(s)) &= p_S(s) \end{aligned}$$

Now consider $P(Y^r = \mathbf{y} \mid X^r = \mathscr{C}^r(s))$:

$$\begin{aligned} P(Y^r = \mathbf{y} \mid X^r = \mathscr{C}^r(s)) &= \prod_{i=1}^{r} P(Y_i = y_i \mid X_i = \mathscr{C}(s)) \\ &= \prod_{i=1}^{r} p_{Y|X}(y_i \mid \mathscr{C}(s)) \end{aligned}$$

So:

$$P(S = s \mid Y^r = \mathbf{y}) = p_S(s) \prod_{i=1}^{r} \frac{p_{Y|X}(y_i \mid \mathscr{C}(s))}{p_Y(y_i)}$$

- Optimal repitition decoder is given by

$$\begin{aligned} \mathscr{C}^{-r}(\mathbf{y}) &= \arg\max_{s \in \mathcal{S}} P(S = s \mid Y^r = \mathbf{y}) \\ &= \arg\max_{s \in \mathcal{S}} p_S(s) \prod_{i=1}^{r} \frac{p_{Y|X}(y_i \mid \mathscr{C}(s))}{p_Y(y_i)} \end{aligned}$$

  Often assume uniform source $S \sim U(M)$

$$\mathscr{C}^{-r}(\mathbf{y}) = \arg\max_{s \in S} \prod_{i=1}^{r} p_{Y|X}(y_i \mid \mathscr{C}(s))$$

- **Examples**:

  BSC Channel given by

$$p_{Y|X}(y \mid x) = \begin{cases} p_e & \text{if } x \neq y \\ 1 - p_e & \text{if } x = y \end{cases}$$

$$\begin{aligned} \mathscr{C}^{-r}(\mathbf{y}) &= \arg\max_{s \in S} \prod_{i=1}^{r} p_{Y|X}(y_i \mid \mathscr{C}(s)) \\ &= \arg\max_{s \in S} p_e^{N_s} (1 - p_e)^{r - N_s} \end{aligned}$$

  where $N_s = \sum_{i=1}^{r} I_{y_i \neq \mathscr{C}(s)}$.

$$
\begin{aligned}
p_e(\hat{S} \neq S) &= P\left(\frac{p_e^{N_{\hat{S}}}(1-p_e)^{r-N_{\hat{S}}}}{p_e^{N_S}(1-p_e)^{r-N_S}} > 1\right) \\
&= P(\gamma^{N_{\hat{S}}-N_S} > 1) \\
&= P(p_e < 0.5 \wedge N < 0) + P(p_e \geq 0.5 \wedge N > 0) \\
&= P\left(\text{number of bit flips} > \left\lceil \frac{r}{2} \right\rceil\right) \\
&= \sum_{n=\frac{r+1}{2}}^{r} p_e^n (1-p_e)^{r-n}
\end{aligned}
$$

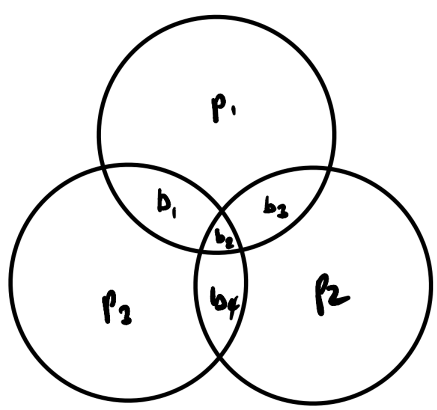
3.3.2 Hamming Codes

- (N, k) block codes $= N$ total bits encoding k bits ($N - k$ error bits).

Definition 3.3.3. (**(7, 4) Hamming Code**) A (7, 4) Hamming Code is a code $\mathscr{C}^{(7,4)} : \{0,1\}^4 \to \{0,1\}^7$ defined by

$$\mathscr{C}^{(7,4)}(b_1 b_2 b_3 b_4) = b_1 b_2 b_3 b_4 p_1 p_2 p_3$$

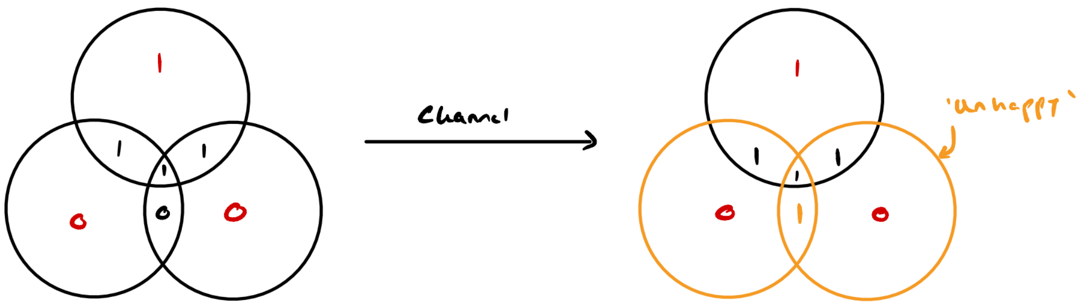
where p_1, p_2, p_3 are parity bits given by:



- **'Syndrome' Decoding**:

  1. Count number of circles that a 'unhappy' (parity-check fails). Forms a 3-bit 'syndrome' **z**.

2. Decoding consists of finding a unique bit inside all the 'unhappy' circles and outside the 'happy' circles that would fix the syndrome.



- (7,4) Codes cannot deal with > 1 bit-flip. Most channels have $p_e \ll 1$ $\implies > 1$ bit-flip is v. unlikely.

- **Linear Codes**: Codes of the form $\mathbf{x} = \mathbf{G}^T\mathbf{s}$ for source input $\mathbf{s}$ and channel input $\mathbf{x}$ (mod 2).

  Decoding given $\hat{\mathbf{s}} = \mathbf{H}\mathbf{y}$. $\mathbf{H}$ must satisfy $\mathbf{H}\mathbf{G}^T = \mathbf{0}$.

- **Linear 'Syndrome' Decoding**: Given $\mathbf{y} = \mathbf{G}^T\mathbf{s} + \boldsymbol{\eta}$, syndrome decoding is the process (using MLE) of finding the most probable $\boldsymbol{\eta}$ s.t $\mathbf{H}\boldsymbol{\eta} = \mathbf{z}$.