Kraft-McMillan Inequality (K-M inequality):

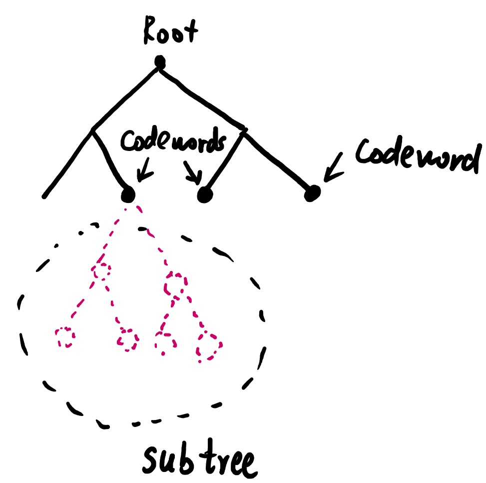
(1) If a code C is uniquely decodable, then K(C) = $\sum_{i=1}^{N} 2^{-l_i} \leqslant 1$
Where N is the number of codewords in code C, and $l_1, l_2, \cdots, l_N$
Are the codeword lengths.

(2) If K(C) $\leqslant 1$, then we can always construct a prefix code with codeword lengths being $l_1, l_2, \cdots, l_N$.

$\uparrow$

Prove (2):

Construct a prefix code:
Assign some vertices as codewords, then we cannot assign codeword to any leaves belonging to the subtree rooted at that codeword.
Look at the number of leaf nodes:



Given the codeword lengths:

$l_1, l_2, \cdots, l_N$

define

$l = \max\{l_1, l_2, \cdots, l_N\}$

Construct a full binary tree of length l, which has 2^l leaf nodes.

Next, assign a codeword to vertex V_1 at level l_1, then the path from the root to vertex V_1 has a binary code with length l_1. And there is a need to prune the subtree rooted at V_1, resulting in a lost of 2^{l-l_1} leaf nodes.

Likewise, the number of leaf nodes lost for each codeword assignment:

Given the codeword lengths :

$$\begin{array}{cccc} l_1, & l_2, & \cdots, & l_N \\ \downarrow & \downarrow & & \downarrow \\ 2^{l-l_1} & 2^{l-l_2} & \cdots & 2^{l-l_N} \end{array}$$

The total number of leaf nodes needed to build a code:

$$\sum_{i=1}^{N} 2^{l-l_i} = 2^{l} \cdot \sum_{i=1}^{N} 2^{-l_i} \leq 2^{l}$$

Sine $K(C) = \sum_{i=1}^{N} 2^{-l_i} \leq 1$

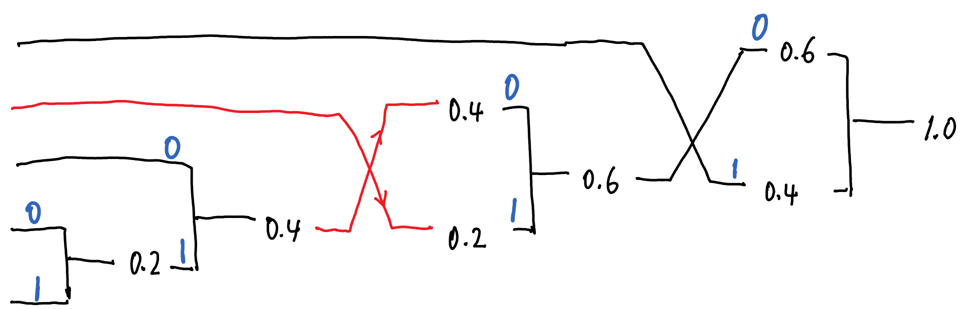
Construct a full binary tree of depth l, which has 2^{l} leaf nodes.

Therefore, we can always construct a prefix code.

Huffman Code (example)

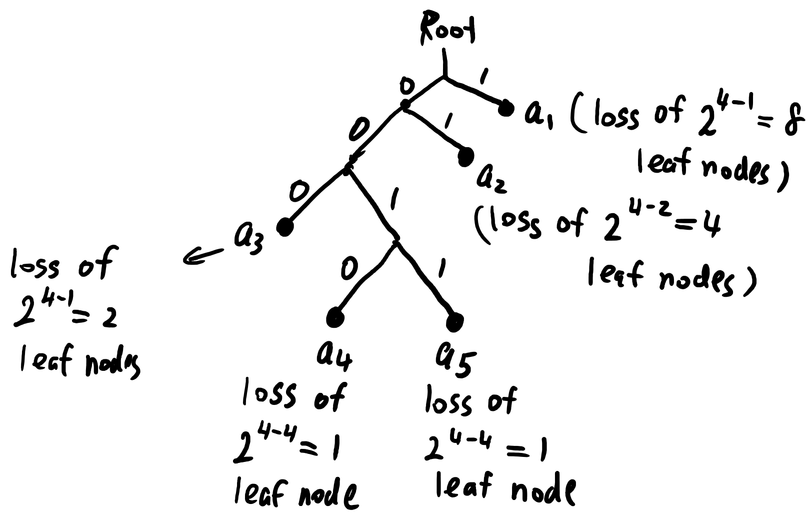
Alphabet =
$\{a_1, a_2, a_3, a_4, a_5\}$

Codeword	Symbol	Prob
1	a_1	0.4
01	a_2	0.2
000	a_3	0.2
0010	a_4	0.1
0011	a_5	0.1



$K(C) = 2^{-1} + 2^{-2} + 2^{-3} + 2^{-4} + 2^{-4} = 1 \leq 1$

$$\ell = \max\{\ell_1, \dots, \ell_5\} = 4$$



Leaf nodes lost:

$$\underset{a_1}{8} + \underset{a_2}{4} + \underset{a_3}{2} + \underset{a_4}{1} + \underset{a_5}{1} = 16 \text{ nodes} = 2^{l} = 2^4$$

- Length of Huffman codes

Answer the following question:
Are we losing on the *coding efficiency* (in terms of average codeword length) if we restrict ourselves to prefix codes?

For a source with alphabet $A = \{a_1, a_2, \dots, a_K\}$ and probability model: $\{P(a_1), P(a_2), \dots, P(a_K)\}$, then the average codeword length (ACL) is given by:

$$\bar{l} = \sum_{i=1}^{k} P(a_i)\, l_i$$

It can be shown that:

$$H(S) \leq \bar{l} \leq H(S) + 1.$$

First prove the lower bound:

$$H(S) - \bar{l} \leq 0$$

where

$$H(S) - \bar{l} = -\sum_{i=1}^{k} p(a_i) \log_2 p(a_i) - \sum_{i=1}^{k} p(a_i) l_i$$

$$= \sum_{i=1}^{k} p(a_i) \log_2 \left[\frac{2^{-l_i}}{p(a_i)} \right]$$

Consider Jensen's Inequality:

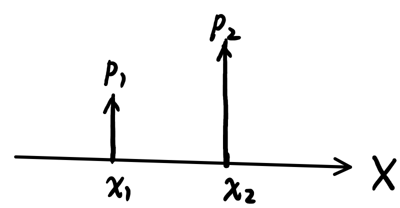
If f(x) is a concave function, then E[f(X)] <= f(E[X]), where X is a random variable, and E[X] is the expected value of X.

E[f(X)] <= f(E[X])

$$\Downarrow$$

$$E[\log_2(X)] \leq \log_2 E[X]$$

Define a two mass-point distribution:



$$E[f(X)] = p_1 \cdot f(x_1) + p_2 \cdot f(x_2)$$

$$f(E[X]) = f(p_1 \cdot x_1 + p_2 \cdot x_2)$$

E[f(X)] <= f(E[X])

$$p_1 \cdot f(x_1) + p_2 \cdot f(x_2) \leq f(p_1 \cdot x_1 + p_2 \cdot x_2)$$

For a general k mass-point distribution:

X (a RV) takes $x_1, x_2, \dots, x_k$

$$\sum_{i=1}^{k} p_i \cdot f(x_i) \leq \underset{\downarrow \; \log_2(\,)}{f} \left(\sum_{i=1}^{k} p_i \cdot x_i \right)$$

Thus,

$$H(S) - \bar{l} = \sum_{i=1}^{k} p(a_i) \log_2 \left[\frac{2^{-l_i}}{p(a_i)} \right] \leq \log_2 \Big\{ \underbrace{\sum_{i=1}^{k} \left[p(a_i) \cdot \frac{2^{-l_i}}{p(a_i)} \right]}_{\sum_{i=1}^{k} 2^{-l_i} = K(C) \leq 1} \Big\} \leq \log_2 1 = 0$$

Hence, $H(S) \leq \bar{l}$.