Date
Page

Lec 9: Data Compression

Sep 20th 2017

Defn- A source code $\mathcal{C}$ for a random variable X is a mapping from

$$x \longrightarrow \mathbb{D}^* \quad \text{(strings of finite length from alphabet } \mathbb{D})$$

$\therefore \forall\ x \in \mathcal{X}_0 \Rightarrow c(x) \in \mathbb{D}^*$

$\ell(x)$ = length of $c(x)$
(# $\mathbb{D}$-symbols in $c(x)$)

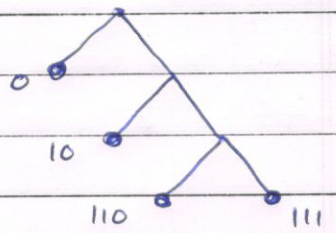
Defn- The expected length $L(\mathcal{C})$ of a source code $\mathcal{C}$ for a random variable X having pmf $p(x)$ is given by,

$$L(\mathcal{C}) = \sum_{x \in \mathcal{X}_0} p(x)\,\ell(x)$$

we assume wolog, $\mathbb{D} = \{0, 1, \ldots, D-1\}$.

- Note that any string of encoded bits (without commas) can be uniquely decoded, if the forward mapping is 1:1. (Eg. 5.1.1 in Cover & Thomas)

The code can be visualised as a tree



Date
Page

Defn- A code is said to be nonsingular if every element in the range of x maps onto a different string in $\mathbb{D}^*$,

ie- $x \neq x' \Rightarrow C(x) \neq C(x')$

Defn- The extension C^* of a code C is the mapping from finite length strings of $\mathcal{X}$ to finite length strings of $\mathbb{D}$, defined by:

$$C(x_1 \dots x_n) = C(x_1)\, C(x_2) \dots C(x_n)$$

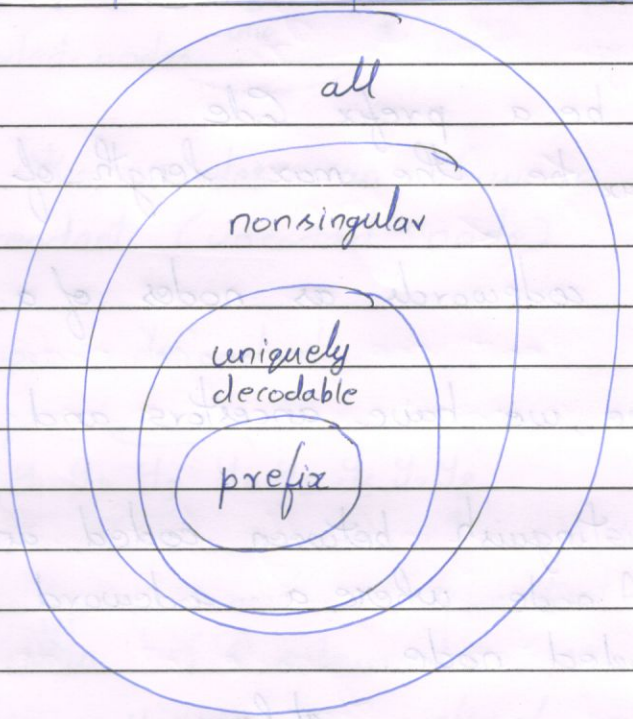
A code is said to be uniquely decodable if its extension is non-singular (however you may need to look at the entire string before decoding even the 1st source symbol).

Defn- A code is said to be a prefix code or an instantaneous code if no codeword is a prefix of any other codeword.

singular ↑ non-singular ↑ uniquely decodable ↑ prefix ↑

Code

X	x_1	x_2	x_3	x_4
1	0	0	10	0
2	0	010	00	10
3	1	01	11	110
4	1	10	110	111



Nesting of different Code Classes

Note- x_3 is uniquely decodable, but must wait till end of sequence for decoding correctly.

Krafts Inequality -

Date
Page

Theorem- A prefix code over an alphabet $\mathbb{D}$ of size D and with codeword $c(x_i)$ having length l_i, $1 \le i \le n$ exists iff,

$$\sum_{i=1}^{n} D^{-l_i} \le 1 \quad \ldots\ldots (1)$$

Proof- Let $\mathcal{L}$ be a prefix code

Let l_{max} be the max length of a codeword

Place the codewords as nodes of a D-ary tree.

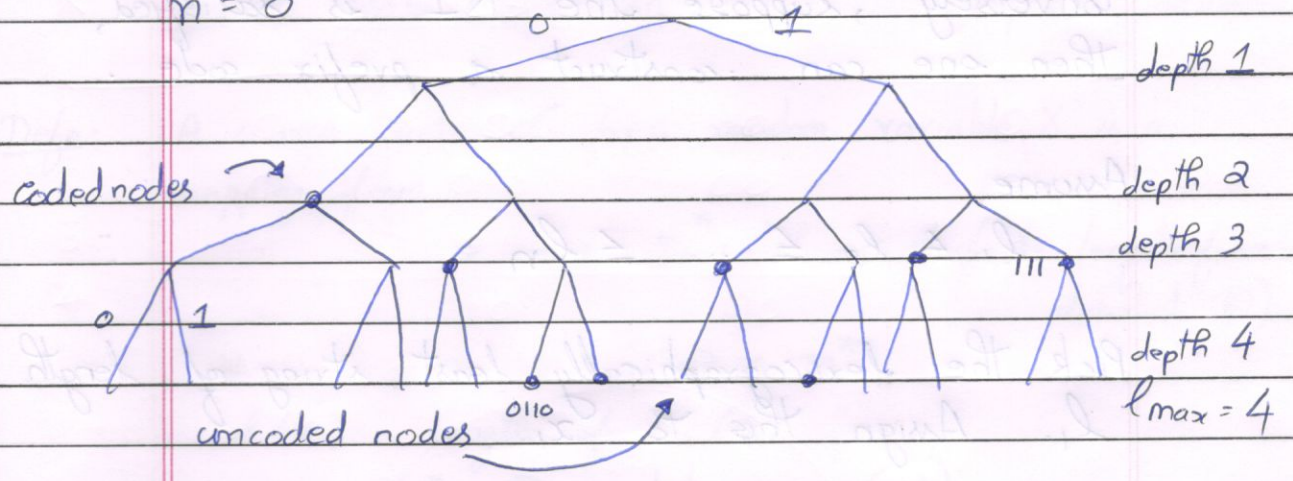
In the tree, we have ancestors and descendants

Let us distinguish between coded and uncoded nodes. A node where a codeword is located is a coded node.

The # of descendants, (coded or uncoded) of a coded node where the code is of length l_i at l_{max} = $D^{l_{max} - l_i}$

$D = 2$

$n = 8$



Claim- No two nodes can have a common descendant (uncoded node)

Pf- Common descendant

$y_1\ y_2\ y_3\ y_4\ y_5\ y_6\ y_7\ y_8$ ← fig A.

Say y_8 is common descendant

But then $\exists$ 2 codeword $y_1y_2y_3y_4$ & $y_1y_2y_3$ (say)

But $y_1y_2y_3$ is prefix of $y_1y_2y_3y_4$.

It can be verified that if the common descendant has two coded nodes as ancestors, then one of the coded nodes must be an ancestor of the other (see figure A)

$$\therefore \sum_{i=1}^{n} D^{l_{max}-l_i} \leq D^{l_{max}} \Rightarrow \boxed{\sum_{i=1}^{n} D^{-l_i} \leq 1}$$

Conversely, suppose the KI is satisfied, then one can construct a prefix code.

Assume,

$$l_1 \leq l_2 \leq \dots \leq l_n$$

Pick the lexicographically least string of length l_1. Assign this to x_1.

$$\therefore D^{-l_1} \leq 1$$

$$\therefore D^{l_{max}-l_1} \leq D^{l_{max}}$$

$\therefore \exists$ at least one uncoded node at depth l_{max} where parent is not a coded node.

Trace the lexicographically least path to the node to depth l_2.

Assign the terminal node at depth l_2 to codeword x_2. Then

$D^{-l_1} + D^{-l_2} < 1$. Continue in this fashion till all codewords are determined.