Information Theory Lecture 5
Kraft Inequality, Lossless Compression

A. Single-letter source codes
B. Extension codes, nonsingular, uniquely decodable, and instantaneous or prefix-free codes
C. Kraft Inequality
D. McMillan Inequality
E. Optimal Codes
F. Huffman Coding Introduction

1

Part 5A: Single-letter source codes

2

Data compression

- Recall AEP compression
  - compress blocks of n symbols to $\approx nH$ bits or $n\log|\mathcal{X}|$ depending on whether $x^n \in A_\varepsilon^{(n)}$.

3

Today's topic

- Today we compress one symbol at a time. (works well)
- Also prove that you can't compress beyond $H(X)$.

4

Source code C

- $X \to C(X)$

  $C(X) \in \mathcal{D}^*$
- $\mathcal{D}^*$ is the set of finite-length strings of D-ary symbols.
- If instead $C(X) \in \mathcal{D}^n$ then the code is fixed-length with every input described by n symbols.

5

Length of code

- $l(x)$ = length of $C(x)$ (# of D-ary symbols)
- $L(C) = \sum_x p(x) l(x) = E_p[l(x)]$

6

Example (D=2)

x	p(x)	C(x)	l(x)
a	1/5	00	2
b	2/5	01	2
c	2/5	1	1

$$L(C) = l(a)p(a) + l(b)p(b) + l(c)p(c)$$

$$= 2\cdot\frac{1}{5} + 2\cdot\frac{2}{5} + 1\cdot\frac{2}{5} = \frac{8}{5} = 1.6$$

$$H(X) = 1.52$$

7

Example (D=2, dyadic pmf)

x	p(x)	C(x)	l(x)=-log(p(x)
a	1/2	1	1
b	1/4	01	2
c	1/8	001	3
d	1/16	0001	4
e	1/16	0000	4

$$l(x) = -\log p(x)$$

$$L(C) = H(X) = 1.875$$

8

Part 5B:
Extension codes, nonsingular,
uniquely decodable,
and
instantaneous or prefix-free codes

Extension Code C^*

- Encode strings of x values in the obvious way

$$C^*(x_1, x_2, ..., x_n) = C(x_1)C(x_2)...C(x_n)$$

- C^n specifies that x's are processed n at a time. C^* doesn't specify.

Nonsingular code and uniquely decodable code

- Nonsingular code
  - $x_i \neq x_j \Rightarrow C(x_i) \neq C(x_j)$
- Uniquely Decodable code
  - C^* is nonsingular.
- Example that is nonsingular but not uniquely decodable:

$C(a)$ is a prefix of $C(b)$.

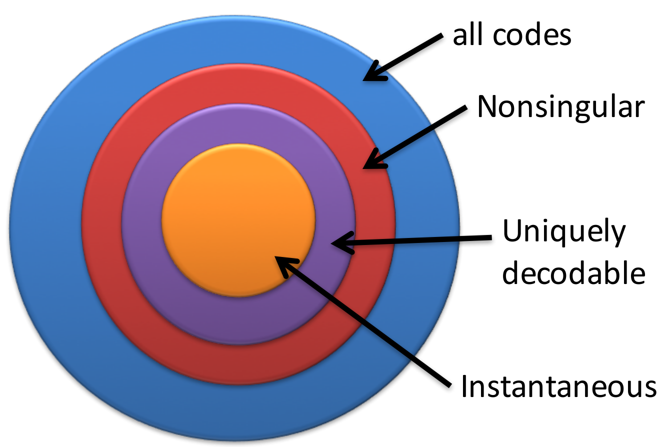
x	C(x)
a	0
b	00
c	1

$$C^*(aa) = 00$$
$$C^*(b) = 00$$

11

Instantaneous or Prefix-Free Codes

- No codeword is a prefix of another codeword.



12

Part 5C: Kraft Inequality

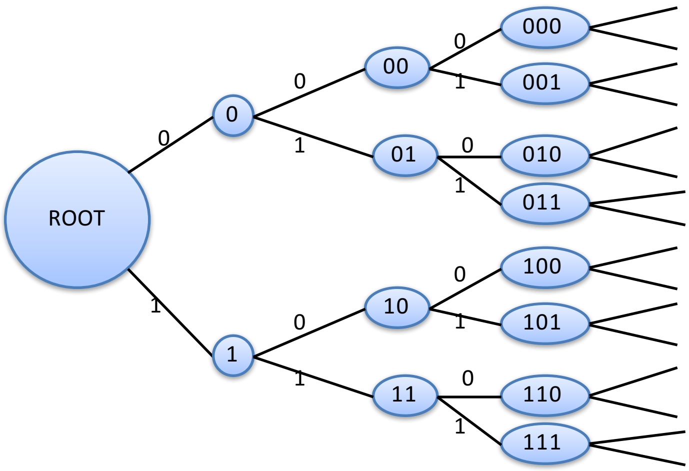
Kraft Inequality

- 1) The codeword lengths of an instantaneous code must satisfy $\sum_i D^{-l_i} \le 1$.
- 2) Given a set of codeword lengths that satisfies $\sum_i D^{-l_i} \le 1$, there is a corresponding instantaneous code.

Let's prove the Kraft inequality…

Tree of codewords

- Consider a tree of possible codewords.



15

Prefix-free codes on the tree

- The prefix-free condition rules out the "children" of any codeword.
- Fix a set of selected codewords. There is a longest codeword, with length $l_{\max}$.
- There are $D^{l_{\max}}$ words of this length.

16

First part of Kraft Inequality

- 1) The codeword lengths of an instantaneous code must satisfy $\sum_i D^{-l_i} \le 1$.

Proof of 1)

- Each valid codeword of length l_i has $D^{l_{\max}-l_i}$ children of length l_{max}.
  - No l_{max} word can be the children of two valid codewords.
  - instantaneous $\Rightarrow \sum_i D^{l_{\max}-l_i} \le D^{l_{\max}}$

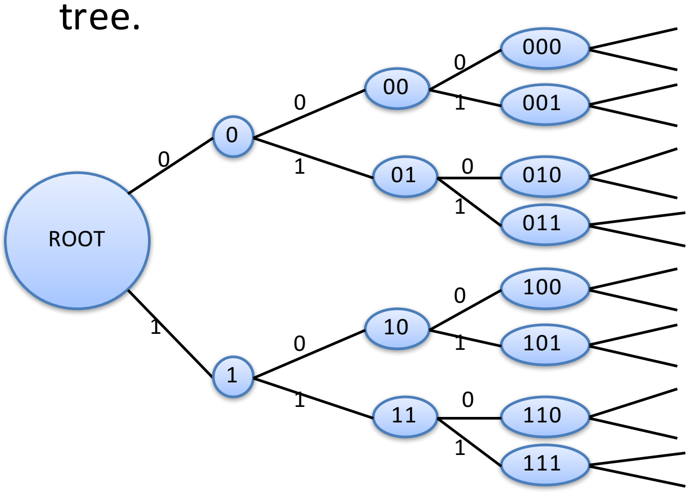
$$\Rightarrow \sum_i D^{-l_i} \le 1$$

Second Part of Kraft Inequality

- 2) Given a set of codeword lengths that satisfies $\sum_i D^{-l_i} \leq 1$, there is a corresponding instantaneous code.

Proof of 2)

- Given $l_1, l_2, \ldots l_n$; if $\sum_i D^{-l_i} \leq 1$ you can always find the prefix free labels by going down the tree.



Part 5D: McMillan Inequality

McMillan Inequality

- All uniquely decodable codes satisfy the Kraft inequality.
- Thus, for any uniquely decodable code, there is an instantaneous code with the same word lengths.

Lemma for proof of McMillan

- Consider C^k $\quad l(x_1, x_2, \ldots, x_k) = \sum_{i=1}^{k} l(x_i)$

$$\boxed{\left(\sum_x D^{-l(x)}\right)^k = \sum_{x^k} D^{-l(x^k)}}$$

  - Example: consider C^2 for two information symbols, a and b

$$\left(\sum_{x\in\{a,b\}} D^{-l(x)}\right)^k = (D^{-l(a)} + D^{-l(b)})^2$$

$$= D^{-(l(a)+l(a))} + D^{-(l(a)+l(b))} + D^{-(l(b)+l(a))} + D^{-(l(b)+l(b))}$$

$$= D^{-(l(a,a))} + D^{-(l(a,b))} + D^{-(l(b,a))} + D^{-(l(b,b))}$$

$$= \sum_{x^k\in\{a,b\}^k} D^{-l(x^k)}$$

Main body of McMillan Proof

$$\left(\sum_x D^{-l(x)}\right)^k = \sum_{x^k} D^{-l(x^k)}$$

$$= \sum_{m=1}^{l_{\max}(C^k)} a(m) D^{-m}$$

$a(m)$ is the number of elements of $\mathcal{X}^k$ that have $l(x) = m$.

$$\leq \sum_{m=1}^{l_{\max}(C^k)} D^m D^{-m}$$

There are at most D^m words of length m for a uniquely decodable code.

$$= l_{\max}(C^k)$$

$$= k l_{\max}(C)$$

Conclusion of McMillan Proof

- $\sum_x D^{-l(x)} \leq (kl_{\max}(C))^{1/k}$ true for all k.
- $\lim_{k\to\infty}(kl_{\max}(C))^{1/k} \to 1$
- Thus $\sum_x D^{-l(x)} \leq 1$ for all uniquely decodable codes.
- So any “good” (i.e. uniquely decodable) code satisfies Kraft.

Part 5E: Optimal Codes (Minimum possible expected code length)

A minimization problem

- Find the minimum $L=E_p[l(x)]$ that satisfies Kraft.

$$\text{minimize} \quad \sum p_i l_i$$
$$\text{Subject to} \quad \sum_i D^{-l_i} \leq 1$$

- This is a convex optimization problem. Let's ignore the requirement that l_i's are integers (lower bounding L).

27

Minimum expected code length

- Using Lagrange Duality (or other solution techniques) optimal l_i's are $l_i = -\log_D p_i$.
- $\sum D^{-l_i} = \sum D^{\log_D p_i} = \sum p_i = 1$
- $\sum p_i l_i = -\sum p_i \log_D p_i = H_D(X)$
- <u>Thm</u>: For any uniquely decodable D-ary code for X, $L_D \geq H_D(X)$.

28

Single letter codes within 1 symbol of $H(X)$

- Set $l_i = \left\lceil \log_D \frac{1}{p_i} \right\rceil$
  - Will this work?
- $$\sum D^{-l_i} = \sum D^{-\lceil \log_D 1/p_i \rceil} \le \sum D^{\log_D p_i} = 1$$
  - Satisfies Kraft

- $\log_D(\frac{1}{p_i}) \le l_i \le \log_D(\frac{1}{p_i}) + 1$
- $H_D(X) \le L \le H_D(X) + 1$
- Our codes is always within one symbol of the entropy.
- This approach achieves entropy when $p_i = D^{-k_i}$ for all i when k_i is an integer.

29

Approaching the entropy rate with blocking.

- We can do even better by blocking n symbols together. $(X_1, \ldots, X_n) = X^n$
- Treat X^n as a single "symbol" and use the same technique:

$$H_D(X^n) \le L^{(n)} \le H_D(X^n) + 1$$

$$\frac{H_D(X^n)}{n} \le \frac{L^{(n)}}{n} \le \frac{H_D(X^n)}{n} + \frac{1}{n} \, .$$

- If $H(\{X_i\})$ exists, $\frac{L^{(n)}}{n} \to H_D(\{X^n\})$.

30

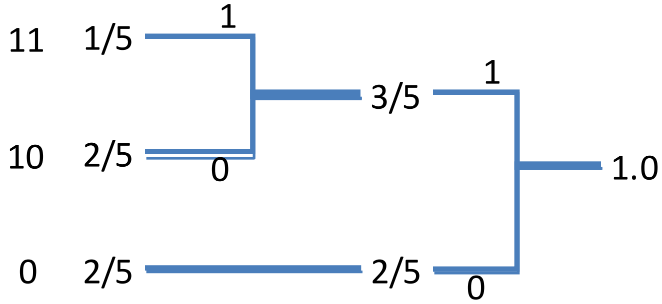
Part 5F:
Huffman Coding Algorithm

Huffman Codes

- An algorithm for constructing the most efficient instantaneous codes.
- Always achieves $L \leq H(X)+1$.
- Sometimes achieves $L = H(X)$.

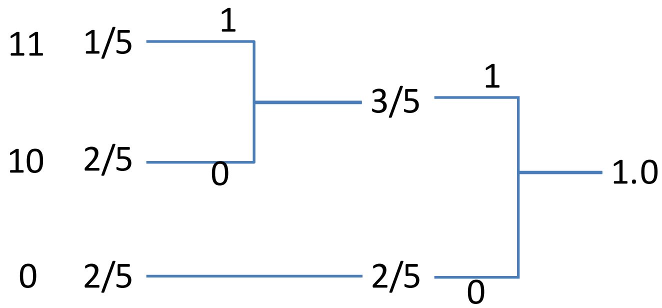
Huffman Coding Example

x	a	b	c
p(x)	1/5	2/5	2/5



Huffman Coding Example

x	a	b	c
p(x)	1/5	2/5	2/5
C(x)	11	10	0



Huffman algorithm

1. Pick the two smallest p(x)'s and draw braches merging them. Label branches with 0 and 1. Extend the other probabilities.

2. Repeat until all probability has merged into one node.

3. Read codeword symbols right to left on the Huffman tree.