LECTURE 5

Last time:

- Stochastic processes
- Markov chains
- Entropy rate
- Random walks on graphs
- Hidden Markov models

Lecture outline

- Codes
- Kraft inequality
- optimal codes.

Reading: Scts. 5.1-5.4.

Codes for random variables

Notation: the concatenation of two strings $\underline{x}$ and $\underline{y}$ is denoted by $\underline{xy}$. The set of all strings over a finite alphabet $\mathcal{D}$ is denoted by $\mathcal{D}^*$. W.l.o.g. assume $\mathcal{D} = 0, 1, \ldots, D-1$ where $D = |\mathcal{D}*|$.

Definition: a source code for a random variable X is a map

$$\begin{aligned} C: \quad & \mathcal{X} \mapsto \mathcal{D}^* \\ & x \rightarrow C(x) \end{aligned}$$

where $C(x)$ is the codeword associated with x

$l(x)$ is the length of $C(x)$

The length of a code C is

$$L(C) = E_X[l(X)]$$

Codes for random variables

C is nonsingular if every element of $\mathcal{X}$ maps onto a different element of $\mathcal{D}^*$

The extension of a code $C : \mathcal{X} \mapsto \mathcal{D}^*$ is the code

$$C^* : \qquad \mathcal{X}^* \mapsto \mathcal{D}^*$$
$$\underline{x}^n \rightarrow C^*(\underline{x}^n) = C(x_1)C(x_2)\ldots C(x_n)$$

A code is uniquely decodable if its extension is nonsingular

A code is instantaneous (or prefix code) iff no codeword of C is a prefix of any other codeword C

Visually: construct a tree whose leaves are codewords

Kraft inequality

Any instantaneous code C with code lengths $l_1, l_2, \ldots, l_m$ must satisfy

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

Conversely, given lengths $l_1, l_2, \ldots, l_m$ that satisfy the above inequality, there exists an instantaneous code with these codeword lengths

Proof: construct a D-ary tree T (codewords are leaves)

Extend tree T to D-ary tree T' with depth l_{MAX}, total number of leaves is $D^{l_{MAX}}$

Kraft inequality

Each leaf of T' is a descendant of at most one leaf of T

Leaf in T corresponding to codeword $C(i)$ has exactly $D^{l_{MAX}-l_i}$ descendants in T' (1 if $l_i = l_{MAX}$)

Summing over all leaves of T gives

$$\sum_{i=1}^{m} D^{l_{MAX}-l_i} \leq D^{l_{MAX}}$$

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

Kraft inequality

Given lengths $l_1, l_2, \ldots, l_m$ satisfying Kraft's inequality, we can construct a tree by assigning $C(i)$ to first available node at depth $C(i)$

Extended Kraft inequality

Kraft inequality holds for all countably infinite set of codewords

Let $n(y_1 y_2 \ldots y_{l_i})$ be the real $\sum_{j=1}^{l_i} y_j D^{-j}$ associated with the i^{th} codeword

Why are the $n(y_1 y_2 \ldots y_{l_i})$s for different codewords different?

By the same reasoning, all intervals

$$\left(n(y_1 y_2 \ldots y_{l_i}), n(y_1 y_2 \ldots y_{l_i}) + \frac{1}{D^{l_i}}\right)$$

are disjoint

since these intervals are all in (0, 1), the sum of their lengths is ≤ 1

For converse, reorder indices in increasing order and assign intervals as we walk along the unit interval

Optimal codes

Optimal code is defined as code with smallest possible $C(L)$ with respect to P_X

Optimization:

minimize $\sum_{x \in \mathcal{X}} P_X(x) l(x)$

subject to $\sum_{x \in \mathcal{X}} D^{-l(x)} \leq 1$

and $l(x)$s are integers

Optimal codes

Let us relax the second constraint and replace the first with equality to obtain a lower bound

$J = \sum_{x \in \mathcal{X}} P_X(x) l(x) + \lambda \left(\sum_{x \in \mathcal{X}} D^{-l(x)} - 1 \right)$

use Lagrange multipliers and set $\frac{\partial J}{\partial l(i)} = 0$

$$P_X(i) - \lambda \log(D) D^{-l(i)} = 0$$

equivalently $D^{-l(i)} = \frac{P_X(i)}{\lambda \log(D)}$

using Kraft inequality (now relaxed to equality) yields

$$1 = \sum_{i \in \mathcal{X}} D^{-l(x)} = \sum_{i \in \mathcal{X}} \frac{P_X(i)}{\lambda \log(D)}$$

so $\lambda = \frac{1}{\log(D)}$, yielding $l(i) = -\log_D(P_X(i))$

Optimal codes

Thus a bound on the optimal code length is

$$-\sum_{i \in \mathcal{X}} P_X(i) \log_D(P_X(i)) = H_D(X)$$

This is lower bound, equality holds iff P_X is D-adic, $P_X(i) = D^{-l(i)}$ for integer $l(i)$

Optimal codes

The optimal codelength L^* satisfies

$$H_D(X) \leq L^* \leq H_D(X) + 1$$

Upper bound: take $l(i) = \lceil \log_D(P_X(i)) \rceil$

$$\sum_{i \in \mathcal{X}} D^{\lceil -\log_D(P_X(i)) \rceil} \leq \sum P_X(i) = 1$$

thus these lengths satisfy Kraft's inequality and we can create a prefix-free code with these lengths

$$\begin{aligned} L^* &\leq \sum_{i \in \mathcal{X}} P_X(i) \lceil -\log_D(P_X(i)) \rceil \\ &\leq \sum_{i \in \mathcal{X}} P_X(i)(-\log_D(P_X(i)) + 1) \\ &= H_D(X) + 1 \end{aligned}$$

We call these types of codes Shannon codes

Optimal codes

Is this as tight as it gets?

Consider coding several symbols together

$C : \mathcal{X}^n \mapsto \mathcal{D}^*$

expected codeword length is $\sum_{\underline{x}^n \in \mathcal{X}^n} P_{\underline{X}^n}(\underline{x}^n) l(\underline{x}^n)$

optimum satisfies

$H_D(\underline{X}^n) \leq L^* \leq H_D(\underline{X}^n) + 1$

per symbol codeword length is

$\frac{H_D(\underline{X}^n)}{n} \leq \frac{L^*}{n} \leq \frac{H_D(\underline{X}^n)}{n} + \frac{1}{n}$

6.441 Information Theory
Spring 2010