# CFCS

## Codes

Miles Osborne (based upon: Frank Keller)

School of Informatics
University of Edinburgh
keller@inf.ed.ac.uk

February 29, 2008

# Coding

Suppose we want to speak to someone on a telephone:

1. Our voice is converted into a sequence of 1s and 0s.
2. These 1s and 0s are transmitted over a network.
3. At the end, the 1s and 0s are reconstructed into our voice.

Step 1 is called *encoding* and step 3 is called *decoding*.

# Coding

A good coding and decoding scheme should have the following properties:

- It should allow us to always recover (most of?) the original data.
  - It needs to be lossless.
  - Mobile phones use lossy encoding.
- It should be efficient.
  - More efficient coding / decoding saves on money / space etc.

# Source Codes

## Definition: Source Code

A source code $C$ for a random variable $X$ is a mapping from $x \in X$ to $\{0,1\}^*$. Let $C(x)$ denote the code word for $x$ and $l(x)$ denote the length of $C(x)$.

Here, $\{0,1\}^*$ is the set of all finite binary strings (we will only consider binary codes).

## Definition: Expected Length

The expected length $L(C)$ of a source code $C(x)$ for a random variable with the probability distribution $P(x)$ is:

$$L(C) = \sum_{x \in X} P(x) l(x)$$

# Source Codes

## Example

Let $X$ be a random variable with the following distribution and code word assignment:

| $x$ | a | b | c | d |
|---|---|---|---|---|
| $P(x)$ | $\frac{1}{2}$ | $\frac{1}{4}$ | $\frac{1}{8}$ | $\frac{1}{8}$ |
| $C(x)$ | 0 | 10 | 110 | 111 |

The average code length of $X$ is:

$$L(C) = \sum_{x \in X} P(x) l(x) = \frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 3 + \frac{1}{8} \cdot 3 = 1.75$$

# Source Codes

What about compressing data?

- Suppose some event $x$ has a probability of being seen $P(x)$.
- Intuitively, the length of the associated code word $l(x)$ should be proportional to this probability:
  - Highly probable events should have short code words.
  - Infrequent events should have longer code words.

# Source Codes

How can we code according to $P(x)$?

- We can think of coding as a game of 20 questions.
- Each question has a yes/no answer: is binary.
- A series of questions is then a series of yes/no answers (bits).

| Event | Questions | Num (q) | $2^{-\text{(Num q)}}$ |
|---|---|---|---|
| a | Yes | 1 | 1/2 |
| b | No Yes | 2 | 1/4 |
| c | No No | 2 | 1/4 |

This uses whole bits.

# Source Codes

If we could use fractional bits:

- Each event would be encoded in $-\log P(x)$ bits.
- The average code length is now:

$$L(C) = \sum_{x \in X} P(x) \cdot -logP(x)$$

This is the entropy of a distribution.

# Properties of Codes

What about ensuring we can always recover the data?

- We need to make sure each code word uniquely corresponds with some event.
- We also want to be able to send multiple code words together:

# Properties of Codes

**Definition: Non-singular Code**

A code is called non-singular if every $x \in X$ maps into a different string in $\{0, 1\}^*$.

- If a code is non-singular, then we can transmit a value of $X$ unambiguously.
- However, what happens if we want to transmit several values of $X$ in a row?
- We could use a special symbol to separate the code words.
- However, this is not an efficient use of the special symbol; instead use *self-punctuating* codes (prefix codes).

# Properties of Codes

### Definition: Extension

The extension $C^*$ of a code $C$ is:

$$C^*(x_1x_2 \ldots x_n) = C(x_1)C(x_2) \ldots C(x_n)$$

where $C(x_1)C(x_2) \ldots C(x_n)$ indicates the concatenation of the corresponding code words.

### Definition: Uniquely Decodable

A code is called uniquely decodable if its extension is non-singular.

- If the code is uniquely decodable, then for each string there is only one source string that produced it;
- However, we have to look at the whole string to do the decoding.

# Prefix Codes

**Definition: Prefix Code**

A code is called a prefix code (instantaneous code) if no code word is a prefix of another code word.

- The coding scheme implicit within $-\log P(x)$ is a prefix code.
- We don't have to wait for the whole string to be able to decode it; the end of a code word can be recognized instantaneously.

# Coding Examples

## Example

The following table illustrates the different classes of codes:

| $x$ | Singular | Non-singular, not uniq. decodable | Uniq. decodable, not instant. | Instant. |
|---|---|---|---|---|
| a | 0 | 0 | 10 | 0 |
| b | 0 | 010 | 00 | 10 |
| c | 0 | 01 | 11 | 110 |
| d | 0 | 10 | 110 | 111 |

## Summary

- Coding / decoding is motivated by efficient communication.
- Prefix codes connect probabilities with compression.
- Not all coding schemes are prefix codes.
- The entropy of a distribution has a coding interpretation.