Run-Length Coding

(Chapter 4)

Chapter Overview

This chapter is the first that deals with Markov sources. After this chapter you should:

- be able to compute the stationary probability of a given two-state Markov source.
- have understood several algorithms, to the point that:
  - you are able to execute by hand a (compression or decompression) algorithm on a given input.
  - you are able to describe the (compression or decompression) algorithm in your own words, in a reasonably precise manner.
  - you are able to understand and explain, why these algorithms perform well. Your explanation and understanding should convey the intuition in a reasonably precise manner.
  - you should know examples of the use of these algorithms in software products, including an explanation of why they are effective in these application areas.
- The algorithm you should have understood is:
  - Run-Length Encoding, particularly the specialised versions of the ITU-T T.4 fax compression standard.

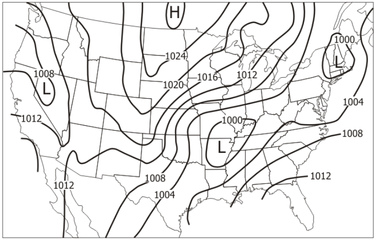
Run-length Encoding (RLE)

- Model: *runs* of symbols are frequent.
  - a "run" of a symbol means the input contains *successive* repetitions of the same symbol.
  - E.g. "aaaaaaaaabbbbcccccccaaaaaabbbbbbbbbbbbbaaaa"
- RLE Basic idea: replace "aaaaabbbbbbb" by "5a7b."
- Memoryless model not suitable. Need a kind of Markov model, called *Capon* model[1].

[1] Jack Capon, "A Probabilistic Model for Run-Length Coding of Pictures," *IRE Transactions on Information Theory*, **5** (1957), pp. 157-163.

Capon Model

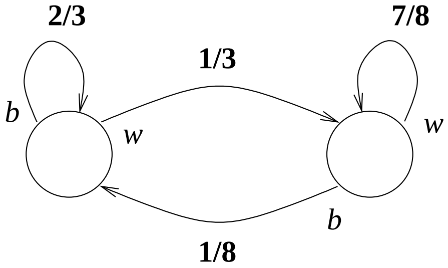
- Invented for compressing weather map data (black/white pictures).



- Pixels are actually only black or white – 1 bit per pixel. *Not* grayscale pictures.

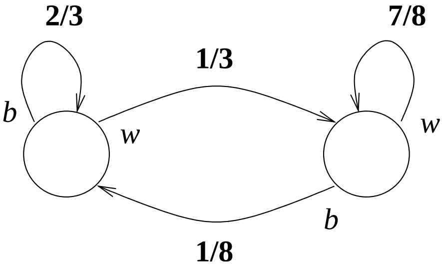
- Two-state Markov model, alphabet $\{b, w\}$ (black/white pixels).
- States called q_b, q_w.

Capon Model: Notation



- $\Pr(q_w)$, $\Pr(q_b)$ — stationary probability of being in q_w or q_b.
  - Abbreviate $p_w = \Pr(q_w)$, $p_b = \Pr(q_b)$.
  - Recall: $H(S) = p_w \times H(q_w) + p_b \times H(q_b)$.
- $p_{ww} = \Pr(q_w \to q_w)$, $p_{bb} = \Pr(q_b \to q_b)$.
  In this example $p_{ww} = 7/8$, $p_{bb} = 2/3$.

Calculating Stationary Probabilities in Capon Models

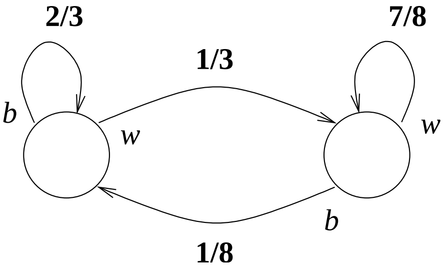


Calculating p_b and p_w:

$$\begin{aligned} p_b + p_w &= 1 \\ (2/3)p_b + (1/8)p_w &= p_b \end{aligned}$$

▷ Solve $p_b = 3/11$, $p_w = 8/11$.

Capon Model: Parameters



Measured parameters[2]:

	Weather Map	Printed Text
p_w	0.887	0.935
p_b	0.113	0.065
$\Pr[w \to b]$	0.027	0.024
$\Pr[b \to w]$	0.214	0.347

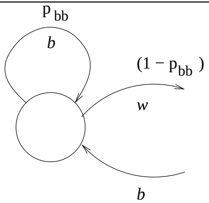
[2] Murat Kunt, "Comparaison de techniques d'encodage pour la réduction de redondance d'images facsimile à deux niveaux", PhD Thesis, EPFL, 1974.

Coding Capon Models

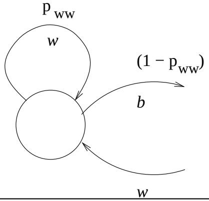
- Two-symbol alphabet – runs alternate.
  - ▷ $wwwbbwwwwbbwwwwwbwwb \equiv w\colon\ 3, 2, 4, 2, 5, 1, 2, 1.$
- How do we code the integers?
  - w and b runs have different distributions.
  - $\Pr[w \text{ run of length } i] = ?$
  - $\Pr[b \text{ run of length } i] = ?$

Coding Capon Models

$$\Pr[\text{black run of length } i] = p_{bb}^{i-1}(1-p_{bb})$$



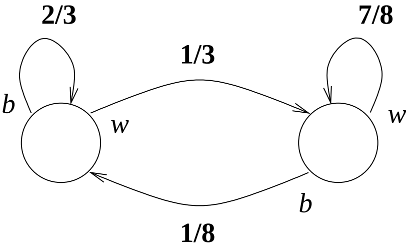
$$\Pr[\text{white run of length } i] = p_{ww}^{i-1}(1-p_{ww})$$



Golomb coding is close to optimal.

- Runs of w: $M = \lceil \frac{1}{1-p_{ww}} \rceil$.
- Runs of b: $M = \lceil \frac{1}{1-p_{bb}} \rceil$.

Example



Pr[black run of length i] $= (2/3)^{i-1} \cdot 1/3$
Pr[white run of length i] $= (7/8)^{i-1} \cdot 1/8$

▷ Most probable run length?

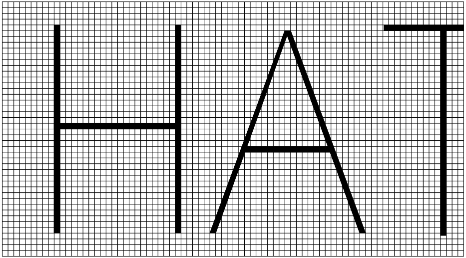
Golomb parameters:

- Runs of w: $M = \lceil \frac{1}{1-7/8} \rceil = 8$.
- Runs of b: $M = \lceil \frac{1}{1-2/3} \rceil = 3$.

RLE case study: Fax Compression

A fax image: 1700×2200 black/white pixels.

- Long runs of white pixels, short runs of black;
- Successive rows of pixels are often quite similar.



ITU-T T.4 standard

Symbols for:

- black and white runs of length $0, 1, \ldots, 63$;
  - B0, B1, ..., B63, W0, W1, ..., W63.
- black and white runs of length $64, 128, 192, \ldots, 1728$.
  - B64, B128, ..., B1728, W64, W128, ..., W1728.

▷ Otherwise too many symbols.

- Black runs are shorter, and most commonly 2-3 pixels. Thus B2 and B3 are common symbols.
- Capon model not really suitable, hence Golomb coding is not the best choice.
- Some others, e.g. EOL (end-of-line).

539W = 512W + 27W

"Remainder" codes

White Run	Code	Black Run	Code
0	00110101	0	0000110111
1	000111	1	010
2	0111	2	11
3	1000	3	10
4	1011	4	011
5	1100	5	0011
....			
45	000001000	45	000101010101
....			

- Black codes and white codes are independent: they can be prefixes of each other why?

“Remainder” codes

White Run	Code	Black Run	Code
0	00110101	0	0000110111
1	000111	1	010
2	0111	2	11
3	1000	3	10
4	1011	4	011
5	1100	5	0011
....			
45	000001000	45	0000101010101
....			

- Black codes and white codes are independent: they can be prefixes of each other why?
  - Because black and white runs alternate, it is “safe”. This keeps codes shorter.

A Run of Length 0?

Runs of length 0 don't exist, so why code for them?

- Codes for black and white runs alternate.
- Sometimes, "alternation" is broken e.g.

$$539W = 512W + 27W$$

- Decoder is aware of this: if it sees a code for a run that is for a multiple of 64, then it knows that the next code is for a run of the same colour. (*)
- What if a run in the raw image was (by chance) a multiple of 64? E.g.

128W, 6B, 10W	**10010 0010 00111**
128W + 12W, 2B, ...	**10010 001000 11 1...**

- This will confuse the decoder. So we write:

$$512W = 512W + 0W$$

- This allows rule (*) to remain valid.