ELEC S347F Multimedia Technologies,
Spring Term, 2021
Tutorial 01 Solution

Question 1: Consider the following sequence of samples:

X: {1,1,3,3,3,3,5,5,5,5,5,5,7,7,7,7,7,7,7,7}

Y: {1,3,4,5,3,4,6,8,9,7,9,11,12}

Z: {1,2,4,4,5,7,7,8,9,10,12,12,14}

Determine which of the above sequence is best modelled by Run-Length Encoding (RLE), Differential Pulse Code Modulation (DPCM) and Linear Prediction (LP), respectively

X: RLE: {(1,2),(3,4),(5,6),(7,8)}

Y: DPCM

DPCM (x_i=x_{i-1}): {1, 2, 1, 1, -2, 1, 2, 2, 1, -2, 2, 2, 1} (only 3 distinct symbols after DPCM)

P(1)=6/13, P(2)=5/13, P(-2)=2/13, Entropy = 1.460 bits/symbol

LP (x_i=i): {0, 1, 1, 1, -2, -2, -1, 0, 0, -3, -2, -1, -1} (5 distinct symbols after LP modelling)

P(0)=3/13, P(1)=3/13, P(-2)=3/13, P(-1)=3/13, P(-3)=1/13, Entropy = 2.237 bits/symbol

Z: LP

LP (x_i=i): {0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1} (only 2 distinct symbols after LP modelling)

P(0)=9/13, P(1)4/13, Entropy = 0.890 bits/symbol

DPCM (x_i=x_{i-1}): {1, 1, 2, 0, 1, 2, 0, 1, 1, 1, 2, 0, 2} (3 distinct symbols after DPCM)

P(1)=6/13, P(2)=4/13, P(0)=3/13, Entropy = 1.526 bits/symbol

Question 2: Find the self-information associated the below events.

(a) $p(A) = 0.125$

(b) $p(A) = 0.25$

(c) $p(A) = 0.5$

(d) $p(A) = 1$

(a) $i(A) = \log_2(1/p(A)) = -\log_2 p(A) = -\log_2(0.125) = -\log_2(2^{-3}) = 3$ bits of information

(b) $i(A) = -\log_2(0.25) = -\log_2(2^{-2}) = 2$ bits of information

(c) $i(A) = -\log_2(0.5) = -\log_2(2^{-1}) = 1$ bit of information

(d) $i(A) = -\log_2(1) = -\log_2(2^0) = 0$ bits of information

Comment: the lower the probability $p(.)$, the higher the associated self-information $i(.)$

Question 3: Find the entropy of the set of symbols A, B, C, D with the below occurrence probabilities respectively:

(a) $p(A) = 0.25$, $p(B) = 0.25$, $p(C) = 0.25$, $p(D) = 0.25$

(b) $p(A) = 1.00$, $p(B) = 0.00$, $p(C) = 0.00$, $p(D) = 0.00$

(c) $p(A) = 0.25$, $p(B) = 0.25$, $p(C) = 0.50$, $p(D) = 0.00$
(d) $p(A) = 0.125$, $p(B) = 0.125$, $p(C) = 0.25$, $p(D) = 0.50$

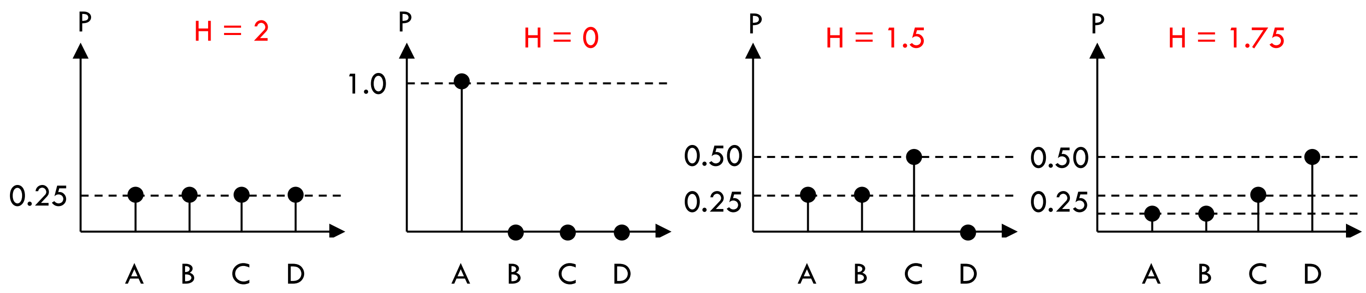
(a) H = p(A)i(A) + p(B)i(B) + p(C)i(C) + p(D)i(D)
H = (0.25)(2) + (0.25)(2) + (0.25)(2) + (0.25)(2)
H = 2 bits/symbol

(b) H = $-1\log_2(1)$ = 0 bits/symbol

(c) H = $-0.25\log_2(0.25) \times 2 - 0.5\log_2(0.5)$ = 1.5 bits/symbol

(d) H = $-0.125\log_2(0.125) \times 2 - 0.25\log_2(0.25) - 0.5\log_2(0.5)$ = 1.75 bits/symbol

(a) (b) (c) (d)



Comments:
Distribution (a) has the highest uncertainty (since events A, B, C and D have equal chance to occur). As a result, its entropy is the higher than (b), (c) and (d).
Distribution (b) has the lowest uncertainty (since event A always happen while B, C and D never happen). As a result, its entropy is the lowest.
For distribution (d), some events have a bit higher chance than others. So its uncertainty is lower than (a) (but higher than (b)). As a result, its entropy is in between that of (a) and (b).
For distribution (c), D has no chance to happen. So its uncertainty is lower than that of (d).

Question 4: Suppose you have a fair dice (i.e. the probabilities of getting “1”, “2”, “3”, “4”, “5” and “6” are all 1/6).

(a) What is the self-information associated with getting a “1” in a toss?
(b) What is the self-information associated with getting a “6” in a toss?
(c) What is the self-information associated with getting a “1” in the first toss, followed by getting a “6” in the second toss?
(d) What is the Entropy for the die?
(e) Suppose the dice is unfair with probabilities of getting “1”, “2”, “3”, “4”, “5” and “6” are 0.2, 0.2, 0.2, 0.2, 0.1 and 0.1, respectively. What are the answers of (a) – (d)?

(a) i(dice=1) = $-\log_2 P$(dice=1) = $-\log_2(1/6)$ = 2.585 bits

(b) same as (a) since P(dice=6) = P(dice=1)

(c) 2.585 bits + 2.585 bits = 5.170 bits

(d) Since all events have the same probability H = i(dice=1) = 2.585 bits

(e) i(dice=1) = $-\log_2 P$(dice=1) = $-\log_2(0.2)$ = 2.322 bits

(f) i(dice=6) = $-\log_2 P$(dice=6) = $-\log_2(0.1)$ = 3.322 bits

(g) 2.322 bits + 3.322 bits = 5.644 bits

(h) H = - P(dice=1) i(dice=1) - P(dice=2) i(dice=2) - P(dice=3) i(dice=3) - P(dice=4) i(dice=4) - P(dice=5) i(dice=5) - P(dice=6) i(dice=6) = 4(0.2)(2.322) + 2(0.1)(3.322) = 1.8576 + 0.6644 = 2.522 bits

Comments: the fair dice has a higher uncertainty than the unfair dice (all faces have equal chances to appear while for the unfair dice some faces have lower chance to appear. The outcome of the unfair dice is more predictable), the entropy (weighted average self-information) of the fair dice is larger than that of the unfair dice.

Question 5: Use the Sardinas-Patterson algorithm to determine which of the below codes are uniquely decodable. In addition, for those uniquely decodable codes determine which of them are instantaneous codes.

(a) {0, 01, 11, 111}
(b) {0, 10, 110, 111}
(c) {10, 00, 001, 101}

(a) Start with the list of codewords {0, 01, 11, 111}

0 is a prefix to 01: generating a dangling suffix of 1 ✓

11 is a prefix to 111: also generating a dangling suffix of 1 ✓

Augment the codeword list with the dangling suffix {0, 01, 11, 111, 1}

1 is a prefix to 11: generating a dangling suffix of 1 ✓

1 is a prefix to 111: generating a dangling suffix of 11 ✗

Iteration	Found Dangling Suffix	New Dangling Suffix	Same as Codeword?
1	1	1	None
2	1, 11	11	Yes, “11” is a codeword

As 11 is a codeword, the code is not uniquely decodable.

(b) Start with the list of codewords {0, 10, 110, 111}

Iteration	Found Dangling Suffix	New Dangling Suffix	Same as Codeword?
1	Nil	N/A	N/A

No codeword is a prefix of any other codeword. Therefore, this code is uniquely decodable.

It is instantaneous since it is a prefix code (no codeword is a prefix of other codewords)

(c) Start with the list of codewords {10, 00, 001, 101}

10 is a prefix to 101: dangling suffix = 1 ✓

00 is a prefix to 001: dangling suffix = 1 ✓

{10, 00, 001, 101, 1}

1 is a prefix to 10: dangling suffix of 0 ✓

1 is a prefix to 101: dangling suffix of 01 ✓

{10, 00, 001, 101, 1, 0, 01}

0 is a prefix to 00: dangling suffix of 0 ✓

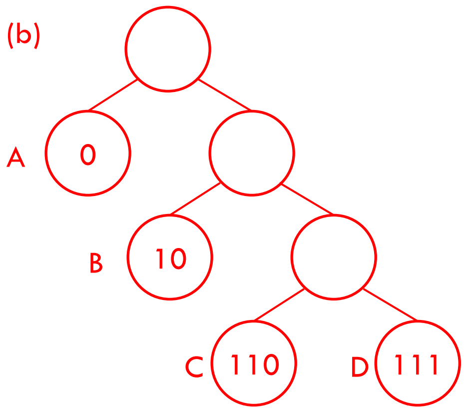
0 is a prefix to 01: dangling suffix of 1 ✓

0 is a prefix to 001: dangling suffix of 01 ✓

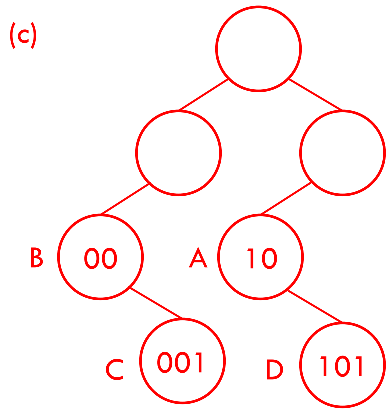
Iteration	Found Dangling Suffix	New Dangling Suffix	Same as Codeword?
1	1	1	None
2	0, 01	0, 01	None
3	0, 1, 01	Nil	N/A

At this point we get no new dangling suffixes therefore this code is uniquely decodable.

It is non-instantaneous since not prefix-free (10 is a prefix to 101, 00 is a prefix to 001).



As seen from the tree diagram, no codeword is the prefix of other codewords. So this code is a prefix code. Thus it is instantaneous (and also uniquely decodable)



As seen from the tree diagram, codeword A is the prefix of D, B is the prefix of C. So this code is a non-prefix code. Thus, it is not instantaneous (but uniquely decodable as proved using the Sardinas-Patterson algorithm) (e.g. 0010.. may be decoded as 00-10.. or 001-00 .., depending on the remaining bit sequence of the encoded string, 101.. may be decoded as 10-1.. or 10-.. depending on the remaining bit sequence of the encoded string)