Compressione dell'informazione / 23-04

$V_x = \{x_1, \dots, x_n\}$ $|V_x| = n$

$C(x)$ codifica binaria per x

$C.\ V_x \to \{0,1\}^+$ $C^+.\ V_x^+ \to \{0,1\}^+$

stringa binaria

es. a, b, c, d $p(a) = \frac{1}{2}$ $p(b) = \frac{1}{4}$ $p(c) = p(d) = \frac{1}{8}$

$c(a) = 1000$

$c(b) = 0100$ $L_C(x) = 4 \quad \forall x$ lunghezza sequenze

$c(c) = 0010$ $H(X) = \frac{1}{2} + \frac{1}{4} + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 6 = \frac{14}{8} = 1{,}75$

$c(d) = 0001$ $\cdot \log_2 2$

4 simboli rappresento con 2 simboli

Decifrabilità univoca

$\forall x, y \in V_x^+ : x \neq y \to C^+(x) \neq C^+(y)$ iniettiva

se a simboli diversi sono associate codifiche diverse

C_1

$c(a) = 1000$ poiché sono tutte della stessa lunghezza e

$c(b) = 0100$ sono diverse, allora sono univocamente

$c(c) = 0010$ decifrabili per qualsiasi parola compongono

$c(d) = 0001$

C_2

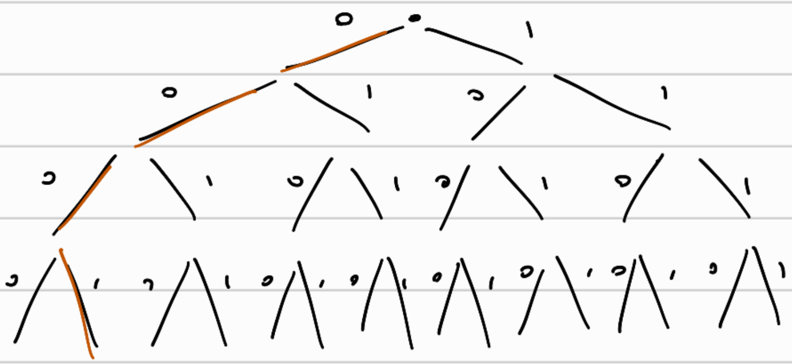
$\frac{1}{2}$ $c(a) = 1$ $\qquad$ $c^+(ab) = c(a)c(b) = 10$ $\qquad$ no univoc. dec.

$\frac{1}{4}$ $c(b) = 0$ $\qquad$ $c^+(c) = 10$

$\frac{1}{8}$ $c(c) = 10$

$\frac{1}{8}$ $c(d) = 01$

Istantaneità (regola del prefisso)



ROOT

1000

la scansione di una codifica è un albero binario (con tante foglie non toccate)

Ottengo "istantaneamente" le sequenze espresse

So quando finisce la sequenza

Univocità $\overset{?}{\rightarrow}$ Istantaneità 1) NO

Istantaneità $\overset{?}{\rightarrow}$ Univocità 2)

1)

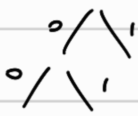
ca 1 $\qquad$ è univoco e non è istantaneo perché ottengo 1

cb 10 $\qquad$ non so ancora se è b, se ho 10 non so ancora se

cc 100 $\qquad$ ho c, " per d, devo attendere quello successivo

cd 1000

2) se posso costruire un albero a cui giungo alle foglie (concatenazione dei simboli) ho una definizione univoca



00 è identificata in un solo modo

$$L(C, V_x) = \sum_{x \in V_x} p(x) \cdot L_c(x)$$

lunghezza attesa
di una codifica

mi aspetto
di utilizzare 2 bit
(< 2 perché non equipr) x scrivere
4 caratteri

$L(C_1, V_x) = 4$ perché sono tutte lunghe 4

$L(C_2, V_x) = \frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 1 + \frac{1}{8} \cdot 2 + \frac{1}{8} \cdot 2 = \frac{5}{4} = 1{,}25 < H(x) = 1{,}75$

C_3
$c(a) = 1$
$c(b) = 01$
$c(c) = 000$
$c(d) = 001$

$L(C_3, V_x) = \frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 3 + \frac{1}{8} \cdot 3 = \frac{14}{8} = 1{,}75$

↳ poiché a si scrive con un bit, la media delle lungh si abbassa

se ho tale uguaglianza ottengo una compressione più efficiente (risparmio dei bit, albero più corto)

$x_1, \dots, x_{|V_x|}$ $\quad l_1, \dots, l_{|V_x|}$

la disuguaglianza di Kraft-McMillan dice che se univocamente decifrabile allora è vera

$$\sum_{i=1}^{|V_x|} 2^{-l_i} \le 1 = \sum_{i=1}^{V} \frac{1}{2^{l_i}} \le 1$$

(e se vale questa allora ∃ cod univoc. ma non è detto che lo siano tutte!)

se > 1 sto 'sprecando', posso ottenere qualcosa di meglio come codifica (L minore)

$$(A)^n = \left(\sum_i 2^{-l_i}\right)^n$$ se > 1 cresce, se < decresce

$$= \sum_{l_1} \sum_{l_2} \dots \sum_{l_n} \underbrace{2^{-l_{i_1}} \cdot 2^{-l_{i_2}} \dots 2^{-l_{i_n}}}$$

pensando $l_1 \le l_2 \le \dots \le l_{|V_x|}$

$n\, l_1$ esp più piccolo
$n\, l_{|V_x|}$ " " grande
} in modulo

$$L \to = \sum_{L=nL_1}^{nL_{|V_x|}} 2^{-L} \underbrace{V_L}$$ → nº di stringhe di lunghezza L

$L_{i_1} + L_{i_2} \ldots + L_{i_n} = L$

$V_L \le 2^L$ ↳ max nº di stringhe generate

$$\le \sum_{L=nL_1}^{nL_m} 1 \le nL_m$$

quindi $A^n \le 1$ altrimenti non vale $\forall n$ e A^n crescerebbe più del termine lineare nL_m

1000
0100
0010
0001

$\frac{1}{16} + \frac{1}{16} + \frac{1}{16} + \frac{1}{16} = \frac{1}{4} < 1$ ok ma spreco

equiprobabili

1
0
10
01

$\frac{1}{2} + \frac{1}{2} + \frac{1}{4} + \frac{1}{4} = 1.5 > 1$ codifica non buona

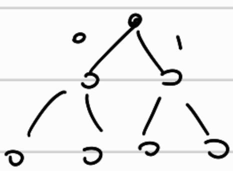
non univoca

con 1, 2, 3, 3 lunghezze

$\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{8} = 1$ codifica perfetta

Immaginando ora che ogni volta divido a metà il valore del padre

↳ "una quantità di miele"



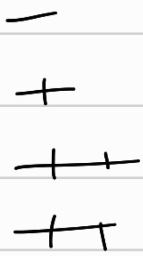
1

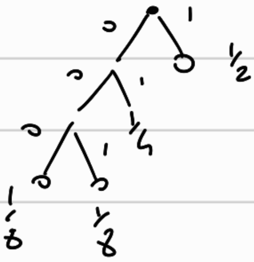
$\frac{1}{2}$ e $\frac{1}{2}$

$\frac{1}{4}, \frac{1}{4}$ e $\frac{1}{4}, \frac{1}{4}$

ma allora nella C_1 precedente $\frac{3}{4}$ di 'quantità' sprecata

1
2
3
3





che è anche istantaneo

1

10

101

100