

Problema generale compressione: data una stringa S contenente

n_1 occorrenze del simbolo s_1

n_2 occorrenze del simbolo s_2

...

n_k occorrenze del simbolo s_k

k è la dimensione dell'alfabeto

$n = n_1 + n_2 + \dots + n_k$ è la lunghezza della stringa

Voglio associare un codice c_i a ogni s_i in modo che la lunghezza totale della codifica sia minima; voglio però che il codice sia univocamente decifrabile.

Il codice:

a 0

b 1

c 00

d 01

e 10

è molto compatto ma non è univocamente decifrabile:

00 potrebbe essere la codifica di "aa" oppure di "c"

TABLE 5.1 Classes of Codes

X	Singular	Nonsingular, But Not Uniquely Decodable	Uniquely Decodable, But Not Instantaneous	Instantaneous
1	0	0	10	0
2	0	010	00	10
3	0	01	11	110
4	0	10	110	111

Example: Morse code. If we see it as a binary code it is not uniquely decodable: in practice it is ternary code with special symbol (pause) after each letter

A _ _	B _ _ _ _	C _ _ _ _	D _ _ _	E _	F _ _ _ _	G _ _ _
H _ _ _ _	I _ _	J _ _ _ _	K _ _ _	L _ _ _ _	M _ _	N _ _
O _ _ _ _	P _ _ _ _	Q _ _ _ _	R _ _ _	S _ _ _	T _	U _ _ _
V _ _ _ _	W _ _ _	X _ _ _ _	Y _ _ _ _	Z _ _ _ _		

The code in the 3rd column is uniquely decodable but not instantaneous:

110000 -> 322

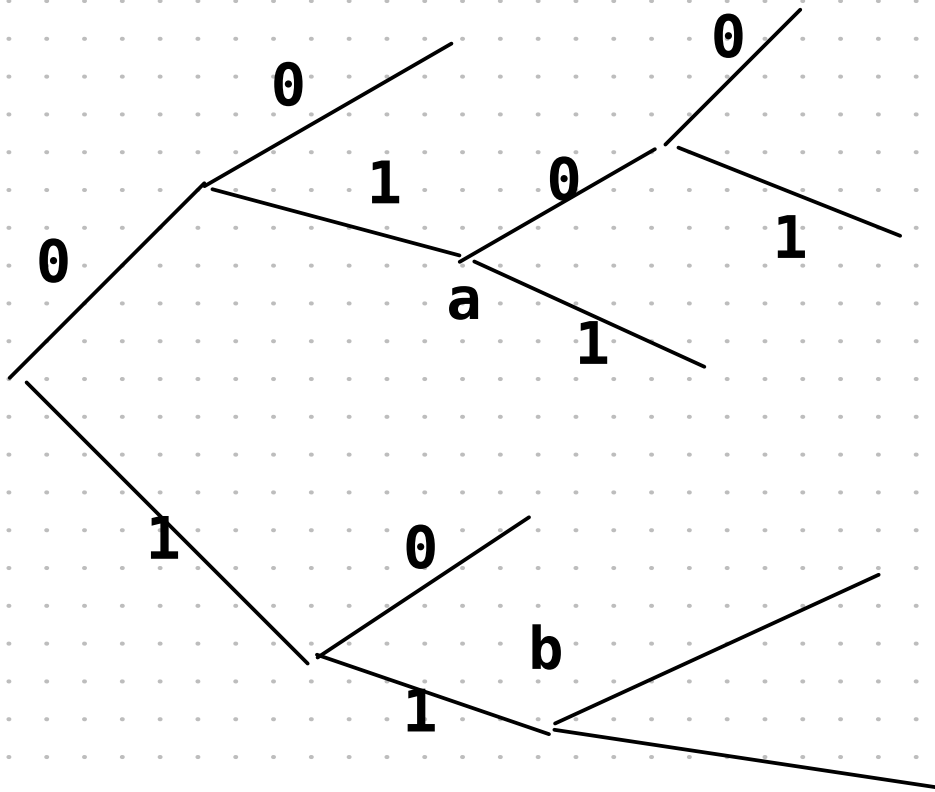
1100000-> 422

we need to count the 0s to figure out if the first symbol is 3 or 4

Ci aiuta a impostare il problema la disuguaglianza di Kraft
Per codeword binarie e codice prefix free deve essere:

$$\sum_{i=1}^k 2^{-|c_i|} \leq 1$$

Idea della dimostrazione: Sia M la dimensione massima di una codeword costruisco albero binario completo di altezza M



associo ad ogni codeword
tutte le foglie che stanno sotto
il nodo che rappresenta
quella codeword, che sono:

$$2^{M-|c_i|}$$

osservo che ogni foglia
appartiene al max a 1 codeword

$$\sum_{i=1}^k 2^{M-|c_i|} \leq 2^M$$

divido per 2^M e ottengo Kraft

Soluzione matematica al problema del codice ottimo

Dati $n_1, \dots, n_k$, $n = \sum_{i=1}^k n_i$

Trovare le lunghezze ottimali $\ell_1, \dots, \ell_k$

che minimizzano la lunghezza della stringa codificata $\sum_{i=1}^k \ell_i n_i$

Cerco un codice prefix free quindi devo rispettare il vincolo della disuguaglianza di Kraft:

$$\sum_{i=1}^k 2^{-\ell_i} \leq 1$$

Ottengo un problema di minimo vincolato sulle variabili $\ell_1, \dots, \ell_k$

$$\min_{\ell_1, \dots, \ell_k} \sum_{i=1}^k \ell_i n_i, \quad \text{con vincolo} \quad \sum_{i=1}^k 2^{-\ell_i} \leq 1$$

Se accetto anche soluzioni con lunghezze non intere posso usare il metodo dei moltiplicatori di Lagrange: il minimo si trova dove si annulla il gradiente di:

$$F(\ell_1, \dots, \ell_k, \lambda) = \sum_i n_i \ell_i + \lambda \left(\sum_i 2^{-\ell_i} - 1 \right)$$

Facendo i conti si ottiene il minimo si ha per

$$\ell_i = -\log_2 \frac{n_i}{n}$$

non so come realizzare codici con lunghezze non intere
ma so che se le lunghezze sono intere non posso fare
meglio del minimo dato dalla formula qui sopra

Questi risultati si generalizzano al caso in cui il numero di possibili simboli è infinito, ad esempio se devo considerare una sequenza di interi positivi:

1 4 2 2 3 10 23 1 2 1

La disuguaglianza di Kraft generalizzata mi dice che

$$\sum_{i=1}^{\infty} 2^{-|c_i|} \leq 1$$

e se l'intero i compare con frequenza n_i/n allora il suo codice dovrebbe avere lunghezza:

$$\ell_i = -\log_2 \frac{n_i}{n}$$

Quando codifichiamo gli interi solitamente le codeword sono date, quindi utilizziamo questo risultato allo "incontrario", diciamo il codice Gamma che codifica l'intero i con $2^{\lceil \log_2 i + 1 \rceil} - 1$ bits è ottimale quando la frequenza di i è

$$2^{-2^{\lceil \log_2 i + 1 \rceil} - 1} \approx 2^{-\log_2 i^2} \approx \frac{1}{i^2}$$