Codifica a blocchi

Un vettore b di k bit di informazione viene codificato in un vettore c lungo n bit che costituisce una parola di codice, con $n > k$. Si ha che c_m è sempre una parola di codice, mentre $\tilde{c_m}$ non è detto che lo sia. Dato $b \to c$, $\mathcal{C}$ l'insieme delle parole di codice, si ha che:

$$P[C] = P[\hat{b} = b] = \sum_{c_i \in \mathcal{C}} P[\hat{b} = b | c = c_i] P[c = c_i]$$

$$\hat{c} = \underbrace{\underset{\alpha \in \mathcal{C}}{\operatorname{argmax}} P[\tilde{c} = \beta | c = \alpha] P[c = \alpha]}_{\text{Criterio MAP (sempre)}} =$$

$$\overset{\text{se } P[c=\alpha]=1/|\mathcal{C}|}{=} \underbrace{\underset{\alpha \in \mathcal{C}}{\operatorname{argmax}} P[\tilde{c} = \beta | c = \alpha]}_{\text{Criterio ML (parole equiprobabili)}}$$

Per non avere accumuli o ritardi deve valere $kT_b = nT_c$, quindi $T_c < T_b$.

Canale BSC

Senza memoria: la probabilità di errore non dipende dai bit inviati precedentemente, quindi la probabilità di ricevere la sequenza 010 è data $p_x(0)p_x(1)p_x(0)$. *Simmetrico*: la probabilità di sbagliare un 1 con uno 0 è uguale alla probabilità di sbagliare uno 0 con un 1, cioè: $P(y = 0|x = 1) = P(y = 1|x = 0)$.

Distanza di Hamming

Si definisce $d_H(a, b) = \#$ di posizioni che hanno valori differenti nelle due sequenze.

$$P[\tilde{c} = \beta | c = \alpha] = P_e^{d_H(\alpha,\beta)} (1 - P_e)^{n - d_H(\alpha,\beta)}$$

$$\hat{c} = \underset{\alpha \in \mathcal{C}}{\operatorname{argmax}} \left(\frac{P_{bit}}{1 - P_{bit}} \right)^{d_H(\alpha,\beta)} \overset{P_{bit} \le 0.5}{=} \underset{\alpha \in \mathcal{C}}{\operatorname{argmin}}\, d_H(\alpha, \beta)$$

Abbiamo definito il Criterio MD, valido se il canale è BSC senza memoria e parole di codice equiprobabili.

Numero di errori rilevabili e correggibili

In un codice a blocco si ha che:

- # max errori sempre rilevabili $= d_{min} - 1$;
- # max errori sempre correggibili $= \lfloor \frac{d_{min}-1}{2} \rfloor$

Bound di Hamming

Sia t il numero di errori che voglio correggere.

$$\underbrace{k/n}_{\text{Rate del codice}} \le 1 - \frac{1}{n} \log_2 \sum_{r=0}^{t} \binom{n}{r}$$

Codice lineare a blocco

Permettono di costruire parole di codice in maniera algebrica. Se $\mathcal{C}$ è un codice lineare a blocco, allora deve valere che $\forall c_i, c_j \in \mathcal{C}$:

$$(c_i + c_j) \in \mathcal{C}, \quad (\beta_i c_i + \beta_j c_j) \in \mathcal{C}, \quad \beta \in \{0, 1\}$$

Norma / Peso di Hamming: $\|x\|_H = \#$ di 1 presenti nella sequenza. In un codice lineare a blocco, il *peso di Hamming del codice* coincide con la distanza minima di Hamming del codice, ovvero $d_{min} = W_H(\mathcal{C})$.

Bound di Singleton

In un codice lineare a blocco si ha che $d_{min} \le n-k+1$.

Rappresentazione matriciale dei codici lineari

La matrice generatrice $\boldsymbol{G}$ mappa blocchi di bit b in ingresso in parole di codice c secondo la regola $\boldsymbol{c = Gb}$. Deve valere: Rango($\boldsymbol{G}$) $= k$.
Per rappresentare tutte le parole di codice non lineare serve una tabella di $n \cdot 2^k$ bit; se il codice è lineare basta una matrice di $n \cdot k$ bit.
Una matrice è in forma sistematica se $\boldsymbol{G} = \begin{bmatrix} \mathbb{I}_k \\ \boldsymbol{A} \end{bmatrix}$. In questo caso si ha che $W_H(\mathcal{C}) =$ peso di Hamming minimo delle colonne.
Matrice controllo di parità: $\boldsymbol{H} = \begin{bmatrix} \boldsymbol{A} & \mathbb{I}_{n-k} \end{bmatrix}$. Deve valere: Rango($\boldsymbol{H}$) $= n - k$ e $\boldsymbol{HG} = \underline{0}$.

Decodifica tramite sindrome

Sia $\boldsymbol{\sigma = H\gamma}$ la sindrome, dove γ è un vettore di n bit; dipende solo dalla sequenza di errore $\boldsymbol{\varepsilon}$, ci sono 2^n possibili sequenze di errore. $\boldsymbol{\sigma} = 0 \Leftrightarrow \gamma \in \mathcal{C}$, altrimenti si cerca il coset relativo (sequenze ε che portano alla stessa sindrome) e si sceglie un elemento con peso di Hamming minimo $\varepsilon_{min}(\boldsymbol{\sigma})$ detto coset leader. $\hat{c} = \gamma + \varepsilon_{min}(\boldsymbol{\sigma})$. Il numero di coset coincide con il numero di sindromi diverse, ovvero $\#\text{coset} = 2^{\#\text{bit sindrome}} = 2^{n-k}$.

Codice di Hamming

Matrice generatrice del codice di Hamming $(7, 4)$:

$$\boldsymbol{G} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \quad \boldsymbol{H} = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Nella matrice $\boldsymbol{H}$ si trovano tutte le sequenze non nulle di $n - k$ bit. Il numero di colonne è $n = 2^{n-k} - 1$. I codici di Hamming soddisfano il bound di Hamming all'uguaglianza e hanno $d_{min} = 3$.

Teoria dell'informazione (pt 1)

Entropia. Data X v.a. discreta l'entropia è

$$H(X) = E_x[i_X(X)] = \sum_{a \in A_x} p_X(a) i_X(a)$$

$$0 \overset{X \text{ deterministica}}{\le} H(X) \overset{X \text{ uniforme}}{\le} \log_2 M \quad M = |A_x|$$

Entropia congiunta. $H(X, Y) = E[i_{X,Y}(X, Y)]$.

$$0 \overset{X,Y \text{ determ.}}{\le} \max \left\{ \underbrace{H(X)}_{Y \text{ func determ. di } X}, \overbrace{H(Y)}^{X \text{ func determ. di } Y} \right\}$$

$$\le H(X, Y) \overset{X,Y \text{ indip}}{\le} H(X) + H(Y)$$

Entropia condizionata.

$$H(X|Y = b) = \sum_a -p_{x|y}(a|b) \log_2 p_{x|y}(a|b)$$

$$H(X|Y) = E_y[H(X|Y = y)] = \sum_b p_Y(b) H(X|Y = b)$$

$$= H(X, Y) - H(Y)$$

$$0 \overset{X,Y \text{ determ.}}{\le} H(X|Y) \overset{X,Y \text{ indip.}}{\le} H(X)$$

Informazione tra due var aleatorie. $I(X;Y) = H(X) + H(Y) - H(X,Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) = I(Y;X)$.

$$0 \overset{X,Y \text{ indip.}}{\le} I(X;Y) \le \min \left\{ \underbrace{H(X)}_{X|Y \text{ determ.}}, \overbrace{H(Y)}^{Y|X \text{ determ.}} \right\}$$

Teoria dell'informazione (pt 2)

Vettore aleatorio. $\underline{x} = [X_1, ..., X_N]$ dove X_n sono var aleatorie discrete, spesso hanno tutte stesso alfabeto. *Densità di probabilità:* $p_{\underline{x}}(\underline{a}) = P(X_1 = a_1, ..., X_N = a_n)$. *Funzione informazione:* $i_{\underline{x}}(\underline{a}) = -\log_2 p_{\underline{x}}(\underline{a})$. *Entropia per simbolo:* $H_s(\underline{x}) = \frac{H(\underline{x})}{N}$ con N lunghezza vettore.

$$0 \leq H_s(\underline{x}) \overset{X_n \text{ i.i.d.}}{\leq} H(X_n) \leq \log_2 M$$

Entropia per simbolo di una sorgente sempre attiva: $H_s(\underline{x}) = \lim_{N\to\infty} \frac{H(\underline{x})}{2N+1}$, se v.a. i.i.d $H_s = H(X_n)$. *Informazione mutua:* $I(\underline{x};\underline{y}) = H(\underline{x}) + H(\underline{y}) - H(\underline{x},\underline{y})$. *Informazione mutua per simbolo:* $I_s(\underline{x};\underline{y}) = H_s(\underline{x}) + H_s(\underline{y}) - H_s(\underline{x},\underline{y})$.

Tasso info messaggio o information rate. Informazione utile che esce da una sorgente, $R(\underline{x}) = \frac{1}{T} H_s(\underline{x})$ dove T periodo di simbolo. *Tasso nominale di informazione:* è tasso massimo di informazione, è dato da $R_N(\underline{x}) = \frac{1}{T}\log_2 M$.

Tasso di info di un canale M-ario $\mathcal{G}$. $R_{\mathcal{G}} = \frac{1}{T} I_s(\underline{x};\underline{y})$, dipende dalla densità di probabilità dell'ingresso. Se il canale è **senza memoria** allora $R_{\mathcal{G}} = \frac{1}{T} I(X_n; Y_n)$,

Capacità di canale

Capacità di un canale M-ario. Definita come il massimo tasso di informazione del canale, scegliendo opportunamente la densità di prob dell'ingresso.

$$C = \max_{\{p_x\}} R(\mathcal{G}) = \max_{\{p_x\}} \frac{I_s(\underline{x};\underline{y})}{T} \overset{\text{no memoria}}{=} \max_{\{p_x\}} \frac{I(X;Y)}{T}$$

Capacità canale binario simmetrico senza memoria. Ottenuta con l'ingresso X uniforme.

$$C = \frac{1}{T_b}\left[1 - \left(P_{bit}\log_2\frac{1}{P_{bit}} + (1-P_{bit})\log_2\frac{1}{1-P_{bit}}\right)\right]$$

Capacità canale con ingresso binario e con erasure. Ottenuta con l'ingresso X uniforme. $C = \frac{1-\alpha}{T}$. Dove α è la probabilità di cancellazione. Per piccoli valori di α, la capacità è maggiore rispetto a quella del canale binario simmetrico senza memoria. **Capacità canale AWGN.** $C = \frac{1}{2T}\log_2(1 + \sigma^2_{S_{Tx}}/\sigma^2_w)$, dove $S_{Tx}(t)$ deve avere una distribuzione gaussiana.

Teorema di Shannon per la codifica di canale

Consideriamo un canale $M-$ario senza memoria con periodo di simbolo T e capacità C. Sia $\{b_l\}$ una sorgente di simboli con tasso di informazione nominale del messaggio R_N. Se $R_N < C$ allora $\forall\delta > 0$ e $\forall n$ suff grande $\exists$ un codice con $\lceil 2^{nRT}\rceil$ parole lunghe n e lettere dell'alfabeto $M-$ario e con probabilità di errore sulla parola che è $P(\hat{c} \neq c) < \delta$.
In un codice binario si ha che $2^k = 2^{nRT} \Rightarrow \frac{k}{n} = RT < CT = \max_{\{p_{\underline{x}}\}} I_s(\underline{x};\underline{y})$.

Codifica di sorgente (pt 1)

L'obiettivo è quello di rendere più compatto il messaggio in uscita dalla sorgente. Il codificatore fa corrispondere a ciascuna parola in $\boldsymbol{\mathcal{D}_x}$ una parola in $\boldsymbol{\mathcal{D}_y}$, dove $\mathcal{D}$ sono detti dizionari. Chiamiamo $\boldsymbol{M_y}$ l'alfabeto delle parole $\underline{y}$. Sia $\boldsymbol{L(\underline{y})}$ la lunghezza della parola $\underline{y}$; definiamo $\boldsymbol{L_y} = \sum_{\underline{a}\in\mathcal{D}_x} p_{\underline{x}}(\underline{a}) L(\underline{y}(\underline{a}))$ la lunghezza media delle parole di codice in uscita dal codificatore.
Codice a prefisso. Un codice a prefisso è un codice di sorgente che non ha nessuna parola di codice che è prefisso di altre parole di codice.
Teorema di Shannon per la codifica di sorgente. Ogni codice di sorgente soddisfa: $L_y \geq \frac{H(\underline{x})}{\log_2 M_y}$. Esiste un codice a prefisso che soddisfa: $L_y < \frac{H(\underline{x})}{\log_2 M_y} + 1$.
Codifica di sorgente ottima. Una codifica di sorgente è ottima se ha L_y più piccola possibile (rispettando sempre il teorema di Shannon). *Teorema:* per un codice a prefisso ottimo, parole di codice a più bassa probabilità hanno lunghezza maggiore. *Teorema:* nel dizionario di un codice a prefisso ottimo, ci sono almeno due parole di codice che hanno lunghezza massima e differiscono per un bit.

Codifica di sorgente (pt 2)

Codice di Shannon-Fano. $l_i = \lceil \log_{1/M_y} p_{\underline{x}}(\underline{a}_i)\rceil$ $\underline{a}_i \in \mathcal{D}_x$. Se le probabilità sono tutte potenze intere di $1/M_y$ allora è ottimo e L_y raggiunge il lower bound del teorema di Shannon.
Procedura di Huffman. Permette di costruire un codice di sorgente binario ottimo a partire dalla $p_{\underline{x}}(\underline{a}_i)$ dell'ingresso. Non garantisce L_y uguale lower bound di Shannon.
Codifica aritmetica - Elias. Il calcolo delle parole di codice è computazionalmente molto meno costoso rispetto ad Huffman. $l_i = \lceil i(\underline{a}_i)\rceil + 1$. La L_y coincide alla lunghezza media del codice di S-F più 1. *Procedura:* divido l'intervallo $[0,1]$ in $|\mathcal{D}_x|$ sotto-intervalli ognuno di lunghezza pari alla probabilità del simbolo associato; trovo il punto medio m_i dell'intervallo; codifico in binario m_i (es: $m_i = 0.11001$); $\underline{y}_i$ = primi l_i bit di m_i (es: $l_i = 4 \Rightarrow \underline{y}_i = 1100$).
Efficienza di un codice di sorgente. $\eta = H(X)/(L_y \log_2 M_y)$, $\eta \in [0,1]$.

Probabilità

Var aleatorie congiunte discrete. Siano(X,Y) v.a. congiunta discreta. *Prob congiunta:* $p_{X,Y}(a,b) = p_X(a)p_{Y|X}(b,a) = p_Y(a)p_{X|Y}(a,b)$. Se X,Y sono indipendenti $p_{X,Y}(a,b) = p_X(a)p_Y(b)$. *Prob marginali:* $p_X(a) = \sum_{b\in A_y} p_{X,Y}(a,b) = \sum_{b\in A_y} p_Y(b)p_{X|Y}(a|b)$, $p_Y(b) = \sum_{a\in A_x} p_{X,Y}(a,b) = \sum_{a\in A_x} p_X(a)p_{Y|X}(b|a)$.
Probabilità di errore media (matrice transizione). $P_{X\neq Y} = \sum_{a\neq b} p_{X,Y}(a,b) = \sum_{a\neq b} p_X(a)p_{Y|X}(b|a)$.

QAM

Dati $\alpha_{n,I}, \alpha_{n,Q} \in \{2l - L - 1, l = 1, ..., L\}$ con $L = \sqrt{M}$ e data la base ortonormale, si ha che il segnale generico nello spazio euclideo è $\underline{s}_n = [\alpha_{n,I}\sqrt{E_h/2}, \alpha_{n,Q}\sqrt{E_h/2}]$. La distanza minima tra due segnali è $\sqrt{2E_h}$. Sia E_h l'energia della base, $E_n = |\underline{\alpha}_n|^2 E_h/2$ l'energia del segnale $n-$esimo, si ha che l'energia media è $E_s = (M-1)E_h/3$. La *probabilità di errore* è data da $P[E] \leq 4(1 - 1/\sqrt{M})Q(\sqrt{E_h/(2\sigma_I^2)}) = 4(1 - 1/\sqrt{M})Q(\sqrt{(3E_s)/[(M-1)2\sigma_I^2]})$. *Probabilità di errore sul Bit:* $P_{bit} \approx P[E]/\log_2 M$ usando la mappatura di Gray per righe e colonne. Una *base ortonormale* è data da $\phi_1 = \sqrt{2/E_h}h_{T_x}(t)\cos(2\pi f_0 t + \phi_0)$, $\phi_2 = -\sqrt{2/E_h}h_{T_x}(t)\sin(2\pi f_0 t + \phi_0)$ ($\cos \perp \sin$).