Capitolo 1

Partizioni su insiemi

Definiamo in questo capitolo alcuni concetti fondamentali: le partizioni su un insieme, il significato di entropia, lo spazio delle partizioni con le operazioni binarie associate e una distanza tra partizioni. Studiamo le proprietà di questa distanza e come amplificarla. Diamo infine qualche esempio sui possibili tipi di partizioni che si possono incontrare su un insieme discreto.

1.1 Generalità

Introduciamo il formalismo e risultati generali per spazi di partizioni e metriche di Rohlin, seguendo l'approccio in [billingsley, casartelli-vivo]. Sia $(\mathbf{M}, \mathcal{M}, \mu)$ uno spazio di probabilità, ovvero un insieme $\mathbf{M}$, una σ-algebra $\mathcal{M}$ di sottoinsiemi di $\mathbf{M}$, una misura normalizzata μ su $\mathcal{M}$. Nei casi trattati $\mathbf{M}$ può essere una sequenza di simboli, un reticolo bidimensionale, un grafo arbitrario.

Introducendo una relazione di equivalenza su$\mathbf{M}$, possiamo definire una particolare classe di sottoinsiemi. Una *partizione* di $\mathbf{M}$ è una collezione finita $\alpha \equiv (A_1, A_2, \ldots, A_N)$ di sottoinsiemi disgiunti misurabili che ricoprono $\mathbf{M}$, cioè $A_i \cap A_j = \emptyset$ se $i \neq j$ e $\bigcup_k A_k = \mathbf{M}$. Gli $\{A_k\}$ sono chiamati *atomi* di α e sono una rappresentano le classi di equivalenza degli elementi di $\mathbf{M}$. L'insieme di tutte le partizioni misurabili è denotato con $\mathcal{Z} \equiv \mathcal{Z}(\mathbf{M})$. La partizione unitaria ν consiste del singolo atomo coincidente con $\mathbf{M}$. È possibile introdurre un ordine parziale su $\mathcal{Z}$, con la relazione $\alpha \leq \beta$ quando β è un raffinamento di α: questo accade quando ogni atomo A_k è esattamente composto da atomi di β, cioè $A_k = \{\bigcup_j \ B_j \mid B_j \in \beta\}$. In questo caso, si dice che α è un *fattore* di di β. La partizione banale $\nu \leq \alpha$, $\forall \alpha$.

I termini come *unità* e *fattore* dipendono dalla definizione di uno pseudo-prodotto commutativo ed associativo, la composizione $\gamma = \alpha \vee \beta$ (o anche $\gamma = \alpha\beta$ ove non vi sia ambiguità). Il *prodotto* è la partizione meno fine di tutte le partizioni con $\gamma \geq \alpha$ e $\gamma \geq \beta$, i cui atomi sono le intersezioni non vuote di tutti gli atomi di α e β. Chiaramente il prodotto con l'unità si comporta come l'identità del prodotto, con $\alpha\nu = \alpha$ per ogni α, mentre $\alpha\eta = \alpha$ quando $\eta \leq \alpha$. Queste proprietà rendono il risultato della composizione una specie di "minimo comune multiplo". Dalla definizione segue anche che $\alpha \vee \alpha = \alpha$, ovvero il prodotto è idempotente.

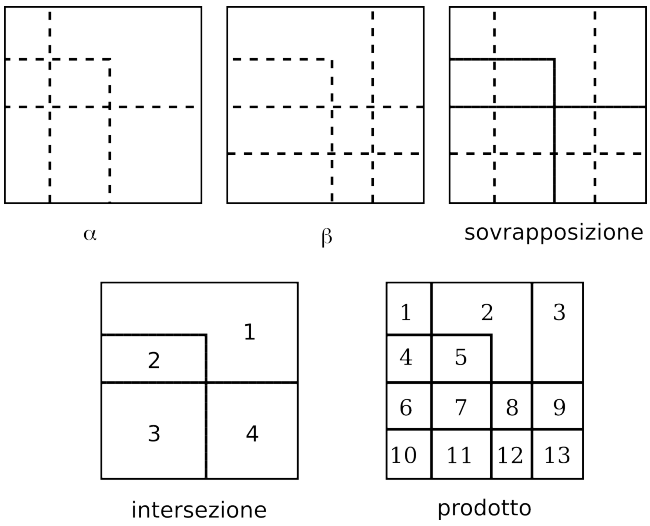

Figura 1.1.2: Intersezione e prodotto come operazioni sui bordi. Le partizioni α e β hanno bordi tratteggiati, con fase complementare, che danno l'impressione di linea continua quando sovrapposti. È evidente in questo modo il tratto di bordi in comune. Gli atomi sono stati numerati per rendere evidente la differenza tra le due operazioni.

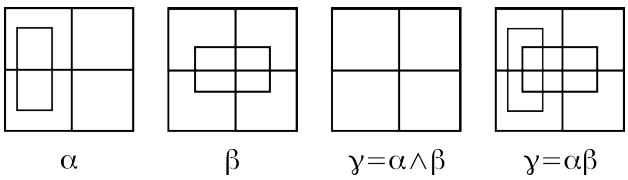

Figura 1.1.1: Esempio di prodotto e intersezione tra due partizioni

L'operazione complementare che implementa un "massimo comune divisore" nello spazio delle partizioni è l'*intersezione* $\sigma = \alpha \wedge \beta$, definita come la partizione più fine tale che $\sigma \leq \alpha$ e $\sigma \leq \beta$. In questo caso, $\alpha \wedge \nu = \nu$ per ogni α, mentre $\alpha \wedge \beta = \nu$ implica che α e β sono *relativamente primi*, cioè non hanno un fattore comune.

Un altro modo di calcolare e visualizzare le operazioni è in termini di *bordi interni* di atomi della partizione, come si vede in figura 1.1.2. Il prodotto $\alpha \vee \beta$ corrisponde alla partizione avente come bordi l'unione dei bordi di α e β, mentre l'intersezione $\alpha \wedge \beta$ ha come bordi l'intersezione di quelli di α e β. Quando nell'intersezione i bordi non si chiudono, sono cancellati dal risultato e gli atomi raggruppati. Poichè la partizione banale non ha bordi tra atomi, si ricavano immediatamente le sue proprietà nel prodotto e nell'intersezione.

Una partizione può rappresentare un esperimento probabilistico con risultati disgiunti $A_1, \ldots, A_N$, dove l'evento *atomico* A_k ha probabilità $\mu(A_k)$. Un *fattore* è quindi un sottoesperimento dell'esperimento più fine, che raggruppa diversi risultati come equivalenti: ad esempio, "pari o dispari" è un sottoesperimento con due atomi, dell'esperimento $\{1, 2, 3, 4, 5, 6\}$ del lancio di un dado.

Sullo spazio $\mathcal{Z}$ possiamo definire dei funzionali *entropia* $H : \mathcal{Z} \to \mathbb{R}^+$, definiti su ogni partizione. In particolare l'*entropia di Shannon*

$$H_S(\alpha) = -\sum_{i=1}^{n} \mu(A_i) \ln \mu(A_i) \tag{1.1.1}$$

L'entropia di Shannon è una misura dell'informazione media ottenuta dall'esperimento. Si vede immediatamente che la partizione banale ν, non codificando alcuna informazione ha entropia nulla in entrambi i casi. Se $\beta = (B_1, \ldots, B_n)$ è un'altra partizione, l'entropia condizionata di α rispetto a β è

$$H_S(\alpha|\beta) = -\sum_{i=1}^{n}\sum_{k=1}^{m} \mu(A_i \cap B_k) \ln \frac{\mu(A_i \cap B_k)}{\mu(B_k)} = H_S(\alpha \vee \beta) - H_S(\beta) \tag{1.1.2}$$

dove si prende per convenzione che $x \ln x = 0$ se $x = 0$. L'entropia condizionata è l'informazione media residua ottenuta da α quando il risultato per β è noto. Si noti che l'entropia di Shannon dipende solo dalla distribuzione delle misure degli atomi, non dalla loro natura o "forma", che potrebbe non avere significato in spazi astratti. Le mutue relazioni tra atomi (e possibilmente le loro forme) al contrario influenzano direttamente l'entropia condizionale.

Definiamo una metrica sullo spazio delle partizioni $\mathcal{Z}(\mathbf{M})$ tramite la distanza di Rohlin d_{R}

$$d_{\mathrm{R}} = H(\alpha|\beta) + H(\beta|\alpha)$$

che misura la complessiva non-similarità tra le partizioni α e β. È possibile dare una definizione alternativa di questa distanza, sfruttando la seconda scrittura della probabilità condizionale, riscrivendo d_{R} come

$$d_{\mathrm{R}} = 2H(\alpha \vee \beta) - H(\alpha) - H(\beta) \tag{1.1.3}$$

La simmetria $d_{\mathrm{R}}(\alpha, \beta) = d_{\mathrm{R}}(\beta, \alpha)$, la positività e la condizione $d_{\mathrm{R}}(\alpha, \alpha) = 0$ sono manifeste, mentre la disuguaglianza triangolare è soddisfatta se H soddisfa alle condizioni di un funzionale entropia.

Se $\mathbf{M}$ è finito, una *configurazione* o *stato* $\mathbf{a}$ su $\mathbf{M}$ è una funzione che assegna ad ogni punto $x_i \in \mathbf{M}$ un valore $a_i = f(x_i)$ nell'alfabeto $\mathbb{K}$. Tutte le possibili configurazioni formano uno spazio $\mathcal{C} \equiv \mathcal{C}(\mathbf{M})$. Su $\mathcal{C}$ la distanza di Hamming è definita come

$$d_{\mathrm{H}}(\mathbf{a}, \mathbf{b}) = \mathcal{N} \sum_i \rho(a_i, b_i)$$

dove $\rho(a_i, b_i)$ è una distanza su $\mathbb{K}$ e $\mathcal{N}$ una possibile costante di rinormalizzazione, che noi porremo uguale a 1. Da notare come questa misura operi tra elementi diversi dalla misura di Rohlin, pur partendo sempre da $\mathbf{M}$.

Entropia topologica

Oltre all'entropia di Shannon, definita tramite la misura μ, è possibile una definizione alternativa che non ve ne fa ricorso. Per questo motivo, è chiamata *entropia topologica.* Dato uno spazio compatto, come $\mathbf{M}$ che stiamo considerando, esiste sempre un ricoprimento finito tramite insiemi aperti. In particolare,

$\exists \delta \in \mathbb{N}^+$ numero minimo di aperti con cui generare tale ricoprimento. La partizione di uno spazio è ricoprimento *minimale*[1], motivo per cui il numero di atomi della partizione satura la disuguaglianza del teorema, $\delta = n$. Una volta stabilito l'esistenza di δ e avendone calcolato il risultato, poniamo l'entropia topologica pari al logaritmo naturale di δ

$$H_T(\alpha) = \ln(\delta) = \ln(n) \tag{1.1.4}$$

Nei casi in cui non vi sia una naturale metrica sullo spazio $\mathbf{M}$, la definizione topologica può essere utile per cercare una misura più intrinseca dell'informazione contenuta nella partizione. Tuttavia, non esiste una definizione di entropia condizionale per H_T e per definire la distanza di Rohlin utilizziamo la sua definizione in termini di prodotto tra partizioni.

Si vede come H_T è un buon funzionale entropia, infatti se $\alpha < \beta$, allora $H_T(\alpha) < H_T(\beta)$, in quanto una partizione strettamente più fine ha banalmente un numero maggiore di atomi. Da questa condizione si ha che per il prodotto

$$\alpha \vee \beta \geq \alpha \implies H_T(\alpha \vee \beta) \geq H_T(\alpha)$$

per α, β generici e la distanza di Rohlin risulta definita positiva.

1.2 Riduzione

L'essenziale dissimilirатà tra due partizioni potrebbe essere confusa ed indebolita dalla presenza di un fattore comune ampio, come ad esempio accade se gli atomi della partizione hanno lunghezza media molto breve, nel qual caso la maggioranza dei confini risulta essere la stessa. Si cerca quindi di eliminare fattori comuni il più possibile, con una *riduzione* che ci si aspetta aumenti la distanza relativa. Tuttavia questa operazione, analoga alla riduzione in minimi termini per frazioni, non è unicamente definita, in quanto le partizioni, a differenza degli interi, non ammettono una univoca fattorizzazione in fattori primi. Il ruolo dei fattori primi (ovvero fattori irriducibili) è giocato dalle sottopartizioni *dicotomiche,* che sono tuttavia ancora estremamente ridondanti ($2^{n-1} - 1$ per partizioni con n atomi).

A partire dalla partizione $\alpha \equiv (A_1, \ldots, A_n)$, definiamo quindi una famiglia ristretta $\mathbf{E}(\alpha)$ di *fattori dicotomici elementari* $\tilde{\alpha}_1, \tilde{\alpha}_2, \ldots, \tilde{\alpha}_n$ con le seguenti caratteristiche:

1. $\mathbf{E}(\alpha)$ deve essere ben definita per ogni $\alpha \in \mathcal{Z}$
2. $\mathbf{E}(\alpha)$ non deve contenere più di n (il numero di atomi in α) fattori elementari
3. $\vee_{k=1}^{n} \tilde{a}_k = \alpha$

Una scelta universale coniste nel prendere come fattori dicotomici $\tilde{\alpha}_k \equiv (A_k, A_k^c)$ le partizioni formate dai singoli atomi e dai loro complementi in $\mathbf{M}$. Fattori di questo tipo sono chiamati *universali semplici.*

Una volta che per due partizioni α e β le famiglie di fattori dicotomici $\mathbf{E}(\alpha)$ e $\mathbf{E}(\beta)$ sono state definite, abbiamo diversi possibili processi di riduzione.

[1] Nulla vieta di avere altri ricoprimenti perfettamente ridondanti, ma sempre con un numero finito di aperti

Definizione 1. Riduzione tramite fattore comune

1. Si definisce il massimo fattore comune $\sigma = \alpha \wedge \beta$
2. Si tralasciano da $\mathbf{E}(\alpha)$ e $\mathbf{E}(\beta)$ i fattori che non sono relativamenti primi con σ, e indichiamo i fattori rimanenti come $\hat{\alpha}_k$ e $\hat{\beta}_k$ rispettivamente. Questo vuol dire che $\hat{\alpha}_k \wedge \sigma = \hat{\beta}_j \wedge \sigma = \nu$.

Definizione 2. Riduzione con eliminazione atomi in comune

1. Si tralasciano da $\mathbf{E}(\alpha)$ e $\mathbf{E}(\beta)$ i fattori che compaiono in entrambe le partizioni. Se indichiamo i fattori rimanenti come $\hat{\alpha}_k$ e $\hat{\beta}_k$ rispettivamente, questo vuol dire che $\forall \hat{\alpha}_k$, $\nexists \beta_j \in \beta | \hat{\alpha}_k = \beta_j$ e viceversa.

Definizione 3. Riduzione con eliminazione fattori simili

1. Si tralasciano da $\mathbf{E}(\alpha)$ e $\mathbf{E}(\beta)$ i fattori che hanno un corrispondente "simile" nell'altra partizione. Questo vuol dire che scartiamo da $\hat{\alpha}$ il fattore

$$\alpha_k \text{ se } \exists\ \beta_j \in \beta, \text{ tale che } \mu(\alpha_k \triangle \beta_j) \leq \epsilon$$

   e viceversa da $\hat{\beta}$ il fattore

$$\beta_k \text{ se } \exists\ \alpha_j \in \alpha, \text{ tale che } \mu(\beta_k \triangle \alpha_j) \leq \epsilon$$

   Il simbolo $\alpha_k \triangle \beta_j$ indica la differenza simmetrica tra i due atomi, ovvero i siti che appartengono ad un atomo ma non all'altro.

Alla fine, per tutti i tipi di riduzione, definiamo le partizioni ridotte come $\hat{\alpha} = \vee_k \hat{\alpha}_k$ e $\hat{\beta} = \vee_k \hat{\beta}_k$, ovvero il prodotto dei fattori dicotomici "sopravvissuti". Nel capitolo sugli algoritmi presenteremo metodi ottimali per il calcolo dei fattori dicotomici per ogni criterio presentato, che presentano una notevole complessità se eseguiti nel modo naïve.

Per il resto della sezione concentreremo la nostra attenzione sulla riduzione tramite fattore comune massimo. Si motiva la scelta del confronto con il fattore comune poichè vi sono casi in cui le partizioni non hanno atomi in comune, ma ciononostante si ha che $\sigma \neq \nu$. Questo accade, per esempio, quando $\alpha < \beta$ strettamente e non vi sono fattori comuni elementari. In questo caso allora $\sigma = \alpha$ e $\hat{\alpha} = \nu$ con questo metodo di riduzione, mentre $\hat{\alpha} = \alpha$ tralasciando i fattori comuni. Può capitare inoltre che anche se le partizioni sono già ridotte, non sono prime tra di loro. In particolare, $H(\sigma)$ è una misura di similarità tra le partizioni.

1.2.1 Amplificazione

Il processo di riduzione porta alla definizione di partizioni con complessità possibilmente inferiore, ovvero $H(\hat{\alpha}) \leq H(\alpha)$. Questo va nel verso opposto quando si considera l'effetto sulla distanza, che invece aumenta.

Il rapporto di *amplificazione* R misura quanto la riduzione ha messo in risalto la differenza tra partizioni. Ne dimostriamo la caratteristica fondamentale:

$$\mathrm{R} = \frac{\mathrm{d_R}(\hat{\alpha}, \hat{\beta})}{\mathrm{d_R}(\alpha, \beta)} \geq 1$$

Proposizione. $d_{\mathrm{R}}(\hat{\alpha},\hat{\beta}) \geq d_{\mathrm{R}}(\alpha,\beta)$

Dimostrazione. Ricordando che $\sigma = \alpha \wedge \beta$ possiamo scrivere $\alpha = \sigma\hat{\alpha}$ e $\beta = \sigma\hat{\beta}$: infatti σ contiene tutti i fattori tralasciati durante la riduzione. Utilizzando ora l'idempotenza del prodotto, $\sigma = \sigma\sigma$, possiamo riscrivere la tesi utilizzando l'equazione (1.1.3)

$$2\,H(\sigma\hat{\alpha}\hat{\beta}) - H(\sigma\hat{\alpha}) - H(\sigma\hat{\beta}) \leq 2\,H(\hat{\alpha}\hat{\beta}) - H(\hat{\alpha}) - H(\hat{\beta})$$

scambiando l'ordine dei termini si ottiene

$$2\,H(\sigma\hat{\alpha}\hat{\beta}) - 2\,H(\hat{\alpha}\hat{\beta}) \leq H(\sigma\hat{\alpha}) - H(\hat{\alpha}) + H(\sigma\hat{\beta}) - H(\hat{\beta})$$

e sfruttando la formula (1.1.2) per l'entropia condizionata, la tesi si riduce a

$$2\,H(\sigma|\hat{\alpha}\hat{\beta}) \leq H(\sigma|\hat{\alpha}) + H(\sigma|\hat{\beta})$$

ma questo è chiaramente vero, in quanto

$$H(\sigma|\hat{\alpha}\hat{\beta}) \leq H(\sigma|\hat{\alpha}) \quad \text{e} \quad H(\sigma|\hat{\alpha}\hat{\beta}) \leq H(\sigma|\hat{\beta})$$

per le proprietà dell'entropia, poichè il termine condizionante è sicuramente maggiore, ovvero

$$\hat{\alpha}\hat{\beta} \geq \hat{\alpha} \text{ e } \hat{\alpha}\hat{\beta} \geq \hat{\beta}$$

□

Da notare che la dimostrazione vale per la distanza di Rohlin definita sia tramite H_S che H_T.

Risulta importante la scelta della famiglia di fattori dicotomici $\mathbf{E}(\alpha)$, che è dettata dalla topologia e geometria dello spazio delle configurazioni. La scelta della famiglia di fattori dicotomici universali semplici è sempre possibile, poichè la determinazione di A_k^c a partire da A_k è un'operazione ben definita in qualunque spazio topologico. Prendere come fattori elementari la parte interna di contorni di cluster ad esempio, richiede un concetto di orientabilità e la possibilità di definire contorni, ovvero insiemi con codimensione 1 su varietà – mentre vorremmo estendere l'analisi anche a grafi generici, privi di strutture geometriche predefinite. Già nel caso lineare è possibile prendere fattori dicotomici diversi e algoritmicamente più performanti, a patto di restringere lo studio alle partizioni con atomi formati da cluster connessi.

1.3 Possibili tipi di partizionamento

L'applicabilità dei metodi discussi è assolutamente generica, estendibile a qualunque spazio di probabilità finito si voglia considerare, vediamo dunque di dare esempi dei possibili spazi $\mathbf{M}$ su cui abbiamo lavorato, con i relativi fattori dicotomici e consequenze computazionali.

Essendo lo studio svolto su calcolatore, lo spazio $\mathbf{M}$ e la sua $\sigma-$algebra sono finiti e discreti. I siti appartenenti ad $\mathbf{M}$ possono essere sempre numerati ordinati in modo opportuno x_i, $i \in \{1,\ldots,L\}$, dove L è il numero totale di siti, che si tratti di un reticolo o di un grafo. Poichè la partizione induce una relazione

di equivalenza sull'insieme $\mathbf{M}$, indichiamo che due siti sono equivivalenti se appartengono allo stesso atomo

$$i \sim j \iff \exists A_k \text{ tale che } i \in A_k,\ j \in A_k$$

Per partizionare i siti in atomi disguinti, richiediamo che la *configurazione* (o *stato*) $\mathcal{C}$, associ ad ogni sito una lettera dell'alfabeto $\mathbb{K}$, considerato finito,$|\mathbb{K}| < \infty$. Il simbolo associato al sito i-esimo, non essendovi ambiguità, sarà d'ora in poi indicato con $f(i)$. Se lo stato del sistema è descritto con variabili continue (o vi è un numero enorme di possibili lettere nell'alfabeto, si pensi ad una variabile a 64 bit rappresentante un numero reale), si può sempre ridurre l'alfabeto ad un insieme $\{k_j\}$ discretizzando i valori della configurazione con il criterio:

$$f(i) := \bar{k} \quad \text{tale che} \quad |f(i - \bar{k})| = \min_{k_j \in \mathbb{K}} |f(i) - k_j|$$

oppure mettendo nello stesso atomo siti vicini che hanno valori distanti meno di ϵ:

$$i \sim j \iff |f(i) - f(j)| \leq \epsilon$$

1.3.1 Sequenze lineari connesse

Consideriamo sequenze lunghe L, provenienti da due casi:

- Problemi di meccanica statistica, in cui la configurazione è una variabile aleatoria, generata algoritmicamente da una catena di Markov in base al modello di Ising monodimensionale, nel qual caso l'alfabeto corrisponde a $\{-1, +1\}$.
- Sequenze di origine biologica, in particolare sequenze di amminoacidi (proteine), in cui $|\mathbb{K}| = 22$.

Lo studio delle sequenze è solitamente svolto con la distanza di Hamming d_H che tuttavia è molto sensibile a variazioni puntuali dei valori in $\mathcal{C}$. I siti che compongono la sequenze non si influenzano, una variazione su un sito può solo variare di $\{-1, 0, +1\}$ la distanza totale.

Ad ogni configurazione possiamo associare una partizione in $\mathcal{Z}$, in cui gli atomi sono formati dai cluster, cioè sottoinsiemi connessi di $\mathbf{M}$ a valori omogenei in $\mathbb{K}$. Questo stabilisce una mappa $\Phi : \mathcal{C} \to \mathcal{Z}$ da ogni configurazione ad una partizione corrispondente, rendendo possibile il confronto tra $d_{\mathrm{H}}(\mathbf{a}, \mathbf{b})$ in $\mathcal{C}$ e $d_{\mathrm{R}}(\alpha, \beta)$ in $\mathcal{Z}$, dove $\alpha = \Phi(\mathbf{a})$ e $\beta = \Phi(\mathbf{b})$. La relazione è chiaramente del tipo molti-a-uno, infatti assegnando ad un segmento di lettere omogeneo in $\mathcal{C}$ un diverso simbolo, non cambia la partizione corrispondente. In simboli, esprimiamo la relazione come

$$i \sim j \iff j = i + 1,\ f(i) = f(j)$$

È evidente quindi che variazioni locali, ad esempio il cambiamento di un singolo simbolo, possono non modificare affatto la partizione

$$\{\ldots, T, T, C, A, A, \ldots\} \overset{\Phi}{\equiv} \{\ldots, T, T, M, A, A, \ldots\}$$

che presenta sì una perdita di informazione, ma permette quindi anche di filtrare molto "rumore" e si è dimostrata una ottima scelta nel caso biologico e una

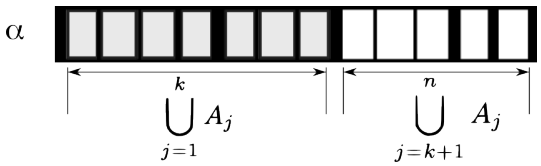

Figura 1.3.1: Fattori dicotomici lineari

necessità nello studio di sequenze di Ising – la hamiltoniana del sistema non distingue i valori associati ai singoli siti, ma solo differenze tra siti vicini. La suddivisione in partizioni si comporta nello stesso modo, estraendo quindi solo informazioni fisicamente rilevanti.

Esempio. Supponiamo di aver partizionato una stringa come {A,A,A,B,B,C,C,D,D,D} o $\{+,+,+,-,-,+,+,-,-,-\}$, aventi la stessa partizione

$$\alpha = \{(1,2,3),(4,5),(6,7),(8,9,10)\}$$

il calcolo esplicito dell'entropia è il seguente:

$$\begin{aligned} A_1 &= (1,2,3) & \mu(A_1) &= \frac{3}{10} \\ A_2 &= (4,5) & \mu(A_2) &= \frac{2}{10} \\ A_3 &= (6,7) & \mu(A_3) &= \frac{2}{10} \\ A_4 &= (8,9,10) & \mu(A_4) &= \frac{3}{10} \end{aligned}$$

Ora, $H_{\mathrm{S}}(\alpha) = -2\,(0.3)\ln(0.3) - 2\,(0.2)\ln(0.2) \simeq 1.36$, mentre $H_{\mathrm{T}}(\alpha) = \ln(4) \simeq 1.38$. Come si vede i valori sono abbastanza simili.

Fattori dicotomici nel caso lineare connesso

Per una partizione $\alpha \equiv (A_1, \ldots, A_n)$, definiamo i fattori dicotomici $\tilde{\alpha}_1, \tilde{\alpha}_2, \ldots, \tilde{\alpha}_n \in \mathbf{E}(\alpha)$ nel modo seguente:

$$\tilde{\alpha}_k = \left\{ \bigcup_{j=1}^{k} A_j,\ \bigcup_{j=k+1}^{n} A_j \right\}$$

dove ricordiamo che n è il numero di atomi. In termini di siti quindi, il fattore dicotomico k-esimo corrisponde alla partizione con tutti i siti corrispondenti ai primi k atomi presi insieme, ed il complementare corrispondente ai siti a partire dall'atomo $(k+1)$-esimo fino ad L, come mostrato in figura 1.3.1.

Con questa particolare scelta, resa possibile dalla topologia connessa e ordinata, poichè abbiamo l'ordine ereditato da $\mathbb{N}$, il processo di riduzione è estremamente semplificato ed efficiente. Partendo da una partizione α, la partizione ridotta $\hat{\alpha}$ è quella in cui sono stati rimossi i bordi in comune tra α e σ, come si vede in figura 1.3.2

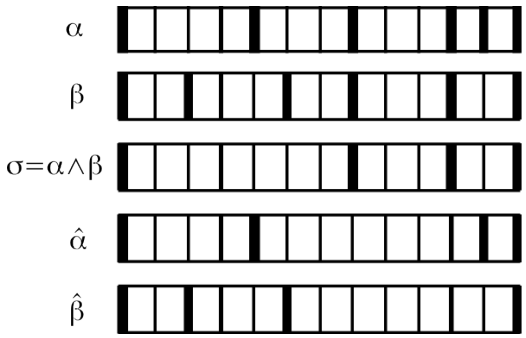

Figura 1.3.2: Riduzione nel caso di sequenze lineari. La riga spessa indica la separazione tra atomi. Si nota come i bordi della partizione comune siano quelli comuni sia ad α che a β, mentre nelle partizioni ridotte non compaiono più i bordi esistenti anche in σ

1.3.2 Sequenze non connesse

Lo studio delle sequenze biologiche, in cui vi è una notevole frammentazione, dovuta alla bassa probabilità di avere molti simboli consecutivi uguali, ha motivato la ricerca di diversi criteri di partizionamento. Il più semplice è considerare come appartenenti allo stesso atomo siti con lo stesso simbolo, non adiacenti ma con la possibilità di saltare al massimo n_s siti in avanti

$$i \sim j \iff |j-i| \leq n_s,\ f(i) = f(j)$$

In questo caso, la topologia nello spazio $\mathcal{Z}$ non è più connessa, per cui l'utilizzo dei fattori dicomici lineari non è possibile. Un'alternativa è quella di prendere per l'atomo $A_k = \{n_{k_1}, n_{k_2}, \ldots, n_{k_n}\}$ una partizione dicotomica costituita da tutti i siti tra il primo e l'ultimo appartenente a A_k e dal complementare, sconnesso, A_k^c

$$\hat{\alpha}_k = \left\{ \bigcup_{i=n_{k_1}}^{n_{k_n}} n_i, \left(\bigcup_{i<n_{k_1}} n_i \right) \bigcup \left(\bigcup_{i>n_{k_n}} n_i \right) \right\}$$

Dati gli scarsi vantaggi di questo approccio, si è scelto tuttavia anche in questo caso di utilizzare i *fattori universali semplici*. Una volta venuta a meno la connessione degli atomi, questo tipo di partizionamento presenta le complessità e caratteristiche di un criterio arbitrario su grafo generico – si veda il prossimo paragrafo per le illustrazioni delle operazioni tra partizioni di questo tipo.

1.3.3 Sequenze su reticoli multidimensionali e grafi

In questo caso, si considera le partizioni in cui fanno parte dello stesso atomo siti con lo stesso simbolo, ma con il vincolo che devono essere *primi vicini* sul reticolo

$$i \sim j \iff d(i,j) = 1,\ f(i) = f(j)$$

dove la distanza indica il numero di passi sul reticolo. Possiamo tuttavia generalizzare arbitrariamente la condizione di vicinanza, per includere secondi vicini, siti con lo stesso simbolo arbitrariamente posti, ecc.

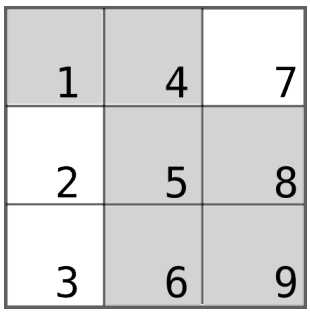

Figura 1.3.3: Mappa in memoria di un reticolo bidimensionale. In grigio e bianco sono evidenziati i valori dalla configurazione, mentre i numeri nell'angolo rappresentano l'ordine dei siti.

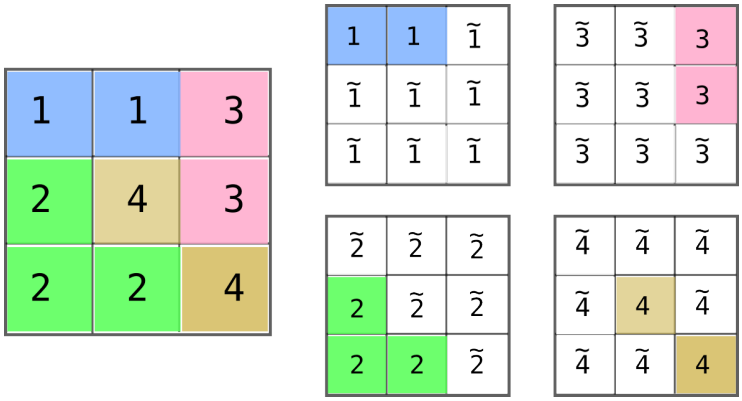

Figura 1.3.4: Partizione con 4 atomi e i 4 fattori dicotomici universali semplici corrispondenti. I numeri indicano l'etichetta corrispondente ad ogni atomo, scelto arbitrariamente.

Illustriamo le possibili operazioni con un reticolo bidimensionale, che tuttavia generalizza immediatamente al grafo arbitrario. In particolare numeriamo anche in questo caso i siti, cosa necessaria per mappare nella memoria di un computer la configurazione. La mappatura in memoria (figura 1.3.3) comporta necessariamente la disposizione di siti "vicini" sul reticolo su posizioni in cui non risultano più contigue in memoria[2]. Nel caso della figura la partizione corrispondente è

$$A = \{(1, 4, 5, 6, 8, 9), (2, 4), (7)\}$$

evidentemente non-connessa. In questo caso i fattori dicotomici anche risultano sconnessi, come nell'esempio di un reticolo 3x3 in figura 1.3.4, in cui si è preso come atomi i siti con lo stesso valore indipendentemente dalla loro posizione.

[2]Ricordiamo la convenzione "per colonne" di rappresentazione di una matrice bidimensionale nella maggior parte dei linguaggi di programmazione, a partire dal Fortran in poi

Nel caso in cui gli atomi sconnessi il prodotto tra partizioni rimane banale da implementare mentre l'intersezione è estremamente complicata – un'implementazione ottimale è data nel capitolo sugli algoritmi.

1.3.4 Grafi arbitrari

L'insieme dei siti in questo caso è sempre ordinabile con $i \in \{1, \ldots, L\}$. Definiamo la *matrice di adiacenza* A_{ij}

$$A_{ij} = \begin{cases} 1 & \text{se i siti sono connessi} \\ 0 & \text{altrimenti} \end{cases}$$

che ci permette di dare la relazione di equivalenza come

$$i \sim j \iff A(i,j) = 1,\ f(i) = f(j)$$

In questo modo ci si slega completamente da qualunque nozione di vicinanza geometrica. In linea di principio, è possibile rimuovere anche la condizione $f(i) = f(j)$, e lasciare alla matrice di adiacenza l'implementazione della relazione di equivalenza. Non vi sono condizioni da porre su A_{ij}, qualunque scelta è accettabile e va a modificare il partizionamento, senza che questo risulti mai incoerente.

Capitolo 2

Algoritmi

Dopo aver visto i possibili tipi di partizione, mostriamo come calcolare qualunque quantità richiesta in modo ottimale. Definiamo prima le operazioni limitate a partizioni con atomi connessi, sia per la loro minore complessità, sia perché nel caso generico si sfrutta la riduzione al caso più semplice già trattato.

2.1 Partizionamento connesso

Nel caso monodimensionale è sufficiente identificare il *bordo sinistro* di ogni atomo, per ricostruire completamente la partizione. Partendo da una sequenza in un array (o segmento unidimensionale finito) `S[i]` lungo L, definiamo l'array binario della partizione `B[i]`, lungo L, con indici che vanno da `0` a `L-1`

$$\texttt{B[i]} = \begin{cases} 1 & \text{il sito } \texttt{i} \text{ identifica un nuovo atomo} \\ 0 & \text{altrimenti} \end{cases}$$

algoritmicamente si procede in due step:

1. il primo sito sicuramente inizia un atomo, quindi `B[0]=1;`
2. iterando su `i`, poniamo `B[i]= (S[i] != S[i-1]);` infatti un nuovo atomo corrisponde ad un valore della sequenza *diverso* dal precedente

Ogni partizione espressa in termini binari ha sempre 1 come primo elemento.

Facendo un esempio, ad una stringa possiamo semplicemente far corrispondere l'array binario corrispondente

```
S=AAAABBBAAAACCCDDDD
B=1000100100010010000
```

Avendo un array che indica esattamente dove si trovano i bordi dei vari atomi, le operazioni di intersezione e prodotto sono immediate. Il prodotto corrisponde alla partizione che contiene i bordi di entrambe le partizioni (OR binario); l'intersezione ha un bordo quando compare contemporaneamente nelle due partizioni (AND binario); il prodotto ridotto contiene invece ogni bordo di una e una sola delle partizioni (XOR binario) – se il bordo appartiene ad entrambe, appartiene al fattore comune, che è stato ridotto.

Vettore	Risultato	Codice C	H_S	H_T
α	1011000110011011O1	`A[]={1,0,1,1,...};`	2.16	2.30
β	1000100100110010O0	`B[]={1,0,0,0,...};`	1.73	1.79
$\sigma = \alpha \wedge \beta$	1000000100010010O0	`C[i]=A[i] & B[i];`	1.33	1.39
$\alpha \vee \beta$	1011100110111011O1	`P[i]=A[i] | B[i];`	2.40	2.49
$\hat{\alpha} \vee \hat{\beta}$	1011100010101001O1	`Prid[i]=A[i] ^ B[i];`	2.09	2.19
$\hat{\alpha}$	1011000010001001O1	`Arid[i]=A[i] ^ C[i];`	1.80	1.94
$\hat{\beta}$	1000100000100000O0	`Brid[i]=B[i] ^ C[i];`	1.06	1.09

Tabella 2.1: Rappresentazione delle operazioni a partire da partizioni binarie. In rosso gli elementi riconducibili a partizioni di α, in blu quelli dovuti a β, in verde gli elementi di σ. Nel prodotto si vede l'effetto sovrapposto di blu e rosso. Nelle partizioni ridotte in verde sono stati evidenziati gli zeri dovuti alla riduzione, ovvero in corrispondenza agli atomi (1 verdi) di σ.

In tabella 2.1 sono mostrate tutte le sequenze generabili a partire da due partizioni binarie `A` e `B`, con le rispettive entropie, che adesso mostriamo come calcolare.

Entropia di una partizione binaria

Il calcolo più semplice è quello dell'entropia topologica, H_T. Avendo un vettore che indica esattamente l'inizio di ogni atomo, basta contare il numero di 1 presenti nel vettore e farne il logaritmo[1].

Per l'entropia di Shannon H_S la cosa è leggermente più complicata[2]:

1. si trovano tutte le posizioni i_k degli 1 nel vettore dato
2. si fa la differenza degli interi cosi trovati, $\mu_k = i_k - i_{k-1}$
3. l'insieme $\{\mu_k\}$ rappresenta le lunghezze degli atomi, ovvero gli intervalli tra 1 e l'ultimo 0 (se esiste) dello stesso atomo

Per l'ultimo atomo ci vuole un trattamento speciale, $\mu_{end} = \mathrm{L} - i_{end}$, ovvero si calcola la differenza tra la lunghezza della sequenza e l'ultimo 1 nel vettore. Il calcolo dell'entropia del vettore $\{\mu_m\} = \{\mu_k\} \cup \mu_{end}$ è molto semplice:

$$\mathrm{L} = \sum_m \mu_m$$

$$H = -\frac{\sum_m \mu_m \ln \mu_m}{\mathrm{L}} + \ln \mathrm{L}$$

la formula utilizzata permette di non dover dividere ogni μ_k per L, e di eseguire un'unica divisione alla fine. L'utilizzo delle misure nonnormalizzate degli atomi consente anche di poter precalcolare tutti i logaritmi necessari, in

[1] in Matlab è immediato: `Htop = log(sum(B));`

[2] In codice Matlab, si definisce prima `H=@(a) -sum(a.*log(a))/sum(a) +log(sum(a));` Avendo il vettore binario A, l'entropia cercata è: `H(diff(find([A,1])));`

Distanza	Formula	Valore	R
Nonridotta, Shannon	$2H_S(\alpha \vee \beta) - H_S(\alpha) - H_S(\beta)$	0.93	-
Ridotta, Shannon	$2\,H_S(\hat{\alpha}\hat{\beta}) - H_S(\hat{\alpha}) - H_S(\hat{\beta})$	1.33	1.46
Nonridotta, topologica	$2H_T(\alpha \vee \beta) - H_T(\alpha) - H_T(\beta)$	0.87	-
Ridotta, topologica	$2\,H_T(\hat{\alpha}\hat{\beta}) - H_T(\hat{\alpha}) - H_T(\hat{\beta})$	1.35	1.53

Tabella 2.2: Tutte le possibili distanze ricavate a partire dalla tabella 2.1

quanto $0 \leq \mu_k \leq \mathrm{L}$ – consente ciò di velocizzare l'esecuzione del programma del 600%.

L'algoritmo presenta complessità $\mathcal{O}(\mathrm{L})$ in quanto si scorre l'array completo, anche se una volta sola: chiaramente non si può far di meglio.

Distanza di Rohlin

Una volta che si è in grado di fare il prodotto, la riduzione e il calcolo dell'entropia, si può immediatamente calcolare la distanza di Rohlin. Per questo motivo è necessario calcolare anche le partizioni ridotte, che sono superflue per il calcolo del prodotto ridotto, ma nella distanza entrano in gioco le entropie dei fattori.

La tabella 2.2 calcola, sfruttando solamente i numeri calcolati nell'esempio precedente, le 4 possibili distanze date le due partizioni binarie.

2.2 Partizioni generali

Vi sono diversi algoritmi in questa sezione, tutti non banali. Introdurremo innanzitutto il prodotto e il calcolo dell'entropia che si riconduce al caso monodimensionale. Successivamente l'intersezione e il partizionamento a partire da sequenza arbitraria.

Nel caso generico non basta indicare l'inizio di ogni atomo, poichè questi possono avere forma (e buchi) di ogni tipo. È necessario etichettare ogni sito con un numero naturale, indicando con $\lambda_\alpha(i)$ l'etichetta corrispondente al sito i-esimo nella partizione α. Ogni etichetta identifica l'atomo di appartenenza:

$$i \sim j \iff \lambda(i) = \lambda(j)$$

Non si possono fare ipotesi sulla posizione relativa dei siti nè sulle etichette, che formano un vettore di lunghezza L.

2.2.1 Prodotto

Dovendo ragionare sui label assegnati ai siti, cerchiamo un modo di labellare in modo univoco i siti nella partizione prodotto. Dalla definizione si ottiene che un sito i appartiene all'atomo $(\alpha \vee \beta)_{jk}$ se appartiene a $\alpha_j \cap \beta_k$, ovvero se ha label "j" nella partizione α e label "k" nella partizione β.

Il prodotto deve mantenere la topologia, quindi prendendo due siti "vicini", ovvero appartenti allo stesso atomo in entrambe le partizioni, devono rimanere nello stesso atomo del prodotto

$$m, n \in \alpha_j \text{ e } m, n \in \beta_k \implies m, n \in (\alpha \vee \beta)_{jk}$$

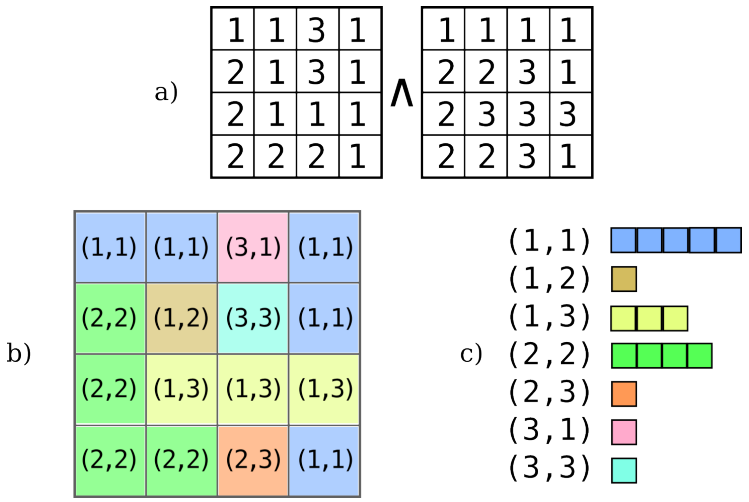

Figura 2.2.1: Calcolo di un prodotto tra partizioni. a)partizioni di partenza; b)partizione prodotto; c)lunghezze degli atomi ordinati del prodotto,al fine di calcolare l'entropia

ma i siti m, n hanno label $\lambda_\alpha(m) = j$ e $\lambda_\beta(m) = k$ rispettivamente per le due partizioni – devono avere lo stesso label nel prodotto, che può dipendere solo dal label degli atomi di partenza. L'unica scelta possibile è assegnare label (j, k) ad ogni sito del prodotto, ovvero la coppia (ordinata) dei label di partenza. Una volta eseguito l'assegnamento, gli atomi sono riconosciuti cercando label dello stesso valore. Tutto il procedimento è illustrato in figura 2.2.1.

Se i label sono rappresentati da interi a 32bit, la coppia può essere rappresentata da un intero a 64bit, avente nei primi 32bit il valore del primo label, e negli ultimi quello del secondo: in questo modo tutte le operazioni sono atomiche su un moderno processore.

Ricapitolando: al sito i-esimo, partendo da $\lambda_\alpha(i) = j$ e $\lambda_\beta(i) = k$, si assegna nel prodotto $\lambda_{\alpha\vee\beta}(i) = (j, k)$:

```
prod[i] = alpha[i] << 32 | beta[i];
```

come si vede dalla formula esplicita, $(j, k) \neq (k, j)$ in quanto gli indici identificano atomi appartenenti a partizioni diverse, con siti in genere diversi – se al posto del OR binario si volesse usare un'altra operazione, ne va cercata una asimmetrica nei due indici, per non unire atomi distinti. Ad esempio una partizione con labels {1,2} è equivalente a {2,1} (sempre due atomi distinti); il prodotto è ancora una partizione con due atomi, ma le etichette del prodotto sono: {(1,2),(2,1)} – devono essere chiaramente diverse.

Anche in questo caso il prodotto comporta un numero di operazioni sito-persito, con complessità $\mathcal{O}$(L).

2.2.2 Entropia

Per calcolare il numero di siti di un atomo, ovvero corrispondente ad ogni etichetta univoca, bisogna innanzitutto scorrere l'elenco per cercare tutte le etichette e ogni volta che se ne trova una non precedentemente catalogata, scorrere

l'intero array in avanti e contare quante volte appare. Questa è chiaramente un'operazione con complessità

$$\sum_{i=1}^{L}(L-i) = \mathcal{O}(L^2)$$

Ma si può fare meglio. Infatti non è necessario scorrere N volte l'intero elenco dei labels. Si può riordinare questo elenco tramite un'operazione di *sort*[3]. L'andamento asintotico è noto in letteratura essere $\mathcal{O}(L \log_2 L)$ nel numero di siti, con una costante di proporzionalità dipendente dall'algoritmo, ma non particolarmente onerosa.

Ordinando l'elenco delle etichette (non importa neanche se in ordine crescente o decrescente) si ottiene un array in cui sicuramente tutti gli elementi uguali sono consecutivi – si procede a questo punto al calcolo dell'entropia come se si trattasse di una sequenza unidimensionale!

La crescita nella complessità del fattore logaritmico è tipica del passaggio dal caso connesso a quello generale e si nota anche in ???

[3]Nello specifico ci affidiamo al *quicksort* delle librerie glibc