Reti neurali e forme di apprendimento

Il più bello dei mari è quello che non navigammo [...]
e quello che vorrei dirti di più bello non te l'ho ancora detto.
Nazim Hikmet

Daniele Tantari *Dipartimento di Matematica - Alma Mater Studiorum Bologna , Bologna, Italy*

In che modo il cervello umano apprende? Che caratteristiche deve avere un dispositivo in grado di apprendere come il cervello umano? A partire da queste due domande i modelli di rete neurale e le reti neurali artificiali stanno ripercorrendo lo stesso percorso formativo di un giovane studente: da reti che sanno imparare a memoria il più possibile e il più velocemente possibile si sta arrivando a modelli e dispositivi più complessi in grado di categorizzare, elaborare mappe concettuali, persino formare concetti e immagini nuove, in un processo che ricorda, senza pretesa di esserlo, la creatività.

L'eterna battaglia dell'insegnante è riuscire a motivare i suoi studenti verso livelli più alti di apprendimento convincendoli del fatto che imparare a memoria non sempre conviene: è una pratica troppo rigida, fragile e con evidenti limiti di scala. Rigida perché processa l'informazione ad un dettaglio estremo e senza l'esercizio critico di distinguere gli elementi più caratterizzanti da quelli meno rilevanti: tutto è estremamente importante e spesso si finisce che niente lo è. Rigidità porta fragilità: quando si impara una nozione a memoria si dà per scontato che questa sia corretta e non ci sono meccanismi di difesa contro un'eventuale scarsa affidabilità della fonte. Fragilità significa anche che se si dimentica qualcosa si dimentica tutto, dato che non ci sono informazioni a cui si riesce a dare priorità. I limiti di scala si riferiscono invece al fatto che la capacità di un qualsiasi dispositivo di memoria o del nostro cervello sono certo incredibili ma non illimitate, e l'apprendimento mnemonico non è quello più economico, quindi non quello più efficace per processare una grande mole di informazione. Nella maggior parte dei casi la capacità di sintetizzare o strutturare l'informazione è fondamentale.

Memorizzazione

Ci sono situazioni in cui tutti i dettagli hanno davvero la stessa importanza e la capacità di memorizzare è certamente indispensabile e in molti casi insostituibile: basti pensare all' importanza di ricordare (o di avere memorizzati in un dispositivo di memoria) un numero di telefono o il codice della propria carta, o di avere dei ricordi dettagliati delle esperienze vissute (anche se spesso ciò che rimane è solo un'idea astratta del ricordo, asciugata di molti dettagli). L' apprendimento mnemonico è il primo passo indispensabi-

le per lo sviluppo di forme di apprendimento più complete. Imparare a memoria non significa solo fissare dei concetti da qualche parte nella mente, significa sopratutto avere una strategia di recupero efficace. Alcuni sistemi fisici, per le proprietà della loro dinamica spontanea, possono essere considerati e usati come dispositivi di memoria, nel senso appena descritto. Si parla ad esempio della memoria di una molla o di un materiale a proposito della loro capacità di tornare in una configurazione di equilibrio. Nel contesto delle reti neurali si usa il termine *content-addressable memory* o memoria associativa. Consideriamo l'evoluzione temporale di un sistema fisico, il cui stato ad ogni istante t è descritto da un generico insieme di coordinate $\boldsymbol{\sigma}(t) = (\sigma_1(t), \dots, \sigma_n(t)) \in \Sigma$. La dinamica ha dei punti limite o attrattori, diciamo $\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_p \in \Sigma$, se, partendo sufficientemente vicini ad uno di questi punti, cioè $\boldsymbol{\sigma}(0) = \boldsymbol{\xi}^\mu + \boldsymbol{\epsilon}$, il sistema evolverà in modo che $\lim_{t\to\infty} \boldsymbol{\sigma}(t) = \boldsymbol{\xi}^\mu$. Se questo succede possiamo interpretare i punti limite come vettori di informazione salvati in memoria e la dinamica come il processo di recupero/ riaffioramento / completamento del ricordo che si attua attraverso la correzione degli errori di un punto di partenza impreciso $\boldsymbol{\sigma}(0)$, in cui il ricordo manca di qualche dettaglio (per esempio la sensazione di avere il ricordo sulla punta della lingua) oppure è ancora molto confuso. Ogni sistema con dinamica ad attrattori è dunque un dispositivo che impara a memoria fintanto che abbiamo la possibilità di inserire vettori di informazione come punti limite della dinamica.

Reti neurali che imparano a memoria

Una rete neurale ricorrente è un sistema dinamico composto da n unità (o neuroni) interagenti $\sigma_1, \dots, \sigma_n$ introdotto come modello semplificato di una rete neuronale biologica diventando poi paradigma per i modelli di reti neurali artificiali. In accordo al modello discreto di McCulloch-Pitts [1], ogni neurone può trovarsi in due stati: $\sigma_i = 1$ quando il neurone emette segnale (*firing*) mentre $\sigma_i = -1$ se il neurone è a riposo. Lo stato del sistema è quindi definito al tempo t dal vettore di coordinate $\boldsymbol{\sigma}(t) = (\sigma_1(t), \dots, \sigma_n(t)) \in \Sigma = \{-1, 1\}^n$. Quando due neuroni i e j sono connessi da una sinapsi, si descrive la forza e il tipo di interazione con la variabile $J_{ij} \in \mathbb{R}$ e ogni neurone riceve un segnale dai neuroni a cui è connesso pari a $h_i = \sum_{j=1}^n J_{ij}\sigma_j$. Ogni neurone possiede una soglia $\theta_i \in \mathbb{R}$ e ad ogni tempo reagisce al segnale in ingresso ribilanciandosi, cioè emettendo segnale se quello in ingresso supera la soglia, $\sigma_i = +1$ se $h_i > \theta_i$, rimanendo a riposo altrimenti, $\sigma_i = -1$ se $h_i < \theta_i$. La sinapsi sarà quindi eccitatoria se $J_{ij} > 0$ o inibitoria se $J_{ij} < 0$ mentre due neuroni non connessi avranno $J_{ij} = 0$. Questo è un modello molto semplificato di una rete neurale biologica e tralascia molti dettagli microscopici riguardanti la struttura spaziale del neurone e delle sinapsi, l'elettrochimica dell'interazione, la presenza di ritardi nella trasmissione del segnale. L'effetto di tutti questi dettagli è usualmente modellizzato dalla presenza di stocasticità nella dinamica che provoca fluttuazioni dalla regola deterministica introdotta precedentemente, per cui

$$\sigma_i(t+1) = \text{sign}\left(h_i(\boldsymbol{\sigma}(t)) - \theta_i + \beta^{-1} z_i(t)\right), \quad (1)$$

dove $\text{sign}(x) = 1, 0, -1$ se x è positivo, nullo o negativo, rispettivamente, $z_i(t)$ sono variabili aleatorie indipendenti e β^{-1} descrive il livello di stocasticità/rumore della dinamica e viene denominata temperatura del sistema. Assumeremo da qui in avanti per brevità che $\theta_i = 0$. Se la distribuzione del rumore è simmetrica, $\mathbb{P}(z \in A) = \mathbb{P}(-z \in A)$, possiamo scrivere in termini probabilistici che

$$\mathbb{P}(\sigma_i(t+1)|\boldsymbol{\sigma}(t)) = F(\beta\sigma_i(t+1)h_i(t)) \quad (2)$$

dove $F(.)$ è la funzione di distribuzione del rumore, $F(z) = \mathbb{P}(z_i(t) < z)$. Una scelta comune è una funzione di distribuzione sigmoidale $F(z) = \frac{1}{2}[1 + \tanh(z)] = e^z/2\cosh(z)$. In una rete neurale ricorrente va inoltre specificato in quale ordine i neuroni si aggiornano. Esistono due casi estremi:

- aggiornamento sequenziale: ad ogni tempo viene scelto a caso un neurone e aggiornato,

$$\mathbb{P}(\sigma_i(t+1)) = \frac{1 + \sigma_i(t+1)\tanh(\beta h_i(t))}{2} \quad (3)$$

- aggiornamento parallelo: ad ogni tempo tutti i neuroni sono aggiornati contemporanea-

mente,

$$\mathbb{P}(\boldsymbol{\sigma}(t+1)) = \frac{e^{\beta \sum_{ij=1}^{n} J_{ij}\sigma_i(t+1)\sigma_j(t)}}{\prod_{i=1}^{n} 2\cosh(\beta h_i(t))} \quad (4)$$

Una rete neurale ricorrente si trova in letteratura con nomi diversi in contesti diversi: *Kinetic Ising Model* [2, 3, 7, 5, 6] in Meccanica Statistica o *Discrete Vector Autoregressive process* in Econometria [7] ed è utilizzata come strumento di analisi di serie temporali multivariate in diversi ambiti, da quello biologico [8, 9, 10, 11, 12] a quello economico e finanziario [13, 14, 15, 16, 17]. Le proprietà della dinamica di una rete ricorrente dipendono fortemente dagli accoppiamenti. Consideriamo per semplicità il caso di una dinamica deterministica, $\beta^{-1} = 0$, e guardiamo qualche esempio in dettaglio:

- $J_{ij} = 0$, $\theta_i \neq 0$: in questo caso i neuroni non interagiscono e il loro stato dipenderà esclusivamente dai valori delle loro soglie. Infatti $\sigma_i(t+1) = -\operatorname{sign}(\theta_i)$, per ogni i, ad ogni tempo $t > 0$ e qualunque sia lo stato iniziale del sistema. $-\boldsymbol{\theta}$ è l'attrattore (globale) della dinamica.

- $J_{ij} = J/n > 0$, $\theta_i = 0$: in questo caso le interazioni sono tutte di tipo eccitatorio e spingono i neuroni ad imitarsi a vicenda. La dinamica $\sigma_i(t+1) = \operatorname{sign}(\frac{1}{n}\sum_j \sigma_j(t)) = \operatorname{sign}(\frac{1}{n}\sum_j \sigma_j(0))$ tende naturalmente su due stati speciali, $\boldsymbol{\sigma}^+ = 1$ quando $m(0) = \frac{1}{n}\sum_i \sigma_i(0) > 0$ e $\boldsymbol{\sigma}^- = -1$ altrimenti, che sono i due punti limite.

- $J_{ij} = J/n < 0$, $\theta_i = 0$ e dinamica parallela: in questo caso $\sigma_i(t+1) = -\operatorname{sign}(\frac{1}{n}\sum_j \sigma_j(t)) = (-1)^t \operatorname{sign}(m(0))$ e la dinamica finisce in un ciclo limite di periodo 2 oscillando indefinitamente tra gli stati $\boldsymbol{\sigma}^+$ e $\boldsymbol{\sigma}^-$.

Più in generale si può dimostrare [19] che, se le interazioni sono simmetriche $J_{ij} = J_{ji}$, da qualsiasi punto iniziale il sistema converge su un punto limite se la dinamica è sequenziale e senza autointerazioni $J_{ii} = 0$ e al più su un ciclo se la dinamica è parallela. Quindi una rete ricorrente ha una dinamica ad attrattori e può essere usata come dispositivo di memoria se siamo in grado di controllare i punti limite della dinamica.

La regola di Hebb

Supponiamo di voler memorizzare i vettori di informazione

$$\boldsymbol{\xi}^\mu = (\xi_i^\mu, \dots, \xi_n^\mu) \in \Sigma, \quad \mu = 1, \dots, p. \quad (5)$$

Dobbiamo costruire una rete di interazioni J_{ij} (e soglie θ_i) in modo che si formino p punti limite nella dinamica della rete in corrispondenza dei vettori di informazione, con un bacino di attrazione non nullo affinché la rete sia in grado di recuperare i vettori memorizzati attraverso la sua dinamica spontanea. Il processo di costruzione delle interazioni ottimali è detto *learning*. Se invece pensiamo alla memorizzazione in un dispositivo artificiale, qualsiasi ricetta per la costruzione da zero delle interazioni che permetta di costruire i punti limite desiderati è una ricetta utile. Se pensiamo al *learning* come al processo biologico per cui la rete neuronale riceve nuovi stimoli (nozioni, esperienze) e le memorizza modificando le proprietà delle sue sinapsi, allora deve essere un processo costruttivo: a partire da una struttura sinaptica esistente J_{ij} (eredità, conoscenze acquisite in passato), la rete si modifica all'arrivo di un nuovo ricordo aggiornando le sue connessioni, ΔJ_{ij}. La regola di aggiornamento deve essere `locale`, cioè deve dipendere da informazioni accessibili alla sinapsi J_{ij}, quindi ai neuroni i e j. Un esempio di regola di aggiornamento sinaptico biologicamente fondata è la cosiddetta `Regola di Hebb`: una rete neurale ricorrente senza alcuna passata esperienza, $J_{ij} = 0$, impara a memoria un vettore di informazione $\boldsymbol{\xi} \in \Sigma$ aggiornando le sinapsi come

$$\Delta J_{ij} = \frac{\xi_i \xi_j}{n}. \quad (6)$$

In questo modo due neuroni che richiediamo essere nello stesso stato nel punto limite tenderanno ad imitarsi grazie ad un incremento positivo della loro sinapsi, mentre tenderanno ad antiimitarsi altrimenti. Il risultato è che la dinamica della rete

$$\sigma_i(t+1) = \operatorname{sign}(\xi_i)\operatorname{sign}(\frac{1}{n}\sum_j \xi_j \sigma_j(t)) = \quad (7)$$

porterà ogni stato iniziale su $\boldsymbol{\sigma} = \boldsymbol{\xi}$ se al tempo zero $m(0) = \frac{1}{n}\sum_j \xi_j \sigma_j(0) > 0$, cioè se lo stato

Mi sforzo ma non ricordo!

Non sempre riusciamo a far affiorare il ricordo se non abbiamo informazioni sufficienti a risvegliarlo. La presenza delle soglie θ_i permette di modellizzare questa situazione. Supponiamo che $J_{ij} = \xi_i\xi_j/n$, con $\sum_i \xi_i = 0$, e $\theta_i = \theta$, $|\theta| < 1$. Se definiamo $m(t) = \frac{1}{n}\sum_i \xi_i\sigma_i(t)$ avremo che

$$m(t+1) = \frac{1}{2}\,\mathrm{sign}(m(t)+\theta)+\frac{1}{2}\,\mathrm{sign}(m(t)-\theta)$$

cioè

$$m(t) = \begin{cases} 1 & \text{se } m(0) > |\theta| \\ -1 & \text{se } m(0) < -|\theta| \\ 0 & \text{se } |m(0)| < |\theta|, \end{cases} \qquad (8)$$

cioè se l'informazione iniziale non è sufficiente (più grande della soglia), la rete finisce su un punto limite corrispondente a nessun ricordo.

della rete $\boldsymbol{\sigma}(0)$ possiede un (seppur piccolo ma positivo) grado di sovrapposizione con il ricordo $\boldsymbol{\xi}$. Come effetto collaterale si è creato un altro stato limite $\boldsymbol{\sigma} = -\boldsymbol{\xi}$ a cui convergono gli stati con $m(0) < 0$. Generalizzando al caso di p vettori di informazione da imparare a memoria avremo che la rete creata avrà interazioni

$$J_{ij} = \frac{1}{n}\sum_{\mu=1}^{p} \xi_i^\mu \xi_j^\mu. \qquad (9)$$

Nel caso in cui i vettori siano mutuamente ortogonali, cioè $\frac{1}{n}\boldsymbol{\xi}^\mu \cdot \boldsymbol{\xi}^\nu = \delta_{\mu\nu}$ che implica necessariamente $p < N$, la rete ricorrente è nota con il nome di Modello di Hopfield [20] e si dimostra che i p vettori di informazione sono tutti stati stazionari e attrattori della dinamica. Se lo stato iniziale possiede informazioni a sufficienza su uno dei vettori memorizzati, diciamo $\boldsymbol{\xi}^\mu$, cioè $m^\mu(0) = \sum_i \xi_i^\mu\sigma_i(0) > 0$, allora la rete riesce spontaneamente a ricordare e a ricostruire perfettamente il ricordo, come in Figura 1 (pannello intermedio).

Oltre agli stati $\pm\boldsymbol{\xi}^\mu$, `memorie fondamentali`, esistono anche altri attrattori della dinamica che corrispondono a misture dei p vettori di informazione fondamentali. Nel contesto della memorizzazione corrispondono a ricordi che si intrecciano, nozioni che si confondono, immagini che si accavallano e sono spesso considerate fallimenti dell'apprendimento, vedi Figura 1 (pannello inferiore). Vedremo nella prossima sezione che la creazione di immagini nuove è invece la capacità fondamentale di una rete neurale che apprende.

Se le memorie fondamentali non sono esattamente ortogonali (questo succede se i ricordi sono variabili aleatorie indipendenti o a maggior ragione se sono correlati), la situazione si complica: non solo nascono nuovi attrattori ma allo stesso tempo le memorie fondamentali possono destabilizzarsi e la rete non è più in grado di ricostruirle esattamente. Il fenomeno è detto *crosstalk*, termine che si riferisce al segnale di disturbo su una memoria dovuto alle interferenze delle altre. Abbiamo infatti che una memoria $\boldsymbol{\xi}^\mu$ è un punto fisso della dinamica solo se $\xi_i^\mu = \mathrm{sign}(\sum_j J_{ij}\xi_j^\mu)$, dove

$$(\boldsymbol{J}\boldsymbol{\xi}^\mu)_i = \xi_i^\mu + \sum_{\nu\neq\mu}^{p} \xi_i^\nu \frac{\boldsymbol{\xi}^\mu \cdot \boldsymbol{\xi}^\nu}{n}. \qquad (10)$$

Il secondo dei termini a destra rappresenta il contributo di *crosstalk* ed è nullo solo se i vettori sono ortogonali. In generale è tanto più grande quanto più i vettori sono correlati e anche nel caso di vettori *random* indipendenti è un contributo a media nulla ma con una varianza che cresce al crescere del numero di vettori/ricordi p che stiamo cercando di memorizzare, il cosiddetto `carico` della rete. Si definisce quindi la `capacità` il massimo numero di vettori di informazioni che la rete è in grado di imparare a memoria e quindi di ricostruire spontaneamente. Nel caso di vettori random scorrelati e a media nulla la capacità del modello di Hopfield cresce come $n/4\log n$ [21]. Nel caso in cui ci accontentassimo di una ricostruzione non perfetta, che preveda qualche errore qui e lì, allora la capacità cresce come una costante (piccola) per n.

La prospettiva meccanico-statistica

Questo è quello che succede a maggior ragione quando consideriamo una dinamica stocastica, con $\beta^{-1} > 0$. In questo caso la dinamica non può avere punti fissi in quanto c'è sempre una probabilità non nulla che qualche neurone cambi il suo stato. C'è bisogno di un linguaggio diver-

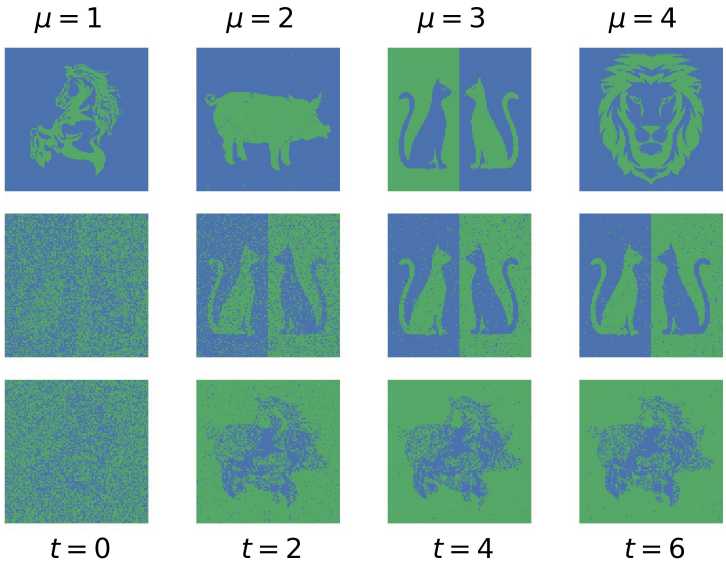

Figura 1: *Dinamica di richiamo in un Modello di Hopfield con $p = 4$ vettori di informazione (immagini) memorizzate (pannello superiore). A partire da uno stato iniziale rumoroso la rete riesce a richiamare alla memoria una delle immagini (pannello intermedio). A causa della correlazione tra le immagini memorizzate a volte la rete finisce su uno stato spurio (pannello inferiore).*

so per descrivere le proprietà della rete, che è quello della probabilità e più in particolare della Meccanica Statistica. L'equazione (3) ad esempio definisce la probabilità di transizione di una catena di Markov che ammette una distribuzione d'equilibrio

$$P_\beta(\boldsymbol{\sigma}) = Z^{-1} \exp\left(\beta \sum_{i<j=1}^{n} J_{ij}\sigma_i\sigma_j\right) \qquad (11)$$

su Σ, dove $Z = \sum_\sigma \exp\left(\beta \sum_{ij}^n J_{ij}\sigma_i\sigma_j\right)$, costante di normalizzazione che assicura $\sum_\sigma P_\beta(\sigma) = 1$, è detta funzione di partizione. Questo significa che le traiettorie del sistema, che descrivono lo stato della rete nel tempo, esplorano lo spazio di tutti i possibili ricordi Σ, ricostruendone ognuno con probabilità $P_\beta(\boldsymbol{\sigma})$. Quest'ultima è detta distribuzione di Boltzmann-Gibbs e conferisce alla rete ricorrente anche il titolo di `Macchina di Boltzmann`, che si aggiunge ai precedenti e viene usato spesso nel contesto delle reti neurali artificiali. Non possiamo più studiare i punti limite di una dinamica deterministica, però ha senso individuare le regioni visitate quasi certamente o con probabilità molto alta. Nel caso del modello di Hopfield, in cui $J_{ij} = \frac{1}{n}\sum_{\mu=1}^{p} \xi_i^\mu \xi_j^\mu$, quando il rumore stocastico non è troppo alto ($\beta \gg 1$) e il carico della rete è sufficientemente basso $p \ll n$, si dimostra che le configurazioni visitate tipicamente formano dei cluster, ed esistono cluster centrati sui vettori di informazione $\{\boldsymbol{\xi}^\mu\}$, cioè in cui

$$\langle m^\mu(\boldsymbol{\sigma})\rangle = \left\langle \frac{1}{n}\sum_{i=1}^{n} \xi_i^\mu \sigma_i \right\rangle = m_\beta^\mu > 0 \qquad (12)$$

e $m_\beta^\nu \sim 0$ per $\nu \neq \mu$. Questi cluster sono detti stati puri e il regime appena descritto viene detto di `richiamo`, in quanto la rete sta imparando a memoria qualitativamente come nel caso deterministico, richiamando alla mente i vettori di informazione immagazzinati nella struttura sinaptica e in modo spontaneo, infatti per $t \gg 1$

$$m^\mu(t) = \frac{1}{n}\sum_{i=1}^{n} \xi_i^\mu \sigma_i(t) \sim m_b^\mu > 0, \qquad (13)$$

e $m^\nu(t) \sim 0$ per $\nu \neq \mu$. Tuttavia se la temperatura della dinamica o il carico della rete diventano troppo alti, nascono due regimi di lavoro differenti:

- regime `paramagnetico`: in questo regime la temperatura della rete è troppo alta ed ogni neurone si comporta perlopiù in modo casuale senza bloccarsi in configurazioni specifiche. Avremo di conseguenza che per ogni μ, $\langle m^\mu(\boldsymbol{\sigma})\rangle \sim 0$, cioè la rete non riesce a richiamare nessun ricordo ma fluttua indefinitamente.

- regime vetroso o di `spin glass`: in questo regime il carico della rete e quindi le interferenze tra i ricordi sono eccessivi. I neuroni si bloccano su configurazioni specifiche che

però sono solo debolmente correlate con i ricordi, cioè ancora una volta, ma per motivi diversi $\langle m^\mu(\boldsymbol{\sigma})\rangle \sim 0$ per ogni μ. In termini di cluster, gli stati puri spariscono così come ogni altro gruppo di configurazioni tipiche correlate con i ricordi.

La capacità del modello di Hopfield come dispositivo che impara a memoria è riassunto quindi nel diagramma di fase approssimato di Figura 2.

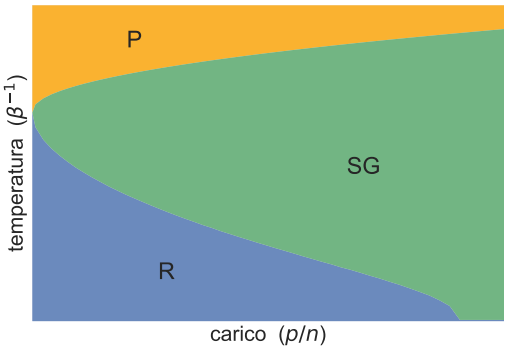

Figura 2: *Diagramma di fase del Modello di Hopfield nel piano carico-temperatura con i tre regimi: R) fase di richiamo, P) fase paramagnetica, SG) fase di spin glass.*

Apprendimento

Nell'apprendimento di alto livello l'obiettivo passa dalla memorizzazione/riproduzione di vettori di informazione alla ricerca di una struttura, un filo comune, di regole generali che li spieghino.

Se nell'apprendimento mnemonico i vettori di informazione (immagini, nozioni, etc.) sono dei modelli ideali da incamerare esattamente, intorno ai quali costruire la struttura sinaptica e a cui la rete/il dispositivo si riferisce ogni volta che viene stimolata differentemente (con la dinamica di richiamo dal punto iniziale al punto limite), nell'apprendimento vero e proprio diventano degli `esempi` di realtà, da studiare, strutturare, per elaborare un modello interno che la realtà la spieghi.

Se nell'apprendimento mnemonico il contesto ideale è rappresentato da pochi vettori di informazione da memorizzare, possibilmente scorrelati per evitare stati spuri, nell' apprendimento propriamente detto la cosa migliore è avere tanti esempi (con la loro struttura di correlazione) dai quali evincere un modello interno idealizzato.

Supponiamo di dover imparare a distinguere animali diversi. Una rete che impara a memoria ha bisogno, quasi biblicamente, di un'immagine ideale per specie da fissare alla mente: ogni immagine nuova richiamerà alla memoria una delle immagini modello, la più simile. Una rete che apprende invece osserva tantissime immagini di animali e le organizza imparando a riconoscerne la struttura di similarità e creando internamente prototipi o caratteristiche idealizzate. In questo senso si parla di apprendimento non supervisionato.

Macchine di Boltzmann che apprendono

La ricerca dei principi di apprendimento non supervisionato (biologico e artificiale) ha motivato lo sviluppo di una prospettiva differente sull'uso delle Macchine di Boltzmann [22, 23]. La distribuzione (11) assegna probabilità alta alle configurazioni $\boldsymbol{\sigma}$ che soddisfano il maggior numero di vincoli della forma $\sigma_i\sigma_j = \mathrm{sign}(J_{ij})$, cioè con energia

$$E = -\sum_{i<j}^{n} J_{ij}\sigma_i\sigma_j \tag{14}$$

più bassa. È la distribuzione di equilibrio di una dinamica stocastica in cui ogni neurone/unità si accende con probabilità (3), che può essere riscritta come

$$\mathbb{P}(\sigma_i(t+1) = 1) = \frac{1}{1 + e^{-\beta\Delta E_i(t)}}, \tag{15}$$

dove $\Delta E_i(t)$ è la differenza di energia (o di vincoli soddisfatti) se il neurone i-esimo passa da spento a acceso. In questo senso gli accoppiamenti possono essere interpretati come i vincoli che devono essere soddisfatti dalle configurazioni più visitate nel tempo dalla rete. Quando la rete riceve un vettore di informazione $\boldsymbol{\xi}^\mu$ i neuroni sono forzati (*clamped*) ad essere nella configurazione $\boldsymbol{\sigma} = \boldsymbol{\xi}^\mu$ e le sinapsi si muovono di conseguenza in un processo di learning. A differenza dell'apprendimento mnemonico in cui $\Delta J_{ij} \propto \xi_i^\mu\xi_j^\mu$, cioè ad ogni passo la rete si concentra per soddisfare i vincoli di un solo vettore di informazione alla volta, nell' apprendimento non supervisiona-

to la rete cerca di costruire una visione d'insieme, un modello interno per spiegare contemporaneamente tutti i vettori di informazione, che in questo contesto chiameremo esempi e il loro insieme *training set* $\mathcal{M} = \{\boldsymbol{\sigma}^a\}_{a=1}^m$[1]. Invece di chiedere rigidamente che i vettori di informazione siano punti limite della dinamica, chiederemo che le proprietà del training set siano riprodotte bene dalla rete, o in altre parole che la distribuzione di probabilità (empirica) dei vettori del training set sia ben approssimata da quella d'equilibrio della rete/macchina. Se $P_\beta(\boldsymbol{\sigma})$ è la distribuzione della macchina di Boltzmann e $\hat{P}(\boldsymbol{\sigma}) = \frac{1}{m}\sum_{a=1}^m \delta_{\boldsymbol{\sigma}\boldsymbol{\sigma}^a}$ la distribuzione empirica del training set, si può definire la distanza tra le due come

$$\mathcal{D}(\hat{P}||P_\beta) = \sum_{\boldsymbol{\sigma}} \hat{P}(\boldsymbol{\sigma}) \log \frac{\hat{P}(\boldsymbol{\sigma})}{P_\beta(\boldsymbol{\sigma})}. \qquad (16)$$

L'obiettivo di un apprendimento non supervisionato è minimizzare questa distanza, o equivalentemente, massimizzare la funzione

$$\mathcal{L} = \sum_{a=1}^m \log P_\beta(\boldsymbol{\sigma}^a), \qquad (17)$$

detta *log-likelihood*, che (logaritmo a parte) rappresenta la probabilità che tutto il training set di esempi sia generato e quindi spiegato dalla rete. Si può massimizzare iterativamente questa quantità calcolando la sua derivata rispetto agli accoppiamenti della rete

$$\frac{\partial \log P_\beta(\boldsymbol{\sigma}^a)}{\partial J_{ij}} = \beta\left(\sigma_i^a\sigma_j^a - \langle\sigma_i\sigma_j\rangle\right), \qquad (18)$$

dove $\langle . \rangle$ indica la media rispetto a P_β, e incrementando il valore delle sinapsi proporzionalmente

$$\Delta J_{ij} \propto \left(\sigma_i^a\sigma_j^a - \langle\sigma_i\sigma_j\rangle\right). \qquad (19)$$

Il metodo è detto *gradient ascent* perché la rete segue la direzione di massima crescita fornita dal gradiente ed è una generalizzazione della regola di Hebb (6). Come la regola di Hebb ha una forma locale e contiene il termine $\sigma_i^a\sigma_j^a$ (equivalente a $\xi_i^\mu\xi_j^\mu$). A differenza di essa non si esaurisce in m (equivalentemente p) passi ma si rinforza fino a convergenza, momento in cui $\langle\sigma_i\sigma_j\rangle$ sarà vicino a $\frac{1}{m}\sum_a \sigma_i^a\sigma_j^a$, cioè la rete avrà imparato la struttura di correlazione del training set. Se vogliamo una rete che impari delle correlazioni di ordine maggiore possiamo generalizzare la macchina di Boltzmann vista finora dividendo i neuroni in due gruppi con funzioni diverse: neuroni `visibili` $\boldsymbol{\sigma}$ e neuroni `nascosti` $\boldsymbol{\tau}$. I neuroni visibili sono l'interfaccia con l'ambiente esterno, quelli che vengono stimolati direttamente (e clamped) da un input di informazione: nelle reti viste finora tutti i neuroni erano visibili. I neuroni nascosti, *hidden units*, non interagiscono mai direttamente con l'ambiente esterno e servono alla rete per spiegare dei vincoli che non possono essere rappresentati da interazioni di coppia tra neuroni visibili, quindi per imparare una realtà più complessa. L'obiettivo della macchina rimane quello di riprodurre al meglio la distribuzione di probabilità empirica del training set $\hat{P}(\boldsymbol{\sigma})$ con la distribuzione d'equilibrio dei suoi neuroni visibili

$$P_\beta(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\tau}} P_\beta(\boldsymbol{\sigma}, \boldsymbol{\tau}), \qquad (20)$$

ottenuta marginalizzando su tutte le possibili configurazioni dei neuroni nascosti. Minimizzando di nuovo la distanza $\mathcal{D}(\hat{P}||P_\beta)$, o equivalentemente massimizzando la log-likelihood, si ottiene una regola di aggiornamento delle sinapsi analoga alla (18) essendo

$$\frac{\partial \log P_\beta(\boldsymbol{\sigma}^a)}{\partial J_{ij}} = \beta\left(\sigma_i^a\sigma_j^a - \langle\sigma_i\sigma_j\rangle\right), \qquad (21)$$

se J_{ij} connette due neuroni visibili. Se la sinapsi $J_{\mu\nu}$ connette due neuroni nascosti

$$\frac{\partial \log P_\beta(\boldsymbol{\sigma}^a)}{\partial J_{\mu\nu}} = \beta\left(\langle\tau_\mu\tau_\nu\rangle_a - \langle\tau_\mu\tau_\nu\rangle\right), \qquad (22)$$

dove $\langle . \rangle_a$ identifica la media rispetto a $P_\beta(\boldsymbol{\tau}|\boldsymbol{\sigma}^a)$, cioè la distribuzione d'equilibrio dei neuroni nascosti ottenuta condizionando i neuroni visibili (clamped) sull'input $\boldsymbol{\sigma}^a$. Infine se la sinapsi $J_{i\mu}$ connette un neurone visibile ed uno nascosto si avrà

$$\frac{\partial \log P_\beta(\boldsymbol{\sigma}^a)}{\partial J_{i\nu}} = \beta\left(\sigma_i^a\,\langle\tau_\nu\rangle_a - \langle\sigma_i\tau_\nu\rangle\right). \qquad (23)$$

[1] Cambiamo notazione, da $\{\boldsymbol{\xi}^\mu\}_{\mu=1}^p$ a $\{\boldsymbol{\sigma}^a\}_{a=1}^m$ per sottolineare che la strategia di apprendimento sarà diversa e differenziare i vettori di informazione di un apprendimento mnemonico dagli esempi di un apprendomento non supervisionato, dove per altro tipicamente $m \gg p$

Si dimostra che la dinamica di apprendimento è convessa fintanto che ci sono solo neuroni visibili mentre possono nascere minimi locali quando la rete usa anche dei neuroni nascosti. Questi minimi locali corrispondono a diversi modi in cui la rete può usare le unità nascoste per spiegare i vincoli impliciti presenti nel training set. Alla fine dell'allenamento, proprio le sinapsi che collegano i neuroni nascosti a quelli visibili ci rivelano quello che la rete ha imparato e in che modo è stato codificato.

Macchine di Boltzmann ristrette

Per capire meglio consideriamo un esempio di rete con neuroni nascosti con una struttura semplice ma allo stesso tempo molto rilevante perché alla base delle moderne strutture di reti neurali artificiali: le Macchine di Boltzmann ristrette (RBM). In questo tipo di rete neurale le sinapsi collegano solamente neuroni visibili a neuroni nascosti, non essendo invece presenti interazioni dirette tra coppie di neuroni dello stesso tipo. Una RBM può quindi essere rappresentata da un grafo bipartito come in Figura 5 e Figura 6 e la sua distribuzione di equilibrio sarà

$$P_\beta(\boldsymbol{\sigma},\boldsymbol{\tau}) = Z_\beta^{-1} P_0(\boldsymbol{\tau}) \exp\left(\beta \sum_{i\mu} J_{i\mu}\sigma_i\tau^\mu\right)$$

dove abbiamo aggiunto il termine $P_0(\boldsymbol{\tau})$ che rappresenta la distribuzione a priori dei neuroni nascosti. La forma specifica di una RBM introduce numerosi vantaggi da un punto di vista computazionale nel calcolo del gradiente e quindi dell'aggiornamento delle sinapsi (23), derivanti principalmente dal fatto che i neuroni nascosti sono indipendenti condizionatamente allo stato dei visibili e viceversa. Da un punto di vista teorico la proprietà più interessante invece è che possiamo facilmente calcolare la distribuzione d'equilibrio sui neuroni visibili marginalizzando sui neuroni nascosti ottenendo

$$\begin{aligned} P_\beta^{\mathrm{RBM}}(\boldsymbol{\sigma}) &= Z_\beta^{-1}\mathbb{E}_{\boldsymbol{\tau}} \exp\left(\beta \sum_{i\mu} J_{i\mu}\sigma_i\tau^\mu\right) \\ &= Z_\beta^{-1} \exp\left(\sum_\mu \psi(\beta \sum_i J_{i\mu}\sigma_i)\right), \end{aligned} \tag{24}$$

dove $\psi(x) = \log \mathbb{E}_\tau[e^{\tau x}]$. La forma specifica dipende quindi dalla forma del *prior*, quindi dalla natura di neuroni nascosti. Supponiamo di avere neuroni nascosti gaussiani $\mathcal{N}(0,1)$, i.e. $P_0(\tau) = \frac{1}{\sqrt{2\pi}} e^{-\tau^2/2}$, allora $\psi(x) = x^2/2$ e

$$P_\beta^{\mathrm{RBM}}(\boldsymbol{\sigma}) = Z_\beta^{-1} \exp\left(\beta \sum_{i<j} \hat{J}_{ij}\sigma_i\sigma_j\right), \tag{25}$$

dove

$$\hat{J}_{ij} \propto \sum_\mu J_i^\mu J_j^\mu. \tag{26}$$

cioè una RBM ha la distribuzione d'equilibrio di un modello di Hopfield con vettori di informazione $\{\boldsymbol{J}^\mu\}$ [24, 25, 27] . Questo è il momento migliore quindi per confrontare un apprendimento non supervisionato (tramite RBM) con uno mnemonico (con apprendimento Hebbiano). A differenza dell'apprendimento mnemonico, in cui le sinapsi sono costituite direttamente dai vettori di infromazione/ modelli osservati $\{\boldsymbol{\xi}^\mu\}$ (disposti con la regola di Hebb), nell'apprendimento non supervisionato le componenti principali delle sinapsi sono i vettori $\{\boldsymbol{J}^\mu\}$ che dipendono dagli esempi $\{\boldsymbol{\sigma}^a\}$ solo tramite il processo di apprendimento (23). Si può dire che nell'apprendimento non supervisionato la rete impara i modelli ($\{\boldsymbol{J}^\mu\}$) dagli esempi invece che imparare a memoria gli esempi stessi. Finché gli esempi sono pochi e ortogonali allora i due apprendimenti sono tutto sommato equivalenti, ma quando gli esempi sono tanti e correlati tra loro la Macchina di Boltzmann imparerà i modelli $\{\boldsymbol{J}^\mu\}$ che sintetizzano al meglio l'informazione del *training set*.

Un esperimento controllato

Per capire meglio le capacità di apprendimento di una RBM possiamo eseguire un esperimento di tipo *teacher-student* [28, 29]. Un insegnante possiede l'informazione $\boldsymbol{\xi}$ che vuole trasmettere al suo studente, tuttavia non vuole che la impari a memoria bensì lo stimola ad apprendere con le sue forze concedendogli solo degli esempi su cui studiare. Un esempio $\boldsymbol{\sigma}^a$ è un'informazione parziale, o un po' distorta, con qualche dettaglio differente dalla nozione esatta costituita dal vettore $\boldsymbol{\xi}$, e può essere costruito a partire da esso semplicemente scambiando qualche bit con pro-

babilità $\epsilon \in (0, 1/2)$[2]: al crescere di ϵ l'esempio sarà via via più lontano dal suo modello mentre se ϵ è prossima a zero gli esempi saranno molto calzanti e porteranno molta più informazione sulla vera nozione da imparare $\boldsymbol{\xi}$. Lo studente riceve quindi un insieme di m esempi $\mathcal{M} = \{\boldsymbol{\sigma}^a\}_{a=1}^m$, il *training set*, da osservare e da cui apprendere. Supponiamo che il metodo di studio dello studente (comprensivo dei suoi dispositivi e il suo cervello) sia quello di una RBM con un neurone nascosto che apprende con lo schema di apprendimento non supervisionato introdotto precedentemente. Al termine dell'apprendimento le sue sinapsi saranno il vettore $\boldsymbol{J}$. Per capire l'efficacia del suo apprendimento dobbiamo valutare la distanza tra $\boldsymbol{J}$ e $\boldsymbol{\xi}$, l'informazione che lo studente non conosceva, e lo facciamo tramite la quantità

$$q = \langle \boldsymbol{J}, \boldsymbol{\xi} \rangle := \frac{1}{n} \sum_{i=1}^{n} \text{sign}(J_i)\xi_i : \qquad (27)$$

più alto è q migliore sarà stata la performance dello studente. In Figura 3 si può seguire la performance dello studente durante l'allenamento.

Quando la dimensione del vettore d'informazione è molto alta ($n \to \infty$) è molto difficile che lo studente riesca solo per caso ad imparare qualcosa, in questo caso si avrà sempre $q \sim 0$, quindi possiamo giudicare la performance positiva non appena $q > 0$. Possiamo distinguere tre regimi (Figura 4):

- $\epsilon \sim 0$: gli esempi $\boldsymbol{\sigma}^a$ saranno molto simili al modello $\boldsymbol{\xi}$ e lo studente può raggiungere un risultato positivo anche imparando a memoria uno degli esempi. In questo caso lo studente può prendere alla lettera ogni parola dell'insegnante, che è molto affidabile o molto (troppo?) buono. Questo è il caso in cui apprendimento mnemonico e non supervisionato sono parimenti efficaci.

- $\epsilon \sim 1/2$ e $m/n \ll 1$: gli esempi sono pochi e hanno pochissima informazione sul modello. In questo caso l'insegnante è un cattivo insegnante e né l'apprendimento mnemonico né quello non-supervisionato possono dare risultati positivi: semplicemente lo studente, anche se molto in gamba, non ha ricevuto l'informazione sufficiente per poter apprendere.

- $\epsilon \sim 1/2$ e $m/n \gg 1$: gli esempi hanno poca informazione sul modello ma sono molto numerosi. Questo è molto probabilmente il

[2] $\epsilon = 1/2$ corrisponde alla perdita totale dell'informazione, se ϵ fosse maggiore gli esempi risultanti sarebbero tipicamente dei negativi, che portano la stessa informazione dell'originale

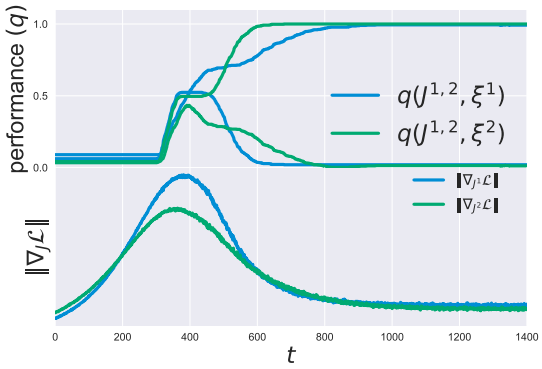

Figura 3: *Monitoraggio della fase di allenamento di una RBM in un esperimento teacher-student. La rete ha 2 neuroni nascosti e deve apprendere 2 vettori di informazione ortogonali tra loro. Quando il gradiente si stabilizza su un minimo le sinapsi allenate $\boldsymbol{J}^1$ e $\boldsymbol{J}^2$ hanno imparato i due vettori di informazione dell' insegnante $\boldsymbol{\xi}^1$ e $\boldsymbol{\xi}^2$*

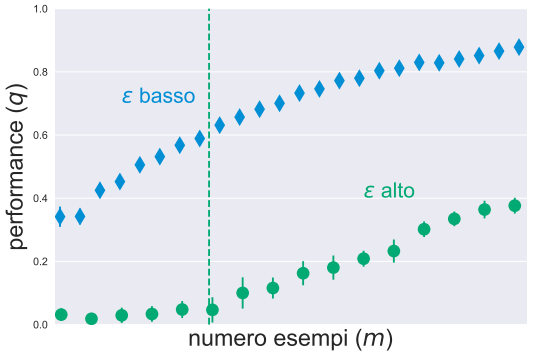

Figura 4: *Esperimento teacher-student e performance dello studente al variare del numero di esempi a disposizione m. Lo studente apprende allenando una RBM su un training set di esempi con rumore ϵ. Finchè il rumore è basso lo studente apprende con pochi esempi e l' apprendimento è mnemonico. Quando il rumore è alto l' apprendimento mnemonico non è efficace e lo studente necessita di un numero di esempi superiori ad una certa soglia.*

caso di un insegnante che vuole stimolare lo studente, che non dà l'informazione tutta insieme ma distribuita su un numero sufficientemente alto di esempi. Se lo studente imparasse a memoria gli esempi ($\boldsymbol{J} = \boldsymbol{\sigma}^a$), non otterrebbe un risultato positivo, perché $q = \langle \boldsymbol{J}, \boldsymbol{\xi} \rangle = \langle \boldsymbol{\sigma}^a, \boldsymbol{\xi} \rangle \sim 1 - 2\epsilon \sim 0$. Il risultato dell'apprendimento tramite RBM è invece positivo.

Le stesse considerazioni si generalizzano al caso di una coppia *teacher-student* con p vettori di informazioni e p neuroni nascosti [28, 29].

Prototipi e caratteristiche

L' esperimento precedente rappresenta un caso molto speciale di apprendimento in cui ciò che si deve imparare è uno o più prototipi ideali e il numero di neuroni nascosti della rete è proprio uguale al numero di prototipi da imparare. Ci sono allora due classi di domande ulteriori che ci si può porre:

- La ricerca di prototipi ideali è l'unica forma di apprendimento?
- Come la struttura della rete (per esempio il numero di neuroni nascosti) influenza l' apprendimento, cioè il modello interno con cui la rete interpreta e spiega la realtà ?

Riguardo la prima questione esistono almeno due modalità differenti di apprendimento o *pattern recognition*: la formazione di `prototipi` e la combinazione di `caratteristiche`. Nel primo caso, come nell'esempio precedente, la rete impara a riconoscere gli oggetti nel complesso, costruendo dei modelli ideali, appunto i prototipi, interpretando la realtà come realizzazione imperfetta di essi. Nel secondo caso la rete impara a riconoscere, estraendole dagli esempi e codificandole nelle sinapsi, delle caratteristiche ed interpreta la realtà come combinazione di esse. Ad esempio una rete può imparare a distinguere i gatti dai maiali costruendo due immagini interne ideali di felinità e suinità (Figura 5) oppure combinando insieme immagini interne di caratteristiche come la forma della coda, del muso etc... (Figura 6)

Una RBM che ha appreso dei prototipi è un modello di Hopfield che genera cluster di configurazioni centrate su $\boldsymbol{J}^\mu$ (i prototipi). In ogni cluster,

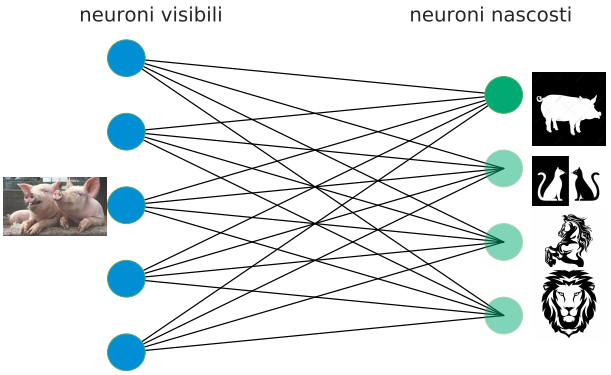

Figura 5: *Schema di una RBM che apprende per prototipi: l' immagine sui neuroni visibili è interpretata come variante di uno dei prototipi memorizzati. Il neurone nascosto corrispondente è acceso mentre tutti gli altri sono quasi spenti.*

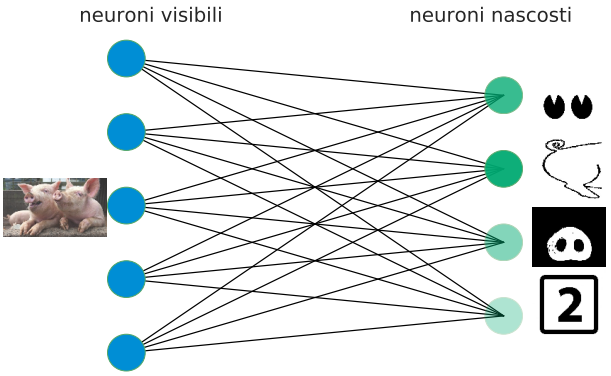

Figura 6: *Schema di una RBM che apprende per caratteristiche: l' immagine sui neuroni visibili è decomposta e interpretata come sovrapposizione delle caratteristiche memorizzate. Più neuroni nascosti, quelli corrispondenti alle caratteristiche utilizzate, sono accesi.*

diciamo il μ- esimo, un solo neurone nascosto τ^μ è acceso, stimolato da

$$P(\tau^\mu|\boldsymbol{\sigma}) = Z_\mu^{-1} \exp\left(\beta m^\mu(\boldsymbol{\sigma})\tau^\mu\right), \qquad (28)$$

dove $m^\mu(\boldsymbol{\sigma}) = \boldsymbol{J}^\mu \cdot \boldsymbol{\sigma} \gg 0$, mentre tutti gli altri sono spenti essendo $m^\nu(\boldsymbol{\sigma}) \ll 1$ per $\nu \neq \mu$. In una RBM che ha estratto caratteristiche i clusters sono centrati su stati spuri che sono combinazioni dei vettori $\boldsymbol{J}^\mu$, questa volta interpretabili come caratteristiche. In questo caso in ogni cluster più di un neurone nascosto è acceso, anche se più debolmente, essendo $m^\mu(\boldsymbol{\sigma}) > 0$ per ogni μ corrispondente ad una caratteristica $\boldsymbol{J}^\mu$ presente nel cluster.

Riguardo la seconda questione, sicuramente la tipologia di apprendimento è influenzata fortemente dai parametri che regolano l'architettura della rete come il numero di neuroni nascosti, la loro natura e la presenza di vincoli sulla struttura sinaptica. In particolare si può mostrare come al variare di questi parametri una RBM allenata su uno stesso training set di esempi passi da un apprendimento per prototipi ad uno per caratteristiche [30, 31].

Va sottolineato infine che un apprendimento di alto livello deve necessariamente saper integrare entrambe le modalità, estraendo caratteristiche e poi combinandole per creare caratteristiche di livello sempre più alto fino a costruire dei prototipi ideali. Per questo motivo i dispositivi più efficienti di reti neurali che apprendono percorrono il paradigma del *Deep Learning*, in cui più RBM sono impilate, e allenate, una dopo l'altra [32].

Abbiamo visto reti neurali che imparano a memoria immagini, abbiamo visto reti neurali che apprendono tramite immagini. Memorizzazione come registrazione, apprendimento come rielaborazione. Ma per una teoria dell'immaginazione propriamente detta manca ancora tanta fantasia.

[1] W. S. McCulloch, W. Pitts: *A logical calculus of the ideas immanent in nervous activity*, The bulletin of mathematical biophysics, 5 (1943) 115.

[2] B. Derrida, E. Gardner, A. Zippelius: *An exactly solvable asymmetric neural network model*, EPL (Europhysics Letters), 4 (1987) 167.

[3] A. Crisanti, H. Sompolinsky: *Dynamics of spin systems with randomly asymmetric bonds: Ising spins and Glauber dynamics*, Physical Review A, 37 (1988) 4865.

[4] B. Dunn,Y. Roudi: *Learning and inference in a nonequilibrium ising model with hidden nodes.* Physical Review E, 87 (2013) 022127.

[5] A. Decelle, Pan Zhang: *Inference of the sparse kinetic ising model using the decimation method*, Physical Review E, 91 (2015) 052136.

[6] C. Campajola, F. Lillo, D. Tantari: *Inference of the kinetic ising model with heterogeneous missing data*, Physical Review E, 99 (2019) 062138.

[7] C. Campajola, F. Lillo, P. Mazzarisi, D. Tantari, *On the equivalence between the Kinetic Ising Model and discrete autoregressive processes*, arXiv preprint arXiv:2008.10666 (2020)

[8] S. Cocco, R. Monasson, L. Posani, G. Tavoni: *Functional networks from inverse modeling of neural population activity*, Current Opinion in Systems Biology, 3 (2017) 103.

[9] T.-A. Nghiem, B. Telenczuk, O. Marre, A. Destexhe, U. Ferrari: *Maximum-entropy models reveal the excitatory and inhibitory correlation structures in cortical neuronal activity*, Physical Review E, 98 (2018) 012402.

[10] U. Ferrari, S. Deny, M. Chalk, G. Tkačik, O. Marre, T. Mora: *Separating intrinsic interactions from extrinsic correlations in a network of sensory neurons*, Physical Review E, 98 (2018) 042410.

[11] S. Tanaka, H. A. Scheraga: *Model of protein folding: incorporation of a one-dimensional short-range (ising) model into a three-dimensional model*, Proceedings of the National Academy of Sciences, 74 (1977) 1320.

[12] A. Imparato, A. Pelizzola, M. Zamparo: *Ising-like model for protein mechanical unfolding*, Physical Review Letters, 98 (2007) 148102.

[13] Stefan Bornholdt: *Expectation bubbles in a spin model of markets: Intermittency from frustration across scales*, International Journal of Modern Physics C, 12 (2001) 667.

[14] J.-Ph. Bouchaud: *Crises and collective socio-economic phenomena: Simple models and challenges*, Journal of Statistical Physics, 151 (2013) 606|.

[15] D. Sornette: *Physics and financial economics (1776 - 2014): puzzles, ising and agent-based models*, Reports on progress in physics, 77 (2014) 062001.

[16] C. Campajola, F. Lillo, D. Tantari: *Unveiling the relation between herding and liquidity with trader lead-lag networks*, Quantitative Finance, in press, (2020) .

[17] C. Campajola, D. Di Gangi, F. Lillo, D. Tantari: *Modelling time-varying interactions in complex systems: the Score Driven Kinetic Ising Model*, arXiv preprint arXiv:2007.15545 (2020)

[18] Y. LeCun, Y. Bengio, G. Hinton: *Deep learning*, Nature, 521 (2015) 436.

[19] A. C. Coolen, R. Kühn, P. Sollich: *Theory of neural information processing systems.* Oxford University Press, Oxford (2005).

[20] J. J. Hopfield, *Neural networks and physical systems with emergent collective computational abilities.* Proceedings of the national academy of sciences, 79(8), 2554-2558 (1982).

[21] R. McEliece, E. Posner, E. Rodemich, S. Venkatesh: *The capacity of the Hopfield associative memory*, IEEE transactions on Information Theory, 33 (1987) 461.

[22] G. E. Hinton, T. J. Sejnowski, D. H. Ackley *Boltzmann machines: Constraint satisfaction networks that learn*. Pittsburgh, PA: Carnegie-Mellon University, Department of Computer Science (1984).

[23] D. H. Ackley, G. E. Hinton, T. J. Sejnowski: *A learning algorithm for Boltzmann machines.,* Cognitive science, 9 (1985) 147.

[24] A. Barra, A. Bernacchia, E. Santucci, P. Contucci: *On the equivalence of Hopfield networks and boltzmann machines*, Neural Networks, 34 (2012) 1.

[25] A. Barra, G. Genovese, P. Sollich, D. Tantari: *Phase transitions in Restricted Boltzmann Machines with generic priors*, Physical Review E, 96 (2017) 042156.

[26] E. Agliari, D. Migliozzi, D. Tantari: *Non-convex multi-species Hopfield models* Journal of Statistical Physics 172 (2018) 1247.

[27] G. Genovese, D. Tantari: *Legendre equivalences of spherical Boltzmann machines.* Journal of Physics A: Mathematical and Theoretical, 53 (2020) 094001.

[28] A. Barra, G. Genovese, P. Sollich, D. Tantari: *Phase diagram of restricted Boltzmann machines and generalized Hopfield networks with arbitrary priors.* Physical Review E, 97 (2018) 022310.

[29] A. Decelle, S. Hwang, J. Rocchi, D. Tantari: *Inverse problems for structured datasets using parallel TAP equations and RBM*. arXiv preprint arXiv:1906.11988 (2019).

[30] J. Tubiana, R. Monasson: *Emergence of compositional representations in restricted Boltzmann machines.* Physical Review Letters, 118 (2017) 138301.

[31] D. Krotov, J. J. Hopfield: *Dense associative memory for pattern recognition.* In Advances in neural information processing systems (2016) 1172.

[32] I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio: *Deep learning (Vol. 1).* MIT press, Cabridge (MA) (2016).

Daniele Tantari: è Professore Associato di Fisica Matematica presso l'Università di Bologna. Si occupa di Meccanica Statistica e sue applicazioni in Biologia, Intelligenza Artificiale e Finanza.