ISBN 978-88-255-3620-1
DOI 10.4399/978882553620113
pp. 225-242 (giugno 2020)

Fidarsi degli algoritmi?

GIANDOMENICO BOFFI

Abstract

It is common to be suspicious of such pervasive algorithms today. A generalized attitude of distrust is not justified, but a prudent awareness is reasonable. The essay points out that the mathematical nature of the algorithms in question is uneven and that therefore the trust accorded to them in social, political and legal terms is differentiated. The limited mathematical details can be briefly scrolled through without affecting the overall understanding.

Keywords: A*lgorithms, Google Bombing, Backlink, Machine Learning, Artificial Neural Networks.*

Per un matematico della mia generazione, laureato una quarantina di anni fa, può destare un qualche stupore la diffusione della parola "algoritmo" nel dibattito pubblico odierno. Mi sembra di ricordare che, fino a non molto tempo addietro, il termine fosse usato, correntemente, solo dagli addetti ai lavori (matematici, logici, informatici, ingegneri, ecc.) oppure, occasionalmente, da persone di cultura. Adesso, tanto per esemplificare, sui media si discute spesso e volentieri se gli algoritmi ci renderanno schiavi, quasi che essi avessero una volontà propria, e si alimenta una certa diffidenza, dal che la domanda posta nel titolo.

Vorrei fornire al lettore non specialista, senza pretesa di completezza, qualche elemento per rispondere alla domanda, che ovviamente è riferita agli algoritmi pervasivi in tanti ambiti della vita odierna. Specificamente, vorrei segnalare che la natura matematica di alcuni di questi algoritmi non è omogenea, e che non omogenea è quindi la fiducia ad essi accordabile. Ne derivano esigenze differenziate di natura sociale, politica e giuridica per l'attuale momento storico[1]. Anticipo che la mia risposta personale alla domanda è che non sia giustificato

[1] Non rientra negli scopi del testo la discussione dello sfruttamento illegale di ampie masse di dati reso possibile dalla disponibilità di algoritmi appropriati.

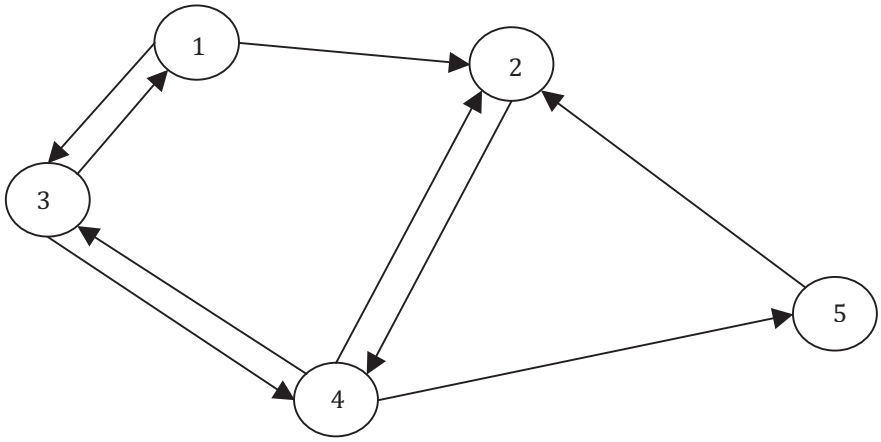
un generalizzato atteggiamento di sfiducia, ma occorra accompagnare la fiducia con una prudente consapevolezza. In fondo è così in tutti gli aspetti della vita umana: la fiducia negli altri è necessaria, ma non può essere cieca, e affinché non sia tale occorre consapevolezza. Ma come può esserci consapevolezza di qualcosa che ci è oscuro? Il tema della fiducia va di pari passo con quello della trasparenza e nel testo accoppierò le due questioni.

L'articolo può essere letto a vari livelli, nel senso che alcuni dettagli matematici, per altro semplici e assai contenuti, possono essere scorsi sommariamente senza inficiare la comprensione complessiva. Ho voluto tuttavia inserire questi dettagli (e la piccola appendice) per offrire al lettore la possibilità – per così dire – di toccare con mano quello di cui concretamente si sta parlando e che, di fatto, è meno misterioso di quel che forse si crede.

Un algoritmo per fare una graduatoria

Un motore di ricerca ha varie esigenze: raggiungere quante più pagine *web* disponibili, indicizzare ogni pagina in maniera tale che sia accessibile a ricerche di parole chiave o di frasi, stilare una graduatoria d'importanza tra le pagine individuate da una ricerca, così da segnalare per prime le più rilevanti. Per soddisfare le sue esigenze, il motore usa vari metodi e algoritmi. Intendiamo focalizzare il discorso sulla parte relativa alla formulazione della graduatoria. Diversi possono essere i modi per stilare una graduatoria d'importanza. Noi concentriamo l'attenzione su uno specifico algoritmo di questo genere, a tutti familiare: l'algoritmo di Google chiamato PageRank[2]. Più particolarmente ancora, supponiamo di volere applicare l'algoritmo alla semplicissima rete di collegamenti illustrata dal disegno seguente (matematicamente, un "grafo orientato").

[2] Il lettore incline alla matematica potrebbe volere consultare ad esempio: K. BRYAN, T. LEISE, *The $ 25,000,000,000 Eigenvector: The Linear Algebra behind Google*, in "SIAM Review", 48 (2006), pp. 569-581; https://epubs.siam.org/doi/abs/10.1137/050623280?journalCode=siread/ (ultimo accesso il 11/09/2019).



I pallini ("nodi") rappresentano pagine *web*. Le frecce ("archi") rappresentano collegamenti iperstestuali (*link*). Ad esempio, nella pagina *1* c'è un *link* alla pagina *2*. Si vuole stabilire la graduatoria di importanza dei nodi assegnando a ogni nodo *i* un numero reale *non negativo x(i)* ("punteggio") tale che a valore maggiore corrisponda rilevanza maggiore. Come assegnare il punteggio *x(i)* a ogni nodo?

PageRank assume anzitutto che il punteggio di un nodo debba riflettere la numerosità dei collegamenti a quel nodo, detti *backlink* (sono esclusi gli auto-collegamenti): il nodo *1* ha un solo *backlink* il nodo *2* ne ha tre, il nodo *3* ne ha due, e così via. PageRank suppone poi anche che siano più significativi i *backlink* provenienti da un nodo importante, avente cioè a sua volta molti *backlink*. Ad esempio, tra i due collegamenti al nodo *4*, suppone che quello proveniente dal nodo *2* sia più importante di quello proveniente dal nodo *3*. Viene allora l'idea di imporre che *x(i)* coincida con la somma delle x*(j)* pertinenti a tutti i nodi *j* che si collegano al nodo *i*.

Poiché tuttavia c'è qualcosa di autoreferenziale in questo (gli archi procedenti da un fissato nodo *i* verso tutti gli altri nodi *j* contribuiscono a determinare i numeri *x(j)* che a loro volta servono a calcolare *x(i)*), PageRank divide ognuno degli addendi *x(j)* per il numero *n(j)* degli archi uscenti dal nodo *j*. Ad esempio, *x(4)* è diviso per *n(4)=3* e quindi il contributo del nodo *4* a ognuno dei tre nodi cui si collega è soltanto un terzo di *x(4)*.

Ricapitolando,

$x(1) = \frac{1}{2}\, x(3)$
$x(2) = \frac{1}{2}\, x(1) + \frac{1}{3}\, x(4) + x(5)$
$x(3) = \frac{1}{2}\, x(1) + \frac{1}{3}\, x(4)$
$x(4) = x(2) + \frac{1}{2}\, x(3)$
$x(5) = \frac{1}{3}\, x(4)$.

Ma si possono trovare davvero numeri non negativi *x(i)* che verifichino tutti i vincoli indicati? Esiste cioè una possibile graduatoria? E se sì, è unica o ce ne sono tante? Quale scegliere nel caso che ce ne sia più di una?

Prima di rispondere, sostiamo un attimo per sottolineare che, per l'algoritmo, una pagina sarà tanto più rilevante quanto più numerosi saranno i collegamenti (da pagine importanti) che riceverà. Nulla garantisce che la numerosità dei collegamenti sia correlata a un effettivo valore dei contenuti, ma d'altra parte in questa maniera si evita di dovere pronunciare giudizi di valore sulle pagine, giudizi che richiederebbero un (presumibilmente arduo) consenso sui criteri di merito.

Sottolineiamo anche che l'algoritmo sarà manipolabile: inondando di collegamenti una medesima pagina, magari insignificante, ne potrò aumentare rapidamente il punteggio, attribuendole grande visibilità. Esemplare al riguardo un caso del 2004, quando un oscuro sito antisemita cominciò improvvisamente a comparire al primo posto della graduatoria per la ricerca associata alla parola *Jew*. Google rifiutò di intervenire sostenendo che ciò fosse dovuto al fatto che la parola *Jew* venisse usata spesso in contesti antisemiti e raccomandò piuttosto di reagire spingendo in alto il punteggio di pagine non antisemite.[3] In anni più recenti, tuttavia, Google ha adottato un atteggiamento più interventista.

Sempre in materia di manipolazioni è d'obbligo citare il cosiddetto *Google bombing*, che però manipola l'algoritmo per assegnare il punteggio più alto a una pagina che non ha nulla a che vedere con il tema della ricerca. Caso famoso nel 2006, quando improvvisamente la ricerca per *miserable failure* cominciò a mostrare al primo posto il sito

[3] Cfr., ad esempio, il seguente indirizzo https://searchenginewatch.com/sew/news/2065217/google-in-controversy-over-top-ranking-for-anti-jewish-site/ (ultimo accesso l'11/09/2019).

del presidente George W. Bush. Anche per casi del genere, tuttavia, Google ha successivamente adottato alcune contromisure.

Torniamo ai nostri quesiti. Notiamo che i vincoli

$$
\begin{aligned}
x(1) &= \tfrac{1}{2}\, x(3) \\
x(2) &= \tfrac{1}{2}\, x(1) + \tfrac{1}{3}\, x(4) + x(5) \\
x(3) &= \tfrac{1}{2}\, x(1) + \tfrac{1}{3}\, x(4) \\
x(4) &= x(2) + \tfrac{1}{2}\, x(3) \\
x(5) &= \tfrac{1}{3}\, x(4)
\end{aligned}
$$

equivalgono a un sistema lineare omogeneo di scolastica memoria

$$
\begin{aligned}
-x(1) + \tfrac{1}{2}\, x(3) &= 0 \\
\tfrac{1}{2}\, x(1) - x(2) + \tfrac{1}{3}\, x(4) + x(5) &= 0 \\
\tfrac{1}{2}\, x(1) - x(3) + \tfrac{1}{3}\, x(4) &= 0 \\
x(2) + \tfrac{1}{2}\, x(3) - x(4) &= 0 \\
\tfrac{1}{3}\, x(4) - x(5) &= 0
\end{aligned}
$$

con cinque equazioni nelle cinque incognite $x(1)$, $x(2)$, $x(3)$, $x(4)$ e $x(5)$.

Per chi ancora ricorda qualcosa dalla scuola, un sistema omogeneo ha sempre almeno una soluzione, quella in cui ogni incognita è zero, ma noi non vogliamo certo assegnare punteggio zero a tutte le pagine della rete di collegamenti. Il fatto che nel sistema indicato la somma dei coefficienti di ogni incognita risulti esattamente uguale a zero (provare per credere) assicura tuttavia che ci siano *infinite* ulteriori soluzioni. Ma questo comporta altre difficoltà: poiché non sono ammessi punteggi negativi, ci sarà almeno una soluzione che assegni numeri non negativi a tutte le incognite? E se ce ne sarà più di una, saranno tutte coerenti fra loro? Coerenti vuol dire che, pur con punteggi numerici diversi, le graduatorie risultanti forniranno sempre lo stesso ordine di rilevanza, sicché ai nostri scopi sarà indifferente scegliere l'una o l'altra.

La circostanza che la rete del nostro esempio sia costituita da un grafo fortemente connesso[4] garantisce in effetti che le infinite soluzioni siano tutte proporzionali tra loro e proporzionali a una di esse

[4] Pensando alle frecce come strade a senso unico, si tratta di un grafo orientato dove è possibile trovare un itinerario per andare da ogni nodo a ogni altro e tornare indietro per un diverso itinerario.

costituita da tutti i punteggi positivi. Per l'esattezza, per ogni numero reale *k* esiste una soluzione

$$x(1) = 2/3\ k \quad x(2) = 7/3\ k \quad x(3) = 4/3\ k \quad x(4) = 3k \quad x(5) = k$$

(come è facile verificare sostituendo alle incognite nel sistema lineare omogeneo) sicché, scegliendo ad esempio $k=1$, si ricavano i punteggi non negativi

$$x(1) = 2/3 \quad x(2) = 7/3 \quad x(3) = 4/3 \quad x(4) = 3 \quad x(5) = 1$$

e la graduatoria

pagina 4
pagina 2
pagina 3
pagina 5
pagina 1.

Si noti che, scegliendo un qualunque altro *k* positivo diverso da *1*, i punteggi *x(i)* numericamente cambiano, ma la graduatoria corrispondente rimane invariata, cioè sono coerenti tra loro tutte le soluzioni con *k* positivo (le uniche che hanno senso per noi, visto che non ammettiamo punteggi negativi). Nella graduatoria trovata, il lettore dovrebbe osservare che il nodo *2* è preceduto dal nodo *4*, pur avendo più *backlink* di esso (tre contro due). Il motivo risiede nel fatto che il nodo *4* riceve in esclusiva l'unico collegamento uscente dall'importante nodo *2*. L'interazione tra i due criteri posti a base dell'algoritmo (tanti *backlink* e *backlink* provenienti da nodi importanti) è alquanto sottile.

Veniamo adesso all'aspetto di PageRank che più riguarda il nostro discorso. Si può avere la sensazione che l'algoritmo sia del tutto trasparente e quindi degno di massima fiducia, a parte eventuali interventi manipolatori. Dopo tutto si tratta soltanto della ricerca delle soluzioni di un sistema lineare omogeneo (anche se nella realtà il numero delle incognite è elevatissimo e occorrono metodi di risoluzione sofisticati). In qualche misura la sensazione è giustificata, come conferma proprio la possibilità di manipolazioni, ma c'è un problema.

Nell'esempio precedente è risultato decisivo, ai fini della stesura di una graduatoria, il fatto che la rete di collegamenti fosse fortemente

connessa, ovvero che godesse di alcune proprietà "buone" – che non menzioniamo neppure – la seguente tabella quadrata *M* (una "matrice" nel linguaggio matematico)

$$\begin{array}{ccccc} -1 & 0 & ½ & 0 & 0 \\ ½ & -1 & 0 & ⅓ & 1 \\ ½ & 0 & -1 & ⅓ & 0 \\ 0 & 1 & ½ & -1 & 0 \\ 0 & 0 & 0 & ⅓ & -1 \end{array}$$

che codifica il precedente sistema lineare omogeneo

$$\begin{aligned} & -x(1) + ½\, x(3) = 0 \\ & ½\, x(1) - x(2) + ⅓\, x(4) + x(5) = 0 \\ & ½\, x(1) - x(3) + ⅓\, x(4) = 0 \\ & x(2) + ½\, x(3) - x(4) = 0 \\ & ⅓\, x(4) - x(5) = 0 \end{aligned}$$

e quindi la rete di collegamenti stessa.

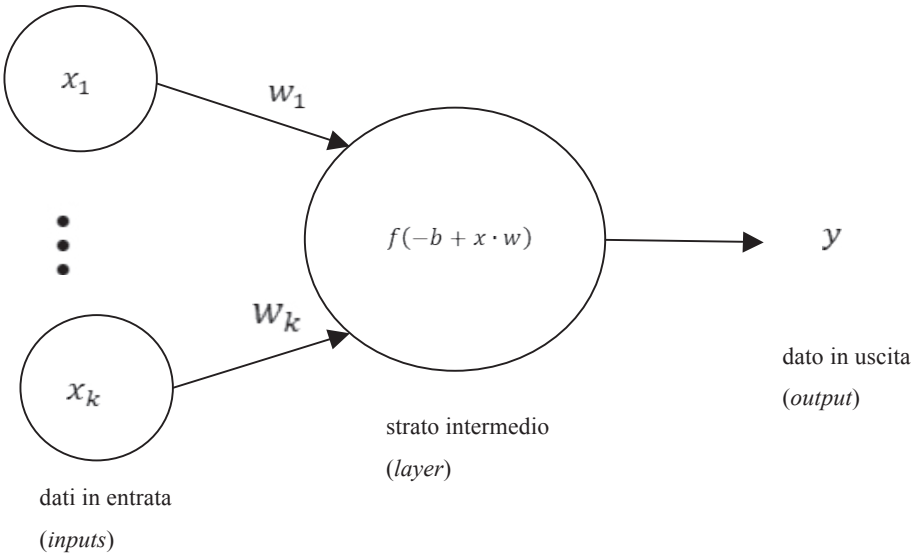
In realtà il *web* è tutt'altro che fortemente connesso e in generale una matrice di collegamenti *M* (con tantissime righe e un ugual numero di colonne) non gode delle dette proprietà "buone". Quel che fa allora l'algoritmo è di sostituire in maniera canonica *M* con una nuova matrice *N* che goda di quelle proprietà e di applicare al *web* originale la graduatoria fornita da *N* (relativa a una rete di collegamenti differente, quella appunto codificata da *N*).

La procedura per passare da *M* a *N* è solo parzialmente nota (è coperta da segreto aziendale) e non sembra esistere una dimostrazione matematica dell'affidabilità per la rete di *M* della graduatoria fornita da *N*. Pertanto, il livello di trasparenza è di fatto più limitato e la fiducia nella graduatoria fornita viene a dipendere essenzialmente dal successo commerciale accumulato dall'algoritmo nel corso degli anni[5].

[5] Richiamo l'attenzione del lettore sul fatto che molta ricerca d'avanguardia sugli algoritmi è oggi condotta in ambito aziendale, non accademico, con tutte le conseguenze del caso.

Un algoritmo capace di apprendere

Il "percettrone" è un algoritmo che costituisce il più semplice esempio di rete neurale artificiale. Le reti neurali artificiali sono così chiamate perché vagamente ispirate al funzionamento della fitta rete costituita dai neuroni del cervello umano. Sono più raramente chiamate reti neuronali, anche se l'aggettivo usato dai biologi per quel che riguarda i neuroni è proprio neuronale. Sono di grande attualità perché si situano al cuore del cosiddetto *machine learning*, il quale presiede a talune delle più note applicazioni dell'intelligenza artificiale, come ad esempio il riconoscimento delle immagini[6]. Il percettrone è inteso come un modello semplificato di un singolo neurone cerebrale, che tipicamente *scarica* (emette un segnale elettrico) se riceve impulsi sufficientemente forti da altri neuroni ad esso collegati. Il modello è rappresentato graficamente dalla figura seguente:



[6] Il sottointeso delle reti neurali artificiali è che ispirarsi alla rete neuronale cerebrale costituisca un buon modo di avvicinarsi alla costruzione di una intelligenza artificiale. Non è ovvio che sia davvero così (ad esempio, la ruota è molto diversa dalle gambe umane, ma assai conveniente per gli spostamenti), anche a prescindere dalla *vexata quaestio* del rapporto mente-cervello. Occorre altresì dire che le reti neurali artificiali sono a loro volta di ispirazione per modellare la nostra comprensione del funzionamento del cervello.

La figura intende rappresentare il calcolo

$$y = f(-b + x \cdot w),$$

dove:

$x \cdot w$ rappresenta la somma di prodotti di numeri reali $x_1 w_1 + x_2 w_2 + \ldots + x_k w_k$ (i coefficienti w_i sono suggestivamente chiamati pesi *sinaptici*); l'ulteriore numero reale b è chiamato valore di soglia; f è la funzione che vale -1 sui numeri reali negativi e vale $+1$ sugli altri numeri reali.

L'idea manifestata dall'effetto combinato dal valore di *soglia* e dal particolare tipo di funzione scelta è quella di un segnale che si attiva da un certo livello in su: y sarà -1 quando $x \cdot w$ risulterà minore di b e scatterà a $+1$ quando $x \cdot w$ risulterà maggiore o uguale a b[7].

Un percettrone può essere usato in due modi.

Nel primo caso, sono fissati pesi sinaptici e valore di soglia, e attribuiamo ogni stringa $(x_1, x_2, \ldots, x_k)$ di dati in entrata a una di due classi: la classe delle stringhe per le quali il percettrone fornisce $y = +1$, oppure la classe delle stringhe cui corrisponde $y = -1$. In altre parole, il percettrone funge da classificatore binario, come suole dirsi.

Nel secondo caso, sono date due classi di stringhe e cerchiamo un percettrone che fornisca sempre $y = +1$ su una delle due classi, sempre $y = -1$ sull'altra; occorre cioè *trovare* pesi sinaptici e valore di soglia che facciano alla bisogna. In effetti non è ovvio *a priori* che percettroni del genere esistano e talvolta non ne esistono proprio. Tuttavia, se esistono, è possibile – a partire da condizioni iniziali di nostra scelta – impostare un calcolo *automatico* che ne individui uno (non necessariamente unico). Si suole quindi dire che la macchina, esami-

[7] Lasciando invariato lo schema, ma scegliendo una funzione f diversa da quella suindicata, invece che di percettrone si tende a parlare, più genericamente, di "neurone artificiale".

nando le due classi di stringhe date, apprende una regola che le distingua: *machine learning* per l'appunto.[8]

Il calcolo automatico procede progressivamente, formulando ipotesi sugli ingredienti cercati a mano a mano che esplora le classi assegnate; ad esempio, se gli ingredienti ipotizzati a un certo momento definiscono un percettrone che in un momento successivo risulta fornire $y = -1$ su una delle stringhe della classe cui è prescritto $y = +1$, il calcolo torna indietro e modifica qualcuno degli ingredienti per eliminare l'errore (si parla di "retropropagazione dell'errore")[9]. In appendice a questo articolo è riportato in dettaglio un esempio assai semplificato della procedura di apprendimento, ambientato nel piano cartesiano, da tutti incontrato a scuola (e non completamente dimenticato, si spera). Il proposito è quello di dare al lettore almeno una sensazione di quel che succede.

Vediamo una situazione concreta. Supponiamo che ogni stringa numerica $(x_1, x_2, \ldots, x_k)$ contenga certi indicatori medici associati a un individuo (i risultati di alcune analisi). Supponiamo altresì che la prima classe, quella cui è prescritto $y = +1$, contenga le stringhe relative a un elevato numero di soggetti che si sa avere sviluppato una determinata malattia nel semestre successivo alle analisi, e che la seconda classe, quella di $y = -1$, altrettanto numerosa, si riferisca a soggetti che non l'hanno sviluppata. Allora, applicando un percettrone, individuato a partire dalle due classi, alla stringa di una nuova persona X, l'eventuale valore $y = +1$ attribuito dal percettrone a quella stringa costituirà un campanello di allarme per X.[10]

[8] Il termine macchina è qui utilizzato nel senso di *macchina algoritmica*; cfr. ad esempio il significato **5.** in http://www.treccani.it/vocabolario/macchina/ (ultimo accesso il 11/09/2019).

[9] A dire il vero, le cose non stanno esattamente così. Poiché le classi di stringhe usate sono generalmente molto ampie e sappiamo che all'interno di grandi masse di dati è inevitabile la presenza di inesattezze (dovute ad esempio ad errate trascrizioni), si ritiene accettabile una macchina che si comporti correttamente quasi sempre, rispetto a un prescelto livello di tolleranza. Una perfetta adeguatezza a tutti i dati, inclusi quelli errati, è anzi ritenuta diminuire il valore predittivo della macchina nei casi nuovi (*overfitting*).

[10] La situazione concreta considerata non è in verità realistica, per vari motivi. Anzitutto occorre pensare non proprio a un percettrone, ma a una macchina algoritmica più complessa, del tipo di quelle discusse dopo. Bisogna poi rilevare la ambi-

Il punto è che, leggendo la stringa di X, magari un medico non sarebbe in grado di capire che il soggetto è a rischio, mentre invece la macchina ha acquisito una qualche capacità *predittiva*, pur non essendo in grado di *spiegare perché* X rischia[11]. Trattandosi comunque di un rischio, ma non di una certezza, la macchina trovata potrebbe anche essere utilizzata per uno *screening* di massa, al fine di segnalare ai medici i casi sui quali concentrare l'attenzione, al fine di determinare la situazione effettiva con ulteriori indagini più approfondite.

Passiamo ad alcune considerazioni sul discorso che ci interessa. La macchina è trasparente? Ci possiamo fidare di lei? Quanto alla prima domanda, occorre intendersi sul significato di trasparenza. Se intendiamo dire che gli ingredienti della macchina sono ispezionabili, la risposta è affermativa (ad esempio posso prendere visione dei pesi sinaptici usati). Se intendiamo dire che la conoscenza degli ingredienti della macchina ci consente di capire il motivo di una predizione relativa a un individuo X qualunque, la risposta è negativa (i pesi sinaptici di cui ho preso visione non mi spiegano, ad esempio, la propensione a sviluppare una certa malattia). È per questo che si dice che una macchina del genere è una *scatola nera*, nel senso di un processo nel quale è opaco il motivo per cui un certo esito è associato ai dati in entrata.

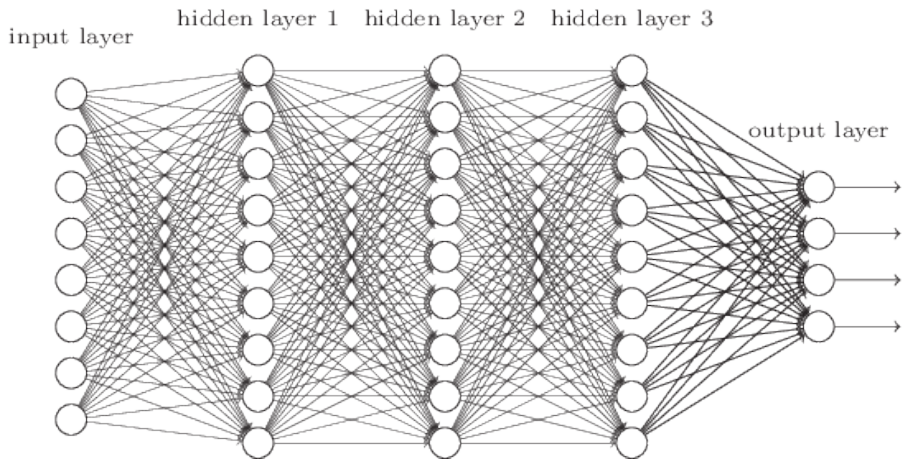
La seconda domanda diventa allora: possiamo fidarci di una siffatta scatola nera? Anche qui è necessario distinguere. Abbiamo già visto che le predizioni della macchina possono risultare errate; molto dipende quindi dal contesto concreto in cui si situano le predizioni e dal rischio che siamo disposti a correre. In un contesto sanitario saremo forse più rigidi che in un contesto finanziario. Ma torneremo su questo nella sezione successiva.

guità del riferimento alla numerosità delle classi usate per addestrare la macchina: servono campioni statisticamente significativi. È anche da supporre che fattori temporali o ambientali non rendano irragionevole l'applicazione della macchina trovata alla ulteriore persona considerata. E così via.

[11] La macchina *spiegherebbe il perché* se fosse in grado di indicare un nesso di *causalità* tra i dati contenuti nella stringa di un X qualunque e la corrispondente previsione fornita, ma essa invece fornisce tale previsione per *associazione* (correlazione). Esistono tuttavia studi mirati a "spiegare" la previsione relativa a un *singolo* X mediante tecniche non ricorrenti a nessi di causalità.

Le reti neurali artificiali

I percettroni hanno capacità molto limitate e le reti neurali artificiali di successo sono costituite da numerosissimi percettroni (o neuroni artificiali), come nella figura seguente, ripresa dal quinto capitolo di un libro liberamente disponibile *online*[12]. Nella figura i neuroni artificiali (e i dati in entrata) sono indicati con pallini e distribuiti in strati (colonne) verticali. Sono omessi i pesi sinaptici sulle frecce. I dati in uscita sono quattro.



La figura va letta da sinistra verso destra, colonna per colonna. La prima colonna contiene i dati in entrata, che vanno ad alimentare i neuroni posti nella seconda colonna. I dati in uscita dalla seconda colonna vanno ad alimentare i neuroni della terza. E così via.

Questa struttura a strati si è rivelata sorprendentemente efficace, più di ogni attesa, e vari studi cercano di capire il perché. Ci sono tuttavia delle criticità:

[12] M. A. NIELSEN, *Neural Networks and Deep Learning*, Determination Press, San Francisco 2015. Scaricabile da http://neuralnetworksanddeeplearning.com/ (ultimo accesso il 11/09/2019).

(1) a parità di insieme di addestramento, si ottengono macchine diverse a seconda del dato iniziale prescelto e della struttura assegnata alla rete; non è ovvio se una macchina sia da ritenersi migliore delle altre;
(2) sempre in fase d'addestramento, come anticipato in nota 10, occorre un equilibrio tra la compatibilità con i dati utilizzati e la consapevolezza che in grandi masse di dati non mancano mai dati incompleti e danneggiati; troppa compatibilità può produrre una macchina poco affidabile per l'esame di casi futuri; in ogni caso le predizioni non sono mai esatte al 100%;
(3) l'esperienza mostra che le macchine risultano efficaci quando operano in contesti molto ben circoscritti, ma non in generale (la trasportabilità è bassa); occorrono cioè macchine diverse per contesti differenti; l'elenco dei campi di applicazione è d'altra parte nutrito, spaziando dal sanitario, al finanziario, all'assicurativo, all'economico, al militare, al ricreativo e ad altro ancora.

La criticità maggiore, che ingloba in qualche modo le altre, è comunque che, come anticipato in nota 12, si tratta di algoritmi predittivi associativi, ma non esplicativi in termini di nessi di causa-effetto. Mentre si può spiegare (almeno in linea di principio) perché PageRank stila una certa graduatoria che a noi può sembrare strana, non si sa spiegare perché una rete neurale artificiale effettui, magari, previsioni mediche sbagliate. Questo ci riconduce al nostro discorso specifico su trasparenza e fiducia.

Certamente una rete neurale artificiale complessa è una scatola nera priva di ogni trasparenza, contenendo milioni di valori numerici. La mancanza di trasparenza è avvertita come un limite anche dagli stessi esperti del settore, i quali stanno cercando di rimediare almeno costruendo delle macchine un po' meno opache ma pur sempre con prestazioni di ottimo livello. Ad esempio, IBM ha presentato alla fine del 2018 una tecnica che, data una rete molto complessa, viene utilizzata per addestrare una rete più semplice a fornire prestazioni sempre più vicine a quelle della rete originale, così da rendere quest'ultima in qualche modo superflua[13].

[13] Cfr.: A. DHURANDHAR, K. SHANMUGAM, R. LUSS, P. OLSEN, *Improving Simple Models with Confidence Profiles*, in «Advances in Neural Information Processing Systems», 31 (2018), pp. 10317-10327. Scaricabile da http://papers.nips.cc/

In termini di fiducia, la combinazione tra l'esistenza di una scatola nera e l'inevitabile presenza di errori da essa compiuti pone il problema di come garantire il fondamentale diritto a una spiegazione laddove il singolo si ritenga danneggiato. È chiaro infatti che il classico diritto alla trasparenza è qui intrinsecamente limitato[14]. In particolare, merita attenzione un fenomeno allarmante: a dispetto di chi afferma che una macchina capace di autoapprendimento può essere più imparziale di un essere umano, una rete neurale artificiale si manifesta talvolta intrisa di pregiudizi razziali, ideologici, ecc., come mostrato da vari casi concreti (vedi l'articolo in nota 15). Vale a dire, ad esempio, che una macchina algoritmica potrebbe esprimere più facilmente parere favorevole alla concessione di un'assicurazione a persone appartenenti a certi gruppi etnici piuttosto che ad altri.

Deviazioni del genere dipendono naturalmente dalla presenza di pregiudizi nei campioni di addestramento, pregiudizi non percepiti da chi ha gestito la fase di addestramento, nonostante tutte le precauzioni suggerite dalle buone pratiche vigenti. E, altrettanto naturalmente, una volta rilevata la deviazione, occorre sostituire del tutto la macchina algoritmica difettosa, dato che non si è in grado di correggere l'interno della scatola nera.

A conclusione di questo articolo, desidero sottolineare con forza che l'intelligenza artificiale contemporanea non è solo reti neurali artificiali: ci sono la programmazione logica, la logica sfumata (*fuzzy*), i metodi semantici, e così via. Ma sono proprio le reti neurali artificiali a suscitare gli interrogativi più grandi in termini di trasparenza e fiducia e su di esse mi sono diffuso, riportando tuttavia in precedenza anche il caso di PageRank per fornire un elemento di contrasto.

Sottolineo altresì che le reti neurali artificiali suscitano anche interrogativi strategici in merito al futuro dell'intelligenza artificiale. Molti

paper/8231-improving-simple-models-with-confidence-profiles/ (ultimo accesso il 11/09/2019).

[14] Cfr.: L. EDWARDS, M. VEALE, *Slave to the Algorithm? Why a 'Right to an Explanation' Is Probably Not the Remedy You Are Looking For*, in «Duke Law & Technology Review», 16 (2017), pp. 18-84. Scaricabile da https://ssrn.com/abstract=2972855 oppure http://dx.doi.org/10.2139/ssrn.2972855 (ultimo accesso il 11/09/2019).

si domandano se una macchina incapace di ragionare in termini di causa ed effetto si possa davvero definire intelligente.[15]

Riferimenti bibliografici

Bryan K., Leise T., *The $ 25,000,000,000 Eigenvector: The Linear Algebra behind Google*, in "SIAM Review", 48 (2006), pp. 569-581. Scaricabile da https://epubs.siam.org/doi/abs/10.1137/050623280?journalCode=siread/ (ultimo accesso il 11/09/2019).

Dhurandhar A., Shanmugam K., Luss R., Olsen p., *Improving Simple Models with Confidence Profiles*, in "Advances in Neural Information Processing Systems", 31 (2018), pp. 10317-10327. Scaricabile da http://papers.nips.cc/paper/8231-improving-simple-models-with-confidence-profiles/ (ultimo accesso il 11/09/2019).

Edwards L., Veale M., *Slave to the Algorithm? Why a 'Right to an Explanation' Is Probably Not the Remedy You Are Looking For*, in "Duke Law & Technology Review", 16 (2017), pp. 18-84. Scaricabile da https://ssrn.com/abstract=2972855 oppure http://dx.doi.org/10.2139/ssrn.2972855 (ultimo accesso il 11/09/2019).

Hartnett K., *To Build Truly Intelligent Machines, Teach Them Cause and Effect*, in "Quanta Magazine", May 15, 2018. https://www.quantamagazine.org/to-build-truly-intelligent-machines-teach-them-cause-and-effect-20180515/ (ultimo accesso il 11/09/2019).

Nielsen M.A., *Neural Networks and Deep Learning*, Determination Press, San Francisco 2015. Scaricabile da http://neuralnetworksanddeeplearning.com/ (ultimo accesso il 11/09/2019).

Sullivan D., *Google In Controversy Over Top-Ranking For Anti-Jewish Site*, in "Search Engine Watch", Apr. 24, 2004. https://www.searchenginewatch.com/sew/news/2065217/google-in-controversy-over-top-ranking-for-anti-jewish-site/ (ultimo accesso il 11/09/2019).

[15] È questa, ad esempio, la posizione del premio Turing Judea Pearl (il premio Turing è una sorta di premio Nobel nel campo dell'informatica). Pearl, nato nel 1936, è stato uno di quelli che più ha contribuito alla storia e al successo delle macchine algoritmiche, ma ritiene che tale successo stia offuscando il vero obbiettivo: la costruzione di macchine capaci di ragionare in termini di causalità. Cfr. l'intervista seguente https://www.quantamagazine.org/to-build-truly-intelligent-machines-teach-them-cause-and-effect-20180515/ (ultimo accesso il 11/09/2019).

Appendice. L'addestramento di un percettrone

Supponiamo di sapere che C_1 sia la classe contenente le due stringhe $s_1 = (1,3)$ e $s_2 = (3,3)$, e C_2 sia la classe contenente le due stringhe $s_3 = (3,1)$ e $s_4 = (1,1)$. Quali possono essere pesi sinaptici e valore di soglia tali che il percettrone $y = f(-b + x \cdot w)$ distingua C_1 ($y = +1$) e C_2 ($y = -1$) ? Qui f è la solita funzione che associa -1 ai numeri reali negativi e $+1$ a tutti gli altri.

Impostiamo inizialmente il nostro percettrone con una *scelta sbagliata*, diciamo $y = f(-4 + x_1 + x_2)$, e vediamo come esso può modificare sé stesso in meglio. [La scelta è sbagliata perché sostituendo per x_1 e x_2 i numeri della stringa s_3 si ottiene $y = +1$ invece del prescritto $y = -1$.]

Poniamo $w(0) = (1,1)$ e $b(0) = 4$ e definiamo la seguente procedura ricorsiva:

$$(*)\quad w(t+1) = w(t) + \big(d(t) - y(t)\big) \cdot s(t)$$
$$(**)\quad b(t+1) = b(t) + \big(d(t) - y(t)\big)$$

dove:

$s(t)$ è la stringa usata per l'addestramento al tempo t
$w(t)$ è la coppia di pesi sinaptici usata al tempo t
$b(t)$ è il valore di soglia usato al tempo t
$y(t)$ è il valore assegnato alla stringa $s(t)$ dal percettrone
$d(t)$ è il valore corretto spettante a $s(t)$,

vale a dire

$$d(t) = \begin{cases} +1 & \text{se } s(t) \text{ è nella classe } C_1 \\ -1 & \text{se } s(t) \text{ è nella classe } C_2 \end{cases}$$

e l'operazione indicata con il puntino è la moltiplicazione di un numero per una coppia di numeri; ad esempio, $2 \cdot (-5,3) = (-10,6)$.

Notiamo che, rispetto a $w(0)$ e $b(0)$, cioè rispetto a $y = f(-4 + x_1 + x_2)$, risulta assegnato a s_1 e s_2 il corretto valore

$+1$, mentre per s_3, come già segnalato, si ha $y(0) = +1$, contro $d(0) = -1$. Applicando $(*)$ e $(**)$ passiamo pertanto a

$$\begin{aligned} w(1) &= (1,1) + \big(-1 - (+1)\big) \cdot (3,1) \\ &= (1,1) + (-2) \cdot (3,1) \\ &= (1,1) + (-6,-2) \\ &= (-5,-1) \end{aligned}$$

e $b(1) = 4 + \big(-1 - (+1)\big) = 2$

che eliminano il problema in s_3.

Il nuovo candidato percettrone è dunque $y = f(-2 - 5x_1 - x_2)$.

Notiamo adesso che il nuovo candidato assegna sì a s_3 (e anche a s_4) il corretto valore -1, ma non è più corretto il valore assegnato a s_1. Infatti, con il nuovo percettrone, per s_1 si ha $y(1) = -1$, contro $d(1) = +1$.

Applicando di nuovo $(*)$ e $(**)$ passiamo quindi a

$$\begin{aligned} w(2) &= (-5,-1) + \big(1 - (-1)\big) \cdot (1,3) \\ &= (-5,-1) + 2 \cdot (1,3) \\ &= (-5,-1) + (2,6) \\ &= (-3,5) \end{aligned}$$

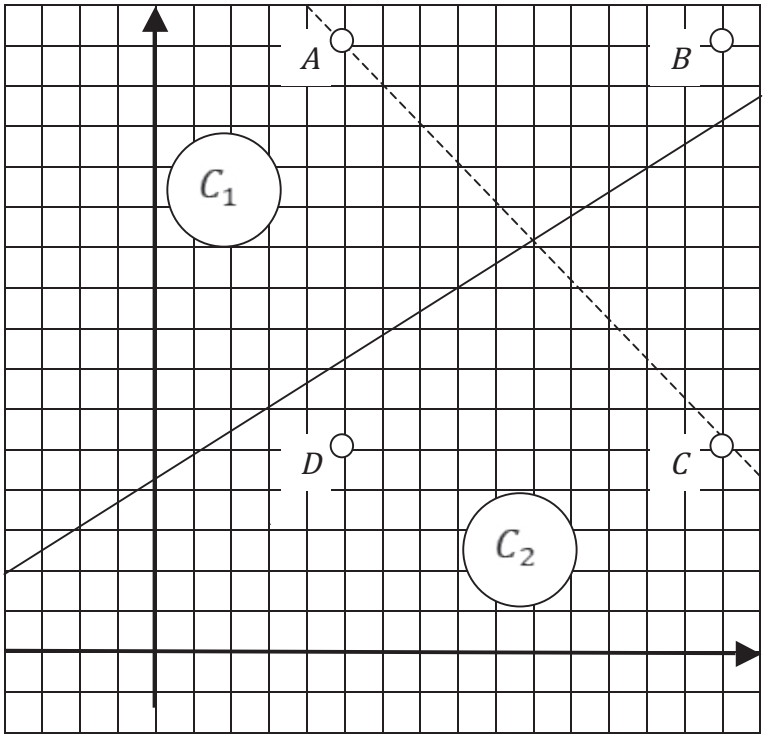
e

$$b(2) = 2 + \big(1 - (-1)\big) = 4$$

che eliminano il problema in s_1 e danno l'ulteriore candidato $y = f(-4 - 3x_1 + 5x_2)$.

Poiché, rispetto a $y = f(-4 - 3x_1 + 5x_2)$, la differenza $d(2) - y(2)$ risulta nulla non solo per s_1, ma anche per tutte le successive stringhe s_2, s_3 e s_4, l'addestramento è finito e $y = f(-4 - 3x_1 + 5x_2)$ è un percettrone che distingue le due classi date C_1 e C_2.

Il piano cartesiano che segue, con ascisse x_1 e ordinate x_2, riassume la situazione (l'unità di misura è pari a 5 quadratini):



I punti A, B, C e D, indicati con un pallino, hanno come coordinate le stringhe s_1, s_2, s_3 e s_4, rispettivamente. La retta tratteggiata ha equazione $x_1 + x_2 = 4$, ottenuta mettendo uguale a zero quel che compare in parentesi nel percettrone *sbagliato* $y = f(-4 + x_1 + x_2)$ da cui siamo partiti; la retta continua ha invece equazione $-3x_1 + 5x_2 = 4$, ottenuta mettendo uguale a zero quel che compare in parentesi nel percettrone finale $y = f(-4 - 3x_1 + 5x_2)$. La retta del percettrone giusto ha le due classi date da parti opposte di essa; la retta del percettrone sbagliato no.

È necessario segnalare che nell'esempio la procedura di addestramento si è conclusa in pochi passi, ma in generale può risultare molto lenta.