Machine Learning: accuratezza, interpretabilità e incertezza

Guido Sanguinetti *School of Informatics, University of Edinburgh, UK; SISSA, Trieste, Italy*

Gli algoritmi di *Machine Learning* sono diventati il motore essenziale dei sistemi di intelligenza artificiale (AI). Nonostante il loro indiscutibile successo, la loro diffusa applicazione a diversi problemi pone molte domande difficili, sia legate al loro impatto sulla società, sia al loro funzionamento tecnico. In questo breve contributo, mi concentro sulla duplice questione della quantificazione dell'incertezza e dell'interpretabilità del modello. Introduco un semplice esperimento teorico per dimostrare che l'importanza di quantificare e scomporre l'incertezza è altamente dipendente dallo scopo del modello, quindi descrivo brevemente i possibili scenari per iniziare ad affrontare questi difficili problemi nell'IA.

Machine Learning algorithms have become the essential engine of artificial intelligence (AI) systems. Despite their unquestionable success, their widespread application to diverse problems poses many difficult questions, both related to their impact on society, and to their technical functioning. In this brief contribution, I focus on the twin issues of uncertainty quantification and model interpretability. I introduce a simple thought experiment to demonstrate that the importance of quantifying and decomposing uncertainty is highly task dependent, and I then briefly describe possible frameworks to start addressing these difficult issues in AI.

Machine Learning: una disciplina in subbuglio

L'ultimo decennio ha assistito ad un'accelerazione esponenziale del progresso nell'intelligenza artificiale

Machine Learning: a discipline in turmoil

The last decade has witnessed an exponential acceleration of progress in Artificial Intelligence (AI).

(IA). Molte pietre miliari apparentemente irraggiungibili sono state conquistate e superate: gli algoritmi di intelligenza artificiale surclassano gli esseri umani in giochi come GO e ATARI [1, 2], svolgono un ruolo centrale in molti processi economici dal settore manufatturiero al settore bancario e stanno facendo grandi passi in avanti nelle scienze, dalla fisica alla biologia. Forse l'*exploit* più impressionante è stata la pubblicazione nel settembre 2020, da parte del quotidiano britannico The Guardian, di un articolo d'opinione interamente scritto da un algoritmo IA [3] L'argomento dell'articolo era una domanda che è sempre più nella mente di molti, scienziati e laici allo stesso modo: "Dobbiamo temere l'ascesa dell'IA?"

Al centro di questo progresso spettacolare c'è l'apprendimento automatico (i.e. *machine learning*, ML). Il ML non è una disciplina nuova: una delle sue principali conferenze, la International Conference on Machine Learning, ha raggiunto la sua trentasettesima edizione nel 2020. Per la maggior parte della sua storia piuttosto oscura, il ML si è concentrato sullo sviluppo di algoritmi informatici che possono essere ottimizzati in ragione dei dati a loro suppliti. Un semplice esempio potrebbe essere l'apprendimento di un classificatore per le immagini associate alle cifre: si cerca una mappa che colleghi i valori di *input* (intensità dei *pixel* in scala di grigi, nel caso delle immagini), con il valore di *output* (l'etichetta della cifra). Il classificatore ha generalmente molti parametri che possono essere regolati in modo tale da minimizzare una funzione di errore su un cosiddetto *training set*, ossia un sottoinsieme dei dati che viene mostrato all'algoritmo; le prestazioni dell'algoritmo vengono quindi valutate empiricamente su un altro set di punti sperimentali, il *test set*, una procedura che può essere giustificata come un'approssimazione Monte Carlo delle prestazioni sui dati estratti dalla effettiva (ma ignota) distribuzione generatrice degli esempi.

Gran parte della ricerca degli ultimi tre decenni è stata dedicata a due compiti principali: sviluppare modi efficaci per riassumere i dati attraverso l'estrazione e la selezione delle loro caratteristiche e lo sviluppo di algoritmi efficienti per eseguire l'inferenza statistica o l'ottimizzazione delle stesse caratteristiche derivate. Esempi del primo tipo di attività includono la vasta letteratura sulla visione artificiale sul rilevamento dei bordi e l'estrazione di caratteristiche, o il campo altrettanto sviluppato dell'analisi grammaticale nell'elaborazione del linguaggio naturale. Esempi del secondo tipo di attività includono algoritmi avanza-

Many seemingly unattainable milestones have been achieved and surpassed: AI algorithms have comprehensively defeated humans at games such as GO and ATARI [1, 2], play a central role in many economic processes from manufacturing to banking, and are making major inroads in sciences from physics to biology. Perhaps most remarkably, in September 2020 the leading UK newspaper The Guardian published an opinion piece entirely authored by an AI algorithm [3] The topic of the article was a question that is increasingly in the mind of many, scientists and lay people alike: "Should we fear the rise of AI?"

At the core of this spectacular progress is Machine Learning (ML). ML is not a new discipline: one of its premiere conferences, the International Conference on Machine Learning, reached its thirty-seventh edition in 2020. For most of its rather obscure history, ML has been focussed on developing computer algorithms which can be tuned by observing data. A simple example could be learning a classifier for digits: one seeks a map that connects input values (pixel intensities in grayscale, in the case of digits), with the output value (the digit label). The classifier generally has many parameters which can be tuned in such a way as to minimise an error function on a so-called *training set*, a subset of the data which is shown to the algorithm; the performance of the algorithm is then empirically evaluated on a separate set of data points, the *test set*, a procedure which can be justified as a Monte Carlo approximation of the performance on data drawn from the true (unknown) data generating distribution.

Much of the research of the last three decades has been dedicated to two major tasks: developing effective ways to summarize data through feature extraction and selection, and developing efficient algorithms to perform statistical inference or optimisation on the features derived. Examples of the first type of task include the vast literature in computer vision on edge detection and feature extraction, or the equally developed field of grammatical parsing in natural language processing. Examples of the second type of task include variational and advanced Markov chain Monte Carlo methods for Bayesian infer-

ti di Markov Chain Monte Carlo per l'inferenza bayesiana, o metodi di ottimizzazione e proiezione casuale per le *kernel machines*. Come molti altri campi scientifici, in particolare nell'informatica, la ricerca in ML è stata condotta principalmente in un'industria artigianale composta da piccoli gruppi formati da un ricercatore principale ed alcuni studenti. Il principale interesse industriale per il ML era molto raro, e in effetti Microsoft era probabilmente l'unico attore industriale significativo nel ML prima del 2010. Anche la più grande conferenza sul ML, NIPS (ora NeurIPS), raggiungeva a malapena 1000 partecipanti e si svolgeva in un'atmosfera informale simile a un ricongiungimento familiare.

Questo stato di cose è cambiato drasticamente nell'ultimo decennio. Ora, tutte le più grandi aziende tecnologiche impiegano legioni di ricercatori in ML e molti professori hanno lasciato il mondo accademico per lavori più redditizi, potenzialmente con un serio effetto a catena sulla capacità delle università di formare nuovi ricercatori in ML. Forse ancor più impressionante, le aziende utilizzano *software* per registrare i propri dipendenti in massa alle principali conferenze di ML già una frazione di secondo dopo l'apertura della registrazione: in pratica, la registrazione a conferenze come NeurIPS o ICML è diventata impossibile per persone che non stanno presentando un lavoro, chiudendo di fatto il campo agli estranei che, in passato, avrebbero potuto partecipare alla conferenza per imparare un pò di più sul ML. Come è successo tutto questo?

L'ascesa del *deep learning*

Il *deep learning* è universalmente riconosciuto come il fattore scatenante nella rivoluzione del ML [4]. L'idea alla base del *deep learning* è che, proprio come il cervello umano sembra imparare solo da esempi senza essere stato progettato per compiti specifici, anche un algoritmo di apprendimento dovrebbe apprendere in modo puramente basato sui dati, evitando qualsiasi ingegnerizzazione delle caratteristiche dei dati ma semplicemente estraendole dagli stessi. Il deep learning lo fa applicando ripetutamente trasformazioni non lineari adattabili ai dati grezzi (si veda il riquadro reti neurali profonde per una breve introduzione ai concetti salienti a riguardo). I modelli costruiti in questo modo sono spesso altamente parametrizzati (cioè hanno molti più parametri liberi che dati suppliti) e di ottimizzazione molto difficile. Quence, or optimisation and random projection methods for kernel machines. Like many other scientific fields, particularly in computer science, ML research was carried out primarily in a cottage industry of small groups of one PI and a few students. Major industrial interest in ML was very rare, and indeed Microsoft was likely the only significant industrial player in ML before 2010. Even the largest ML conference, NIPS (now NeurIPS), hardly reached 1000 participants and encouraged a family reunion-like informal atmosphere.

This state of affairs changed dramatically in the last decade. Now, all of the largest tech companies employ legions of ML researchers, and many professors have left academia for more lucrative jobs, potentially with a serious knock-on effect on the ability of Universities to train new ML researchers. Perhaps more impressively, companies use software tools to register their own employees en masse onto the premiere ML conferences only a fraction of a second after registration is open. Practically, registering for conferences such as NeurIPS or ICML has become impossible for people who are not presenting a paper, effectively closing the field to outsiders who might once have gone to a conference to learn a bit more about ML. How did this all happen?

The rise of deep learning

Deep learning is universally recognised as the trigger of the ML revolution [4]. The idea behind deep learning is that, just as the human brain seems to learn only from examples without being engineered for specific tasks, so should a learning algorithm also learn in a purely data-driven fashion, avoiding any feature engineering and simply extracting features itself. Deep learning does so by repeatedly applying tuneable non-linear transformations to the raw data (see box Deep Neural Networks for a very brief introduction to the fundamental concepts). The models constructed in this way often are highly over-parametrised (i.e., they have many more tuneable parameters than data) and of very difficult optimisation. This and other problems meant that deep neural networks were rapidly abandoned after a short-lived

sto e altri problemi hanno fatto sì che le reti neurali profonde siano state rapidamente abbandonate dopo una popolarità di breve durata negli anni ottanta.

Verso la fine del primo decennio di questo secolo, l'interesse per le reti neurali profonde è tornato a crescere, principalmente grazie al massiccio aumento della disponibilità di dati per addestrarle e dei significativi progressi nell'*hardware* dei computer (in particolare le unità di elaborazione grafica, GPU) che permettono l'ottimizzazione su larga scala. Nel giro di pochi anni, gli algoritmi di *deep learning* hanno vinto con un margine considerevole molte competizioni di visione artificiale e hanno stabilito nuovi standard nel riconoscimento vocale, nell'elaborazione del linguaggio naturale e nella traduzione automatica, per citare solo alcune delle loro principali aree di applicazione. Ciò ha generato sia un enorme interesse industriale, sia un'esplosione di attività di ricerca che non ha precedenti nella storia dell'informatica o, forse, della scienza.

Interpretabilità e incertezza: sono veramente importanti?

Una delle principali conseguenze del passaggio al *deep learning* è stata la perdita dell'interpretabilità umana degli algoritmi. Mentre una caratteristica dei modelli realizzati a mano era che questi venivano generalmente progettati per riflettere il problema in questione, le reti neurali profonde seguono pedissequamente i dati, rimescolandoli attraverso diverse trasformazioni non lineari e rappresentazioni di apprendimento che –sebbene statisticamente efficaci– di solito non corrispondono a caratteristiche interpretabili dall'uomo. Inoltre, per costruzione, le reti neurali profonde hanno uno spazio molto complesso di configurazioni di parametri quasi equivalenti: questo implica che, anche se fosse possibile identificare una spiegazione plausibile per una previsione, molte altre spiegazioni (corrispondenti a configurazioni di parametri equivalenti) potrebbero essere ugualmente valide.

Strettamente correlato a questo è il problema dell'incertezza: molti approcci statistici classici di ML hanno consentito esplicitamente l'incorporazione dell'incertezza negli algoritmi, propagando il rumore attraverso l'algoritmo e quantificando l'incertezza nel risultato finale. Le reti profonde, d'altra parte, sono progettate per approssimare una funzione de-

popularity in the eighties.

Towards the end of the first decade of the century, interest in deep neural networks was resurgent, primarily due to massive increases in data availability, and significant advances in computer hardware (graphic processing units, GPUs) which could support optimisation on such large-scale tasks. Within a few years, deep learning algorithms were winning by a considerable margin many computer vision competitions, and setting new standards in speech recognition, natural language processing and machine translation, to name only some of the major areas of application. This has engendered both huge industrial interest, and in an explosion of research activity which is unprecedented in the history of computer science or, possibly, science.

Interpretability and uncertainty: do they matter?

A major consequence of the shift towards deep learning has been the loss of human interpretability of the algorithms. While hand-crafted features and models were generally designed to reflect the problem at hand, deep neural networks follow the data, scrambling through several non-linear transformations and learning representations that, while statistically effective, do not usually correspond to human-interpretable features. Additionally, by construction deep neural networks have a very complex landscape of nearly equivalent parameter configurations: this implies that, even if it were possible to identify one plausible explanation for a prediction, many other explanations (corresponding to equivalent parameter configurations) might be equally valid.

Closely related is the issue of uncertainty: many classical statistical ML approaches explicitly enabled the incorporation of uncertainty in the algorithms, propagating noise through the algorithm and quantifying the uncertainty in the final result. Deep networks, on the other hand, are designed to approximate an unknown *deterministic* function of the input, and in

terministica sconosciuta dell'*input*, ed in generale si limitano a predizioni puntuali. Questo è vero sia per l'incertezza sull'*output* finale, sia per l'incertezza sulla configurazione dei parametri che hanno portato a tale previsione: nessuna delle due può essere generalmente quantificata o scomposta lungo fattori contributivi interpretabili.

Tutto questo importa? La questione è aperta al dibattito. Ad esempio, l'UE impone legalmente un livello (minimo) di interpretabilità per qualsiasi utilizzo dell'IA in un'ampia gamma di settori che coinvolgono il benessere umano e la società. Tuttavia, molti sostenitori di alto profilo dell'apprendimento profondo hanno fortemente discusso contro l'interpretabilità, sulla base del fatto che sono preferibili prestazioni migliori. A volte vengono utilizzati esempi medici: preferiamo un medico che può curarci meglio o un medico che può spiegare meglio la motivazione alla base della diagnosi?

Direi che l'importanza dell'interpretabilità dipende in gran parte dal compito dato alla rete. Per espandere il mio argomento, consideriamo le seguenti previsioni fittizie:

1. Il computer dice: "questa immagine contiene un gatto seduto su un tavolo";
2. Il computer dice: "Good Morning = Buongiorno = Guten Morgen";
3. Il computer dice: "in base alla tua genetica ed al tuo stile di vita, la tua aspettativa di vita residua è di 20 anni, 5 mesi e tre giorni".

Questo esempio è chiaramente costruito per essere estremo, con lo scenario 3 radicalmente diverso da 1 e 2; tuttavia, credo che esso evidenzi una serie di questioni su cui penso valga la pena riflettere. Prima di tutto, lo scenario 3 implica una previsione reale su un fatto futuro, al contrario di una previsione statistica, la cui bontà sia eventualmente valutabile su un *test set.* In quanto tale, lo scenario 3 non può essere verificato indipendentemente in alcun modo (se non aspettando i 20 anni prescritti). Quindi, sembra almeno ragionevole aspettarsi di essere in grado di capire come sia stata raggiunta tale conclusione. In secondo luogo, proprio perché lo scenario tre appartiene al futuro, potrebbe essere possibile intervenire per modificare il risultato. Tuttavia, come si può capire quali azioni potrebbero essere più efficaci, se non si sa cosa abbia contribuito al raggiungimento della conclusione? È evidente che sia l'interpretabilità

general limit themselves to point predictions. This is true both for uncertainty on the final predicted output, and uncertainty on the configuration of parameters that led to the prediction: neither generally can be quantified or decomposed along interpretable contributing factors.

Does all of this matter? The question is open for debate. For example, the EU mandates legally a (minimum) level of interpretability for any usage of AI in a wide range of areas involving human welfare and society. However, many high-profile proponents of deep learning have powerfully argued against interpretability, on the grounds that better performance is to be preferred. Medical examples are sometimes used: do we prefer a doctor that can cure you better, or a doctor who can explain better the motivation behind the diagnosis?

I would argue that the importance of interpretability is largely task-dependent. To expand on my argument, let's consider the following fictitious predictions:

1. Computer says: "this image contains a cat sitting on a table";
2. Computer says: "Good morning = Buongiorno = Guten Morgen";
3. Computer says: "Based on your genetics and lifestyle, your remaining life expectation is 20 years, 5 months and three days".

This example is clearly constructed to be extreme, with scenario 3 being radically different from 1 and 2; however, I believe it highlights a number of issues which I think are worth reflecting upon. First of all, scenario 3 involves a real prediction about a fact in the future, as opposed to a statistical prediction to be evaluated as held-out data. As such, scenario 3 cannot be verified independently in any way (except by waiting the prescribed 20 years). Hence, it seems at least reasonable to expect to be able to understand how the conclusion was reached. Secondly, precisely because scenario three is in the future, it may be possible to take action to modify the outcome. However, how can I understand which actions might be most effective, if I don't know what contributed to the conclusion? It is evident that both interpretability *AND* uncertainty quantification are needed whenever rational decision making is involved. Finally, while scenarios 1 and 2

sia la quantificazione dell'incertezza siano entrambe necessarie ogni volta che è coinvolto un processo decisionale razionale. Infine, mentre gli scenari 1 e 2 non hanno un impatto immediato sulla vita di una persona, lo scenario 3 sì, e quindi si può sostenere che la sua importanza diretta per qualcuno sia molto più alta.

Vorrei proporre che queste tre semplici domande possano essere poste ad un qualsiasi tipo di previsione:

1. è probabile che la previsione abbia un impatto significativo sulla vita delle persone?
2. la previsione può essere facilmente convalidata in modo indipendente?
3. può essere intrapresa un'azione per modificare il risultato previsto (se in futuro)?

Si noti che prima domanda è già al centro del requisito legale per la spiegabilità nell'UE. Le risposte a queste tre domande insieme, a mio parere, determinano in gran parte se l'interpretabilità e la quantificazione dell'incertezza sono una bella aggiunta o una componente indispensabile di qualsiasi sistema di IA.

Interpretabilità: due approcci possibili

Naturalmente, non sono il primo a discutere questioni di interpretabilità in ML. In effetti, l'IA spiegabile (XAI) è un'area di ricerca in rapida espansione a sé stante e diverse direzioni di ricerca interessanti sono in fase di sviluppo attivo (vedere ad esempio [9] per una recente indagine sul panorama della ricerca in questo campo). Sebbene sia impossibile rendere giustizia al campo, descriverò brevemente due approcci alternativi al problema.

Metodi *post-hoc*

Diversi metodi mirano a spiegare il processo decisionale mediante algoritmi di apprendimento profondo in modo post-hoc. In pratica, ciò consiste nell'individuare quali caratteristiche iniziali abbiano maggiormente contribuito ad una decisione. Ad esempio, il metodo di *layer-wise relevance propagation* [10] spiega la classificazione di un'immagine monitorando l'impatto che l'attivazione di un singolo pixel ha sulla classificazione finale. In questo modo, il metodo

do not immediately have an impact on a person's life, scenario 3 does, and so one may argue that its direct importance to someone is much higher.

I would like to propose that these three simple questions could be asked of any type of prediction:

1. is the prediction likely to have a significant impact on people's life?
2. can the prediction be easily independently validated?
3. can action be taken to modify the predicted outcome (if in the future)?

Notice that Q1 is already at the core of the legal requirement for explainability in the EU. The answers to these three questions together, in my opinion, largely determine whether interpretability and uncertainty quantification are a nice addition or an indispensable component of any AI system.

Interpretability: two possible approaches

Naturally, I am not the first to discuss issues of interpretability in ML. In fact, explainable AI (XAI) is a rapidly expanding research area in its own right and several interesting research directions are being actively developed (see e.g. [9] for a recent survey of the research landscape in this field). While it is impossible to do justice to the field, I will briefly describe two alternative approaches to the problem.

Post-hoc methods

Several methods aim at explaining decision making by deep learning algorithms in a post-hoc way. In practice, this consists in identifying which initial features contributed most to a decision. For example, the method of *layer-wise relevance propagation* [10] explains the classification of an image by tracking the impact that an individual pixel activation has on the final classification. In this way, the method can highlight (for example by colouring them) the most

può evidenziare (ad esempio colorandoli) i *pixel* più rilevanti di un'immagine, dando una spiegazione intuitiva dell'origine del risultato finale. Naturalmente, l'idea non è specifica per le immagini, ma applicabile a qualsiasi *input* ad alta dimensionalità.

Il principale punto di forza di questa classe di metodi è che non compromettono le prestazioni, poiché vengono applicati dopo che l'algoritmo di ML è stato addestrato. Tuttavia, rimangono molte sfide aperte: innanzitutto sembra dubbioso che tali metodi consentano agli utenti di comprendere l'incertezza nelle previsioni, poiché vengono semplicemente applicati post-hoc su un algoritmo pre-addestrato (deterministico). In secondo luogo, le caratteristiche rilevanti sono raramente facilmente riconducibili a categorie comprensibili dall'uomo. Ad esempio, due immagini di cani potrebbero essere entrambe classificate correttamente a causa di sottoinsiemi di *pixel* completamente diversi, –uno a causa della coda e l'altro a causa delle orecchie– rendendo difficile trovare una spiegazione generale di cosa sia un cane secondo l'algoritmo di ML. Ancora più importante, modificare leggermente l'immagine di un cane potrebbe cambiare completamente l'insieme di caratteristiche che l'algoritmo ritiene essere rilevanti per il suo riconoscimento. Ciò ha un impatto drammatico sulla terza domanda della sezione precedente: come possiamo agire riguardo a una previsione, quando la spiegazione offerta è fragile alle perturbazioni?

Interpretabile per costruzione: modelli bayesiani gerarchici

Un'alternativa naturale è progettare algoritmi che siano interpretabili direttamente, ricollegandosi essenzialmente a ciò che i modellisti statistici hanno fatto per decenni. Una struttura particolarmente attraente è fornita dai modelli bayesiani gerarchici (HBM, vedere ad esempio [11]). Tali modelli scompongono esplicitamente l'incertezza lungo componenti ampie e gerarchicamente organizzate. Questa idea è al centro di molte applicazioni mediche stratificate: ad esempio, la variazione totale di un determinato biomarcatore all'interno di una popolazione potrebbe essere scomposta in diverse fonti parzialmente annidate, a partire dalla variazione dovuta a sesso, etnia, fattori di stile di vita e terminando con intrinseca variabilità a livello del singolo individuo. Questi modelli, formulati in termini di probabilità condizionali e solitamente addestrati tramite inferenza sulla

relevant pixels in an image, giving an intuitive explanation for the origins of the final result. Naturally, the idea is not specific to images, but could be applied to any high-dimensional input.

The major strength of this class of methods is that they do not compromise performance, as they are applied after the ML method has been trained. Nevertheless, several open challenges remain. First of all, it seems dubious that such methods would enable users to understand uncertainty in predictions, since they are simply applied post-hoc to a pre-trained (deterministic) method. Secondly, the relevant features are rarely easily relatable to human-understandable categories. For example, two images of dogs could be both correctly classified due to completely different subsets of pixels, one because of the tail, and the other because of the ears, making it difficult to find a general explanation of what is a dog according to the ML algorithm. More importantly, slightly modifying the image of one dog could completely change the set of features that the algorithm finds to be relevant. This impacts dramatically on the third question in the previous section: how can we act about a prediction, when the explanation offered is fragile to perturbations?

Interpretable by design: hierarchical Bayesian models

A natural alternative is to design algorithms to be interpretable directly, essentially reconnecting with what statistical modellers have been doing for decades. A particularly attractive framework is provided by hierarchical Bayesian models (HBMs, see for example [11]). Such models explicitly decompose uncertainty along broad, hierarchically organised components. This idea is at the core of many stratified medicine applications: for example, the total variation in a certain biomarker within a population might be decomposed across several partially nested sources, starting from variation due to gender, ethnicity, lifestyle factors, and ending with intrinsic variability at the level of the single individual. These models, formulated in terms of conditional probabilities and usually trained via posterior inference, precisely quantify and decompose uncertainty in predictions, and are still very

distribuzione a posteriori, quantificano e scompongono con precisione l'incertezza nelle previsioni e sono ancora ampiamente utilizzati in particolare nelle applicazioni mediche.

Naturalmente, tali modelli possono essere utilizzati solo laddove esiste una solida conoscenza di base del fenomeno in esame, poiché sono intrinsecamente sviluppati su misura ed adattati alle applicazioni specifiche. Non avrebbe molto senso sviluppare HBM per immagini naturali, dove non c'è una chiara comprensione del processo generativo sottostante. Ancora più importante, gli HBM sono spesso vincolati a forme funzionali parametriche e relativamente semplici, a causa delle difficoltà computazionali nell'esecuzione dell'inferenza bayesiana in modelli complessi. Tuttavia, recenti sviluppi –ad esempio nelle applicazioni alla genomica *high-throughput* [12, 13]– stanno iniziando ad allentare questi vincoli, introducendo dipendenze funzionali non lineari, guidate dai dati all'interno dei modelli e sfruttando strumenti avanzati di ML come inferenza variazionale stocastica [14].

Discussione

Le tecnologie di Intelligenza Artificiale alimentate da algoritmi di *machine learning* hanno già avuto un impatto enorme sulla nostra vita e sulla nostra società e probabilmente continueranno a farlo nel prossimo futuro. Dati gli enormi livelli di investimento nelle tecnologie da parte di attori sia pubblici che privati è chiaro che il loro utilizzo diventerà sempre più diffuso. È quindi essenziale che potenziali aree problematiche in questa tecnologia, o nella sua applicazione, siano identificate e corrette tempestivamente.

In questo breve contributo, ho tentato di spiegare come la mancanza di spiegabilità e quantificazione dell'incertezza siano, a mio parere, aspetti estremamente problematici delle attuali tecnologie che derivano intrinsecamente dalla progettazione interamente data-driven dei moderni algoritmi di ML. Tali problemi sono fondamentalmente alla radice anche di molti altri problemi, ampiamente pubblicizzati, connessi all'uso delle tecnologie di ML. L'ingiustizia, ad esempio, è una conseguenza quasi inevitabile della mancanza di interpretabilità: è estremamente difficile implementare le nozioni di equità in algoritmi che si basano interamente su dati presi da una società ingiusta. Un altro problema importante è la difficoltà nell'adattare gli algoritmi a cambiamenti (noti) delle condizioni al contorno: mentre per gli algoritmi fatti

widely employed particularly in medical applications.

Naturally, such models can only be used where there is a solid knowledge base, as they are intrinsically bespoke and tailored to specific applications. It would not make much sense to develop HBMs for natural images, where there is no clear understanding of the underlying generative process. More importantly, HBMs are often constrained to parametric and relatively simple functional forms, due to computational difficulties in performing Bayesian inference in complex models. However, recent developments, for example in applications to high-throughput genomics [12, 13], are starting to relax these constraints, introducing non-linear, data-driven functional dependencies within models and taking advantage of advanced ML tools such as stochastic variational inference [14].

Discussion

AI technologies powered by ML algorithms have already massively impacted our life and society, and are likely to continue to do so in the foreseeable future. Given the enormous levels of investment by both public and private actors in the technologies, it is clear that their usage will become more and more widespread. It is therefore essential that potential fault lines either in the technology or in its application are identified and corrected early.

In this brief contribution, I have attempted to explain how lack of explainability and uncertainty quantification are, in my opinion, hugely problematic aspects of current technologies which arise intrinsically from the entirely data-driven design of modern ML algorithms. Such issues are fundamentally at the root also of many other, widely publicised problems with ML technologies. Unfairness, for example, is an almost unavoidable consequence of lack of interpretability: it is extremely difficult to implement notions of fairness in algorithms which are entirely reliant on data taken from an unfair society. Another major issue is the difficulty in adapting algorithms to (known) changes in conditions: while for handcrafted algorithms this is in principle straightforward, as they usually contain explicit, knowledge derived models of how external covariates affect predictions,

a mano questo è in linea di principio semplice, poiché di solito tali algoritmi contengono modelli espliciti e derivati, (anche) da tale conoscenza, per molte `scatole nere` di impiego nel ML, la soluzione è raccogliere un nuovo *training set*, cosa che potrebbe essere irrealizzabile.

La comunità ML è ben consapevole di questi problemi ed in effetti sono stati fatti diversi tentativi per risolverli. Nella parte finale di questa relazione ho evidenziato due linee di ricerca diverse ma complementari a tal proposito. Entrambe sono oggettivamente lontane dall'aver quadrato l'intrattabile cerchio del mantenimento dell'accuratezza fornendo interpretabilità e quantificazione dell'incertezza, ma sono, a mio parere, passi preziosi nella giusta direzione.

Tuttavia, è importante sottolineare che la maggior parte dei metodi di ML, inclusi tutti quelli qui descritti, sono di natura fondamentalmente correlativa: i modelli sono individuati nei dati associati ai risultati di interesse, ma la ricostruzione del flusso causale rimane oltre il loro scopo. Incorporare idee e metodi del ML negli approcci di inferenza causale [15] è certamente una grande sfida per il futuro dell'IA

for many black-box ML approaches the solution is to gather a new training set, which may be infeasible.

The ML community is well-aware of these problems, and indeed several attempts are being made to address them. In the final part of this report, I have highlighted two different but complementary lines of research. Both are objectively far from having squared the intractable circle of retaining accuracy while providing interpretability and quantifying uncertainty, but they are, in my opinion, valuable steps in the right direction.

Nevertheless, it is important to underline that most ML methods, including all the ones described here, are fundamentally correlative in nature: patterns are spotted in the data which are associated with outcomes of interest, but reconstructing the causal flow remains beyond their scope. Incorporating ideas and methods from ML in causal inference approaches [15] is certainly a grand challenge for the future of AI.

Reti neurali profonde

Come funzionano le reti neurali profonde? Le reti profonde sono costituite da un gran numero di cosiddetti `neuroni`, organizzati in `strati`, collegati tra loro in modo gerarchico e *feed-forward*. I dati di *input* sono essi stessi visti come una collezione di attivazioni neuronali: tipicamente, ad ogni dimensione dell'input corrisponde un neurone e per convenzione lo strato di input è considerato il primo livello. Nella sua forma più elementare, ogni neurone nello strato t riceve segnali di *input* da (un sottoinsieme) di neuroni nello strato $t-1$, e quindi esegue le seguenti operazioni: combina linearmente questi segnali (utilizzando un insieme di coefficienti regolabili chiamati `pesi sinaptici`), quindi applica una funzione non lineare $\sigma\colon \mathbb{R} \to \mathbb{R}$ allo scalare risultante.

Deep neural networks

How do deep neural networks work? Deep networks are made up of large numbers of so-called `neurons` which are organised in `layers`, connected to each other in a hierarchical, feed-forward manner. Input data points are themselves seen as a collection of neuronal activations: typically, each input dimension corresponds to one neuron, and by convention the input layer is considered the first layer. In its most basic form, each neuron in layer t receives inputs from (a subset) of neurons at layer $t-1$, and then performs the following operations: it combines linearly the inputs (using a set of tuneable coefficients called `weights`), and then applies a nonlinear function $\sigma\colon \mathbb{R} \to \mathbb{R}$ to the resulting scalar.

Possono essere usati diversi tipi di non linearità (le cosiddette `unità`): i primi tentativi usavano unità sigmoidali come la tangente iperbolica, mentre ad oggi si preferiscono le unità lineari rettificate (i.e. ReLU). È importante sottolineare che le unità devono essere differenziabili quasi ovunque affinchè si possano applicare metodi di ottimizzazione basati sul calcolo del gradiente. In sintesi, ogni neurone i allo strato t produce un *output*

$$o_{it} = \sigma\left(\sum_{j \in \mathscr{J}_{it}} w_{it} o_{j(t-1)}\right)$$

dove $\mathscr{J}_{it}$ è l'insieme di neuroni allo strato $t-1$ che alimentano il neurone i allo strato t. Lo strato finale implementa il classificatore: nel caso di un classificatore binario, tale strato è costituito da un singolo neurone che esegue una regressione lineare generalizzata e fornisce un numero compreso tra 0 e 1. Da questa prospettiva, si possono immaginare gli strati intermedi come dispositivi efficaci che permettono di far imparare alla rete nel suo insieme una `rappresentazione` dei dati ottimale, che li renda separabili da un classificatore lineare. L'apprendimento viene eseguito tramite la discesa del gradiente su una funzione di errore scelta in modo appropriato (tipicamente si usa la *cross-entropy* per la classificazione). Indicata come $y_i \in \{0,1\}$ l'etichetta associata all'istanza di addestramento i e come $f(\mathbf{x}_i, \mathbf{w}) \in [0,1]$ la previsione ad essa associata della rete neurale, la funzione da minimizzare è

$$L(Y,X,W) = \sum_{i \in \mathscr{I}} \Big[y_i \log\left(f(\mathbf{x}_i, \mathbf{w})\right) + (1-y_i)\left(\log\left(1 - f(\mathbf{x}_i, \mathbf{w})\right)\right) \Big]$$

dove $\mathscr{I}$ è l'insieme di indici delle istanze di addestramento e X, Y e W indicano –collettivamente e rispettivamente– dati di input per l'addestramento, relative etichette di addestramento ed i pesi della rete. Poiché le unità sono scelte per essere differenziabili quasi ovunque, ne consegue che anche la funzione da minimizzare è quasi ovunque differenziabile, quindi la minimizzazione può essere ottimizzato rispetto ai pesi mediante metodi di discesa del gradiente.

Different types of nonlinearity can be used (so called `units`): early efforts used sigmoidal units such as hyperbolic tangent, while more recently rectified linear units are preferred. Importantly, units need to be differentiable almost everywhere to apply gradient-based optimisation methods. In summary, each neuron i at layer t produces an output

$$o_{it} = \sigma\left(\sum_{j \in \mathscr{J}_{it}} w_{it} o_{j(t-1)}\right)$$

where $\mathscr{J}_{it}$ is the set of neurons at layer $t-1$ feeding into neuron i at layer t. The final layer implements the classifier: in the case of a binary classifier, it consists of a single neuron performing generalised linear regression and outputting a number between 0 and 1. In this sense, one can view the purpose of the intermediate layers as a device to learn a `representation` of the data that makes it optimally separable by a linear classifier. The learning is performed by gradient descent on an appropriately chosen error function (typically, the *cross-entropy* loss function for classification). Denoting as $y_i \in \{0,1\}$ the label associated with training instance i, and as $f(\mathbf{x}_i, \mathbf{w}) \in [0,1]$ the associated prediction of the neural network, the loss function is

$$L(Y,X,W) = \sum_{i \in \mathscr{I}} \Big[y_i \log\left(f(\mathbf{x}_i, \mathbf{w})\right) + (1-y_i)\left(\log\left(1 - f(\mathbf{x}_i, \mathbf{w})\right)\right) \Big]$$

where $\mathscr{I}$ is the index set of the training instances and X, Y, and W denote collectively training inputs, training labels and weights of the network. Since the units are chosen to be differentiable almost everywhere, it follows that the loss function is also a.e. differentiable, and can be optimised w.r.t. the weights by gradient descent methods.

I gradienti vengono calcolati automaticamente usando la differenziazione simbolica e, per ottenere la scalabilità, vengono usati sottoinsiemi casuali di dati (detti *minibatches*) usati per calcolare prontamente ogni passo del gradiente, portando a una `discesa stocastica lungo il gradiente`, che appare anche più efficace nell'evitare minimi locali. In pratica, i minibatch non vengono scelti del tutto casuale ma mediante un protocollo che assicuri che l'intero set di addestramento venga utilizzato iterativamente attraverso quella che viene chiamata `epoca` dell'ottimizzazione.
Le reti neurali profonde possono essere pensate come uno schema di approssimazione di funzioni basato su funzioni di base adattive; è noto da molto tempo che le reti profonde possono approssimare arbitrariamente bene qualsiasi funzione regolare [5]. Ma come possono funzionare in pratica? Dopo tutto, per costruzione, le reti profonde hanno un numero molto elevato di pesi e un numero quasi infinito di simmetrie (ad esempio, la permutazione dei neuroni all'interno di uno strato dovrebbe portare a soluzioni identiche). Empiricamente, la presenza di molti ottimi locali o l'overfitting non sembrano essere veri problemi (sebbene le moderne reti neurali siano piene di trucchi euristici per evitarli). Recenti lavori teorici hanno anche dimostrato che, in un limite termodinamico adeguato, il problema degli ottimi locali è naturalmente evitato e la convergenza globale è ottenuta dalla discesa stocastica lungo il gradiente [6, 7, 8].

Gradients are computed automatically using symbolic differentiation, and, in order to achieve scalability, random subsets of the data (*minibatches*) are used to compute each gradient step, leading to a `stochastic gradient descent`, which also appears more effective in avoiding local optima. In practice, minibatches are not chosen entirely at random but in a schedule that ensures that the whole training set is used iteratively through what is called an `epoch` of the optimisation.
Deep neural networks can be thought of as a function approximation scheme based on adaptive basis functions; it has been known for a long time that deep networks can approximate arbitrarily well any smooth function [5]. But how can it possibly work in practice? After all, by construction deep networks have a very large number of weights and a nearly infinite number of symmetries (e.g., permuting neurons within a layer should lead to identical solutions). Empirically, local optima or overfitting do not appear to be problems (although modern neural networks are replete with heuristic tricks to avoid such problems). Recent theoretical work has also shown that, in a suitable thermodynamic limit, local optima issues are naturally avoided, and global convergence is achieved by stochastic gradient descent [6, 7, 8].

[1] V. Mnih, et al.: *Human-level control through deep reinforcement learning*, Nature 518 (2015) 529.

[2] D. Silver, et al.: *Mastering the game of go without human knowledge*, Nature 550 (2017) 354.

[3] https://www.theguardian.com/commentisfree/2020/sep/08/robot-wrote-this-article-gpt-3

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

[5] G. Cybenko: *Approximation by superpositions of a sigmoidal function*, Math. of Contr. Sign. and Sys. 4 (1989), 303.

[6] G.M. Rotskoff, E. Vanden-Eijnden: *Neural networks as interacting particle systems: Asymptotic convexity of the loss landscape and universal scaling of the approximation error*, arXiv preprint arXiv:1805.00915 (2018).

[7] S. Mei, A. Montanari, P.M. Nguyen: *A mean field view of the landscape of two-layer neural networks*, Proc. Natl. Acad. Sci. USA, 115 (2018) E7665.

[8] S.S. Du et al., *Gradient descent finds global minima of deep neural networks*, arXiv preprint arXiv:1811.03804 (2018).

[9] W. Samek, et al., *Explainable AI: interpreting, explaining and visualizing deep learning*, W. Samek, G. Montavon, A. Vedaldi, L. K Hansen, K.-R. Müller (Eds.) Springer-Nature, Berlin (2019) 16.

[10] S. Bach et al.: *On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation*, PLoS One 10 (2015) e0130140.

[11] A. Gelman et al., *Bayesian data analysis*, CRC Press, New York (2013).

[12] N. Eling et al., *Correcting the mean-variance dependency for differential variability testing using single-cell RNA sequencing data*, Cell systems, 7 (2018) 284.

[13] C.A. Kapourani, R. Argelaguet, G. Sanguinetti, C.A. Vallejos, *scMET: Bayesian modelling of DNA methylation heterogeneity at single-cell resolution*, bioRxiv, Cold Spring Harbor Laboratory (2020).

[14] R. Ranganath, S. Gerrish, D. Blei, *Black box variational inference*, Proc. of Seventeenth International Conference on Artificial Intelligence and Statistics, PMLR (2014) 814.

[15] J. Pearl, *Bayesianism and causality, or, why I am only a half-Bayesian*, Foundations of bayesianism (2001) 19.

Guido Sanguinetti: è professore di Fisica Applicata ed ha la Cattedra di Data-Science all'International School for Advanced Studies (SISSA) in Trieste. È inoltre Professor of Computational Bioinformatics alla School of Informatics, University of Edimburgh, dove ha insegnato dal 2010. I suoi interessi risiedono nel machine learning applicato a sistemi biomedicali, in particolare problemi di systems biology ed high-throughput biology.

Guido Sanguinetti: is Professor of Applied Physics and Chair of Data Science at the International School for Advanced Studies (SISSA) in Trieste. He is also Professor of Computational Bioinformatics at the School of Informatics, University of Edinburgh, UK, where he has worked since 2010. His interests are in machine learning applied to biomedical systems, in particular problems in systems and high-throughput biology.