

ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA

CAMPUS DI CESENA

DIPARTIMENTO DI INGEGNERIA DELL'ENERGIA
ELETTRICA E DELL'INFORMAZIONE
"GUGLIELMO MARCONI"

CORSO DI LAUREA IN INGEGNERIA BIOMEDICA

**IL RUOLO DELL'INTELLIGENZA
ARTIFICIALE NELLE NEUROSCIENZE**

Elaborato in

Fondamenti di Strumentazione Biomedica

Relatore
Prof. Stefano Diciotti

Presentata da
Andrea Betti

Correlatrice
Federica Giorgini

Anno Accademico 2021/2022

Indice

Introduzione.....	1
Capitolo 1 – L’Intelligenza Artificiale.....	4
1.1 Cenni Storici.....	4
1.2 Machine Learning e Deep Learning.....	10
1.3 Supervised Learning.....	15
1.3.1 Support Vector Machines (SVM).....	17
1.3.2 Decision Tree e Random Forest.....	18
1.3.3 Naive Bayes.....	21
1.3.4 Regressione Lineare e Logistica.....	23
1.3.5 Reti Neurali Artificiali.....	25
1.4 Unsupervised e Reinforcement Learning.....	32
1.5 Analisi delle Prestazioni.....	33
Capitolo 2 – L’Intelligenza Artificiale nelle Neuroscienze.....	41
2.1 Supervised Learning per le Neuroscienze.....	45
2.1.1 Farmaci: Proprietà e Riutilizzo.....	46
2.1.2 Studi Clinici.....	53
2.1.3 Attività Neurale.....	60
2.1.4 Segmentazione dei Tumori.....	64
2.1.5 Predizione della Diagnosi e della Prognosi.....	67
2.2 Selezione delle Caratteristiche.....	71
2.3 Modellistica del Cervello.....	72
Capitolo 3 – L’Intelligenza Artificiale per la Salute Mentale.....	74
3.1 L’IA per il Disturbo Depressivo Maggiore.....	75
3.2 L’IA per i Disturbi dello Spettro Autistico.....	87

Conclusioni.....	100
Bibliografia.....	102
Sitografia.....	107

Introduzione

L'Intelligenza Artificiale (IA) è una disciplina moderna, appartenente alla branca dell'informatica, che negli ultimi anni ha contribuito, nonché permesso, lo sviluppo di molteplici settori, da quello del commercio a quello elettronico, a quello politico. Sono state fornite diverse definizioni dell'IA nel corso degli anni; ad esempio, l'Oxford Dictionary la definisce come:

“La teoria e lo sviluppo di sistemi informatici capaci di eseguire compiti che richiedono normalmente l'intelligenza umana, come la percezione visiva, il riconoscimento vocale, il processo decisionale e la traduzione tra lingue.”

L'IA è quindi quella disciplina che, tramite lo studio e l'implementazione di fondamenti teorici ed algoritmi matematici, mira a fornire ad un calcolatore elettronico abilità e performance proprie dell'intelligenza umana. Nata attorno alla metà degli anni '50, si è lentamente evoluta nel tempo, fino ad esplodere negli ultimi due decenni. I motivi che hanno fatto sì che l'IA esplodesse negli anni recenti sono diversi, anche se tra di essi due sono particolarmente rilevanti:

- I dati: grazie alla notevole diffusione e sviluppo della tecnologia, al giorno d'oggi si sono rese disponibili grosse moli di dati, dai formati di testo a immagini e video. Se da un lato ciò ha reso estremamente complicata e dispendiosa l'analisi di questi dati da parte dell'uomo in termini di tempo, di denaro ed anche a seguito della loro grande variabilità, dall'altro lato ciò ha fornito una fonte immensa di materiale indispensabile, come si vedrà nei capitoli successivi, per allenare e valutare i modelli che si basano sull'IA, migliorando le loro performance. Al tempo stesso, un sempre più frequente utilizzo di codifiche delle informazioni e standard a livello internazionale e mondiale ha permesso che i dati, acquisiti

in un determinato posto e momento, potessero essere usati facilmente, previo rispetto della privacy, da chiunque.

- La potenza di calcolo: l'analisi della grande quantità di dati sopra citata e l'impiego di algoritmi sempre più sofisticati e complessi richiede potenze di calcolo sempre maggiori. Questo problema, negli anni, è stato affrontato in maniere diverse e complementari: da un lato, seguendo la tendenza evolutiva dei processori, si sono migliorate in generale le prestazioni delle CPU (unità di elaborazione centrale) e GPU (unità di elaborazione grafica); in parallelo a ciò, si sono sviluppati algoritmi di deep learning ottimizzati per sfruttare l'accelerazione della GPU, mentre le CPU e le librerie di software che vengono eseguite su di esse si sono evolute per diventare molto più performanti nelle attività di deep learning come, ad esempio, attraverso profonde ottimizzazioni del software e l'aggiunta di hardware IA dedicato.

Grazie alla sua recente evoluzione, l'IA ha iniziato ad essere sempre più usata nell'ambito biomedico, ovvero il campo che riguarda la biologia, la medicina e le loro ricerche integrate. Lo scopo della seguente tesi sarà quello di illustrare lo stato dell'arte, tramite alcune tra le più recenti ricerche, riguardante le applicazioni dell'intelligenza artificiale nelle neuroscienze, mostrando gli ultimi sviluppi e novità relativi a questo campo ed evidenziando i vantaggi, sempre più crescenti, che l'IA sta apportando ai vari aspetti delle neuroscienze. Nello specifico, si approfondirà il tema della salute mentale, ovvero del benessere emozionale, psicologico e sociale di un individuo, con particolare attenzione ai relativi disturbi, come il disturbo depressivo maggiore (DDM) e i disturbi dello spettro autistico (DSA).

Nel capitolo 1 si introdurrà l'intelligenza artificiale, descrivendo i principali tratti del suo percorso evolutivo ed approfondendo i temi del machine learning e del deep learning, analizzando alcuni degli

algoritmi più usati nel campo delle neuroscienze e spiegando in che modo vengono descritte le prestazioni di questi algoritmi.

Nel capitolo 2 ci si focalizzerà sulle neuroscienze, mostrando alcuni esempi che l'IA riscontra in questo campo, tra cui prevenzione, diagnosi e trattamento delle patologie, scoperta e studio delle proprietà di nuovi farmaci e studi clinici.

Infine, nel capitolo 3 si scenderà più nel dettaglio nel campo della salute mentale, concentrandosi in particolar modo sul disturbo depressivo maggiore (DDM) e sui disturbi dello spettro autistico (DSA), riportando alcuni recenti studi sull'impiego dell'IA in queste patologie.

Capitolo 1: L'Intelligenza Artificiale

1.1 Cenni Storici

Sebbene gli eventi che hanno portato alla nascita della Intelligenza Artificiale (IA) siano numerosi, il termine Intelligenza Artificiale fu utilizzato per la prima volta da John McCarthy nel 1956 durante il convegno estivo tenutosi presso il Dartmouth College di Hanover nel New Hampshire, ed è a quell'anno che quindi si fa risalire l'effettiva nascita dell'IA come disciplina di natura propria.

Tuttavia, prima di proseguire con la storia dell'IA, vale la pena ricordare alcuni eventi, antecedenti al 1956, che hanno influenzato l'evoluzione dell'IA. Tra questi, vi sono i due articoli “On Computable Numbers, With An Application To The Entscheidungsproblem” (Turing, 1936) e “Computing Machinery and Intelligence” (Turing, 2009), rispettivamente del 1936 e del 1950, entrambi redatti da Alan Turing. Il primo articolo è importante perché dà una risposta all'Entscheidungsproblem (“Problema della decisione”), ovvero ad un quesito proposto dal matematico tedesco Hilbert nel 1928, il quale richiedeva di mostrare un “metodo meccanico” (ovvero un algoritmo) che, fornito di tutti gli assiomi della matematica e che ricevesse in input un enunciato, fosse in grado di dire se tale enunciato fosse universalmente valido, ovvero dimostrabile o meno a partire dagli assiomi usando le regole della logica. Per risolvere questo problema, Turing ipotizzò una macchina, oggi detta “Macchina di Turing”, sulla base della quale dimostrò come esista un enunciato a cui nessun “metodo meccanico” possa dare una risposta certa, dando quindi una risposta negativa all'Entscheidungsproblem. Se da un lato questo lavoro pose le basi della “teoria della calcolabilità”, disciplina comune sia alla matematica e

all'informatica, strettamente legata al concetto di algoritmo e molto importante nell'ambito dell'IA, questo lavoro dimostrò anche l'esistenza di limiti nel campo dell'IA.

Il secondo articolo, "Computing Machinery and Intelligence", introdusse invece il "Test di Turing", cioè un criterio utilizzato per stabilire se un calcolatore esibisce un comportamento intelligente, ovvero se è ovvero in grado di pensare, di concatenare idee e di esprimerle. Sebbene questo criterio sia stato a lungo utilizzato per valutare quanto simile ad un essere umano potesse essere una IA, negli ultimi anni si sono sollevati diversi dubbi sulla sua validità. Per esempio, un'IA che conosce istantaneamente una risposta, potrebbe imparare ad aspettare qualche secondo prima di rispondere, per essere più simile ad un essere umano. Tuttavia, questo comportamento non la rende più "intelligente", anzi, ne limita le sue performance.

Altri due eventi antecedenti alla nascita dell'IA sono i due lavori svolti da McCulloch e Pitts (McCulloch & Pitts, 1943), e da Marvin Minsky e Dean Edmonds. Il primo gruppo creò una rete di neuroni, ognuno con due soli stati possibili ("acceso" o "spento"), il cui passaggio da uno stato all'altro era determinato dalla presenza/assenza di stimoli proveniente dai neuroni vicini. Questa rete fu ritenuta il primo lavoro strettamente attinente al campo dell'IA. Sempre McCulloch e Pitts arrivarono a dimostrare, inoltre, che qualsiasi funzione computabile e tutti i connettivi logici (and, or...) possono essere implementati da una rete neurale artificiale. Il secondo gruppo, composto da Marvin Minsky e Dean Edmonds, creò quella che è definita la prima rete neurale artificiale, denominata SNARC (Stochastic Neural Analog Reinforcement Calculator), una rete casualmente connessa con circa 40 sinapsi in totale.

Nonostante questi primi sporadici esempi di IA e della sua implementazione, l'IA inizia ad assumere una sua relativa importanza dal convegno tenutosi nel 1956 al Dartmouth College, dove si propose di "esaminare la congettura che ogni aspetto

dell'apprendimento od ogni altra caratteristica dell'intelligenza possa essere, in linea di principio, descritto in modo tanto preciso da poter far sì che una macchina lo simuli” (*La grande scienza. Intelligenza artificiale in «Storia della Scienza»*, s.d.); sulla base di ciò, McCarthy propose la creazione di un gruppo di dieci studiosi il cui compito sarebbe stato quello di creare, entro i successivi due mesi, una macchina in grado di imitare l'apprendimento e l'intelligenza umana. Sebbene questo obiettivo non venne raggiunto, questo congresso fu l'occasione per far conoscere tra di loro diversi esperti dell'informatica, i quali gettarono le basi e promossero nei primi anni lo sviluppo dell'IA.

In questo stesso convegno, ad attirare l'attenzione fu anche un programma, denominato “Logic Theorist” (LT), sviluppato da Allen Newell, Herbert Simon e Cliff Shaw. Questo programma, che verrà poi riconosciuto come “il primo programma di intelligenza artificiale”, era in grado di dimostrare teoremi partendo dai principi della matematica.

Negli anni successivi, complice una grande espansione dei supporti informatici usati, si registrarono una serie di successi nell'ambito dell'IA. Nel 1958, ad opera dello psicologo statunitense Frank Rosenblatt, nacque il percettrone, il quale aveva lo scopo di simulare un neurone; nei fatti, il percettrone era un classificatore binario che, ricevendo in ingresso due o più input, li elaborava fornendo in uscita un valore, che, nei primi modelli del percettrone, poteva assumere solo i valori +1 o -1. Dal 1956, la ricerca nel campo dell'IA assunse sostanzialmente due tendenze differenti:

- Da un lato vi era un gruppo, guidato da Newell, Simon e Shaw, il quale era interessato allo sviluppo di un elaboratore basato sulla simulazione degli aspetti cognitivi e dell'intelligenza umana. Un primo risultato fu il “General Problem Solver” (GPS), un programma che, a differenza del precedente LT, mirava a riprodurre i metodi di risoluzione dei problemi usati normalmente dagli esseri umani. Questo primo gruppo si basava quindi sul “paradigma della simulazione”.

- Dall'altro lato vi era invece quel gruppo interessato al puro miglioramento delle prestazioni degli elaboratori, indipendentemente dal processo utilizzato per fare ciò. A differenza del primo gruppo, questo secondo gruppo si basava sul “paradigma della prestazione”.

In parallelo a questi due gruppi, nel 1958 McCarthy ideò e sviluppò “Lisp”, un linguaggio di programmazione che assunse un ruolo dominante nei primi decenni dell'intelligenza artificiale. Negli stessi anni, McCarthy descrisse l'”Advice Taker”, un programma ideale, mai effettivamente realizzato, che può essere visto come il primo sistema completo di IA, progettato per trovare soluzione a problemi di natura non più strettamente matematica.

Tuttavia, a questi primi anni ricchi di enfasi e di iniziali successi, seguirono successivamente anni di progressivo rallentamento nel campo dell'IA dovuti a diversi motivi: nonostante i primi risultati fossero promettenti, erano pur sempre limitati alla risoluzione di problemi o strettamente matematici, o problemi non matematici ma comunque molto semplici. Sebbene si credesse che la risoluzione di problemi più ampi e complessi necessitasse semplicemente di competenti hardware e di memoria sempre più potenti e capienti, così non si rivelò essere. Inoltre, nacquero i primi problemi nel campo delle reti neurali artificiali, quando Minsky e Papert, nel loro documento “Perceptrons” del 1969, dimostrarono come un semplice percettrone non fosse in grado di implementare la funzione logica XOR. Al tempo stesso fallirono diversi progetti di traduzione automatica; questi fallimenti erano dovuti al fatto che questi primi modelli di traduzione automatica basavano la loro traduzione sul piano puramente sintattico, traducendo quindi letteralmente parola per parola, e non anche sul piano semantico, quindi sul significato della frase. Questi fallimenti portarono alla riduzione degli investimenti da parte del governo degli Stati Uniti d'America e da quello inglese in questi progetti.

Per far fronte a questo periodo di rallentamento, negli anni successivi si decise di cambiare approccio: prima di tutto, invece

che porsi come obiettivo la costruzione di una macchina dotata di intelligenza generale, in grado di spaziare in qualsiasi dominio e risolvere qualsiasi problema, ci si concentrò sulla costruzione di macchine specializzate in specifici ambiti; in secondo luogo, invece che cercare di creare macchine basate solo sulla conoscenza teorica dell'ambiente in cui avrebbero dovuto operare (metodologia definita "approccio debole"), si decise di creare delle macchine basate su una conoscenza più approfondita e dettagliata del settore in cui sarebbero state impiegate. Queste macchine, definite "sistemi esperti", basavano la loro conoscenza sempre sulla conoscenza teorica del loro dominio di applicazione, ma a ciò vi si affiancarono anche le specifiche regole pratiche ed euristiche, basate sull'esperienza propria tipica di un esperto umano del settore. Queste regole pratiche, opportunamente codificate ed implementate all'interno del "sistema esperto", avrebbero permesso a quest'ultimo di risolvere i problemi simili a quelli affrontati dall'esperto umano. Uno dei primi "sistemi esperti" fu il MYCIN, un programma, scritto in Lisp, in grado di diagnosticare infezioni batteriche del sangue e di suggerire quindi l'uso di un particolare antibiotico, il cui dosaggio era rapportato al peso del paziente. Per far ciò, il programma MYCIN si avvaleva di un insieme di circa 600 regole e di una serie di domande, poste al medico, la cui risposta doveva essere un semplice sì/no. Basandosi su ciò, alla fine il programma stilava una lista di possibili batteri colpevoli, ordinati in base alla loro probabilità di essere l'effettiva causa della infezione, il ragionamento fatto dietro ciascuna diagnosi e il corrispondente corso di trattamento con antibiotico.

In concomitanza a ciò si vide il progressivo ritorno alla rete neurale grazie alla reinvenzione e implementazione con successo dell'algoritmo di apprendimento delle reti neurali detto "back-propagation" (retropropagazione), basato sulla retropropagazione dell'errore e già ideato circa quindici anni prima da Bryson e Ho.

Negli ultimi anni, complice anche il progresso tecnologico, l'IA è letteralmente esplosa, arrivando ad abbracciare una moltitudine di

domini. Di seguito sono riportati solo due tra i più famosi campi di applicazione in cui l'IA ha raggiunto livelli degni di nota:

- Veicoli a guida autonoma: tra i primi esempi di applicazione dell'IA nel settore automobilistico vi è Alvin (Autonomous Land Vehicle In a Neural Network), una rete neurale di tre layer progettata per il “road following”, allenata mediante l'algoritmo di back-propagation usando delle simulazioni di immagini di strade e successivamente testata, con successo, a bordo dell'autovettura dell'università Carnegie Mellon. Nell'ultimo decennio, questo campo ha subito grossi sviluppi grazie all'interesse di numerose aziende, di cui Tesla ne è forse l'esempio più lampante.
- Gioco: già nel 1957 Herbert Simon stimò che, nel giro di un decennio, le macchine basate sull'IA sarebbero state in grado di competere con i campioni di scacchi. Questa previsione fu parzialmente giusta: anche se non si avverò nell'arco di un decennio come inizialmente ipotizzato, nel 1996, a Filadelfia, la macchina Deep Blue, prodotta da IBM e progettata appositamente per giocare a scacchi, batté l'allora campione del mondo di scacchi Kasparov. Altra pietra miliare nel mondo dei giochi si ebbe nel 2015, quando il software AlphaGo, sviluppato da Google Deepmind, sconfisse l'allora campione europeo Fan Hui in una sfida del gioco Go, battendolo su tutte le cinque partite disputatesi. Divulgato al pubblico nel 2016, questo avvenimento fece molto scalpore perché il gioco Go era in precedenza considerato, data la sua complessità e l'estremamente grande numero di mosse possibili, al di fuori della portata delle tecniche di machine e deep learning del tempo.

1.2 Machine Learning e Deep Learning

Due branche molto importanti e molto usate all'interno dell'intelligenza artificiale sono il machine learning (apprendimento automatico) e il deep learning (apprendimento profondo).

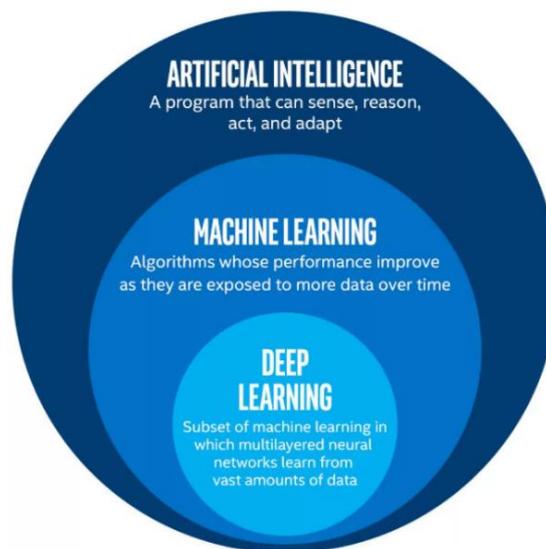


FIGURA 1.2.1: IA, ML, DL (TOUGER, 2018)

Così come per l'IA, anche per il machine learning sono state date, nel corso degli anni, diverse definizioni. Tra le prime formulate, vi è quella di Arthur Samuel, un informatico americano pioniere nel campo dell'IA, che nel 1959 definì il machine learning come “il campo di studio che fornisce ai computer l'abilità di imparare senza essere programmati in modo esplicito”. Quindi, il machine learning può essere visto come l'insieme di tecniche (algoritmi) che permettono ad un computer di imparare dai dati ad effettuare determinate operazioni senza che venga esplicitamente programmato un software specifico per fare tali operazioni. Gli ingredienti principali del machine learning possono essere riassunti in tre gruppi: gli algoritmi, che permettono alla macchina di apprendere autonomamente; i dati, necessari alla macchina da un

lato per imparare a risolvere il problema e dall'altro lato utili per valutare le prestazioni della macchina; infine, la potenza di calcolo, indispensabile per analizzare i dati e per far funzionare gli algoritmi.

Per quanto concerne l'approccio di machine learning alla risoluzione dei problemi, questo si contrappone a quello che invece è l'approccio tradizionale. Un programma che implementa un approccio tradizionale risolve un problema attraverso lo studio del problema, cioè analizzando quest'ultimo cercando delle caratteristiche e dei pattern tipici di esso, la scrittura del programma, in cui, basandosi sulle caratteristiche e pattern individuati nella fase precedente, si scrive il programma/algoritmo (ovvero un insieme di istruzioni il cui scopo è quello di risolvere il problema) ed infine valutando il programma. In caso di corretto funzionamento, il programma può essere usato; in caso di non corretto funzionamento, si ripetono i primi due passaggi, alla ricerca di errori all'interno del programma o dovuti ad una errata analisi del problema. Questo approccio è sinteticamente illustrato nella seguente figura (Fig. 1.2.2):

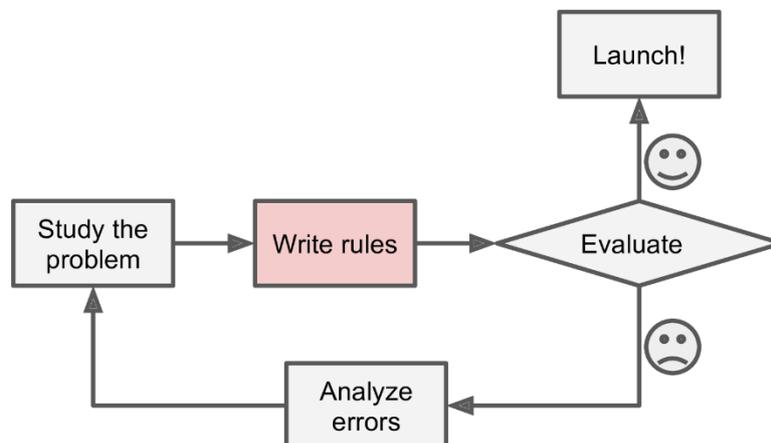


FIGURA 1.2.2: APPROCCIO TRADIZIONALE (GÉRON, 2019)

I problemi che possono insorgere utilizzando questo approccio possono essere di diversa natura: in prima istanza, può succedere che il problema sia non banale e che, di conseguenza, il programma risulti eccessivamente lungo; ciò fa sì che sia l'analisi del problema

che l'esecuzione del programma diventino troppo dispendiosi dal punto di vista temporale. Può inoltre succedere che lo studio del problema sia infruttuoso o troppo complesso, cosa che porterebbe a non poter scrivere l'algoritmo. Quello che si può fare, anche al fine di compensare le eventuali lacune nella conoscenza del problema, è utilizzare l'approccio di machine learning (Fig. 1.2.3), il quale è in grado di apprendere automaticamente dai dati l'algoritmo per la risoluzione del problema.

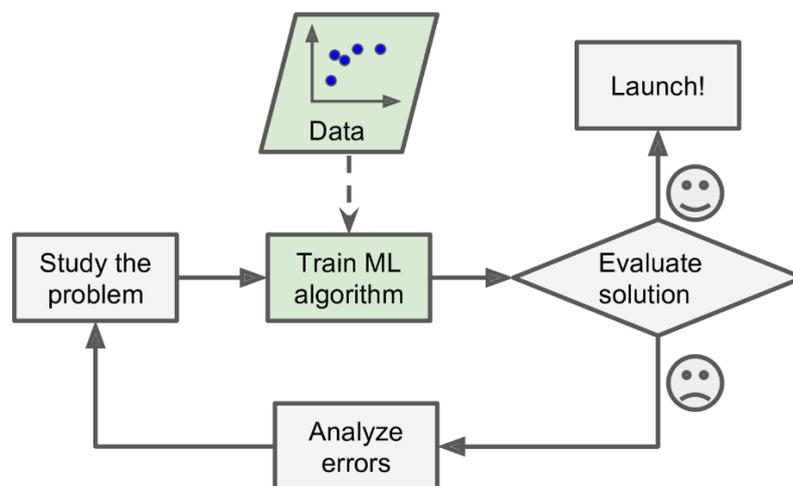


FIGURA 1.2.3: APPROCCIO DI MACHINE LEARNING (GÉRON, 2019)

I vantaggi nell'utilizzo di questo approccio possono essere diversi: come detto in precedenza, può essere usato in caso di incompleta comprensione del problema, colmandola con i dati; il programma che ne scaturisce è spesso più breve, più accurato e più facile da mantenere.

Inoltre, basandosi sull'approccio di machine learning, si è in grado di automatizzare il processo di adattamento del programma a nuovi dati, senza doverlo modificare manualmente ogni volta (Fig. 1.2.4).

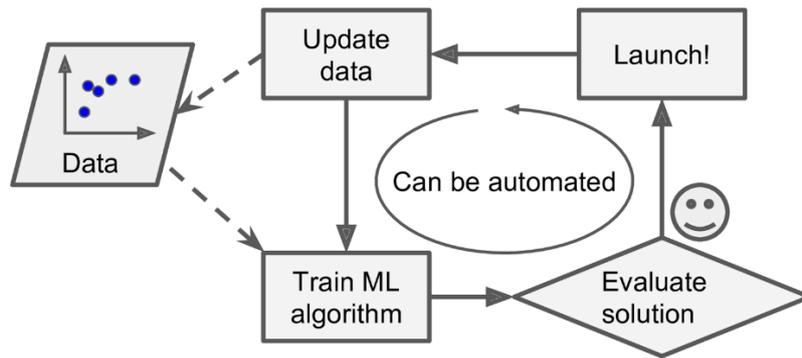


FIGURA 1.2.4: ADATTAMENTO AUTOMATICO (GÉRON, 2019)

Infine, il machine learning può essere utile per aiutare l'uomo ad imparare (Fig. 1.2.5): gli algoritmi di ML possono essere analizzati per vedere cosa hanno imparato e in che modo risolvono il problema (anche se non sempre questa procedura è così immediata e semplice come sembra); oppure, si può utilizzare il ML per processare una grossa quantità di dati al fine di scovare pattern non subito visibili (data mining).

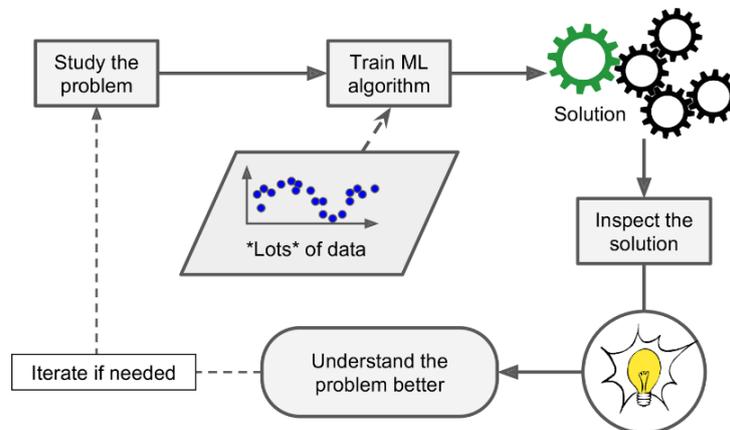


FIGURA 1.2.5: ML IN AIUTO ALL'UOMO (GÉRON, 2019)

Quindi, riassumendo, l'approccio di ML risulta conveniente per:

- Problemi molto complessi, la cui risoluzione con approcci tradizionali richiederebbe troppo tempo e programmi troppo lunghi e articolati.

- Problemi eccessivamente complessi, di cui si conosce poco e la cui analisi risulta infruttuosa; in questo caso, saranno i dati relativi al problema a “parlare da sé”.
- Problemi mutevoli, ovvero problemi che vanno incontro a cambiamenti nel tempo e che il ML riesce ugualmente a gestire grazie alla sua capacità di adattarsi automaticamente.
- Aiutare l’uomo, fornendo intuizioni sulla risoluzione di problemi o per scovare pattern difficilmente visibili in un grosso insieme di dati.

Gli ambiti in cui al giorno d’oggi è utilizzato il ML sono molteplici, tra cui: comprensione del linguaggio parlato, riconoscimento ottico dei caratteri, face detection, filtro per le e-mail spam, diagnosi medica e relativo trattamento, predizione del meteo, etc... Per fare ciò esistono diversi tipi di sistemi di ML, che possono essere suddivisi secondo tre criteri principali:

- In base al fatto che vengano o meno allenati con la supervisione umana (supervised, unsupervised, semisupervised e reinforcement learning).
- In base al fatto che possano o meno imparare man mano dai dati (online vs. batch learning).
- In base al fatto che lavorino comparando i nuovi dati con i dati conosciuti o che, scovando un pattern nell’insieme training data, costruiscano un modello predittivo (instance-based vs. model-based learning).

Alcuni di questi sistemi verranno meglio approfonditi nei paragrafi successivi, spiegandoli nel dettaglio e riportando alcuni esempi di algoritmi più usati nel campo biomedico.

Per quanto riguarda invece il deep learning (DL), esso è un sottoinsieme del machine learning, che usa complessi algoritmi, ispirati al cervello umano e al suo funzionamento con lo scopo di imparare da un grosso insieme di dati etichettati. Le tecniche di DL si basano spesso su reti neurali profonde, ovvero reti neurali artificiali con più di un hidden layer (vedere paragrafo 1.3.5), tra

cui reti neurali profonde feedforward, convoluzione di reti neurali profonde, deep belief network, reti neurali ricorsive, che sono state applicate nella visione artificiale, nel riconoscimento automatico del discorso, nell'elaborazione del linguaggio naturale, nel riconoscimento audio e nella bioinformatica.

La principale differenza tra ML e DL risiede nel fatto che mentre per il ML è necessario l'utilizzo di dati strutturati, le cui caratteristiche sono spesso estratte manualmente dall'uomo, nel DL l'utilizzo di dati strutturati non è più necessario, dal momento che il sistema stesso identifica nei dati le caratteristiche distintive adeguate, senza la necessità di una categorizzazione dall'esterno. Per permettere ciò, tuttavia, il sistema di DL ha spesso bisogno di una grossa quantità di dati, molti di più di quanti richiesti dal ML.

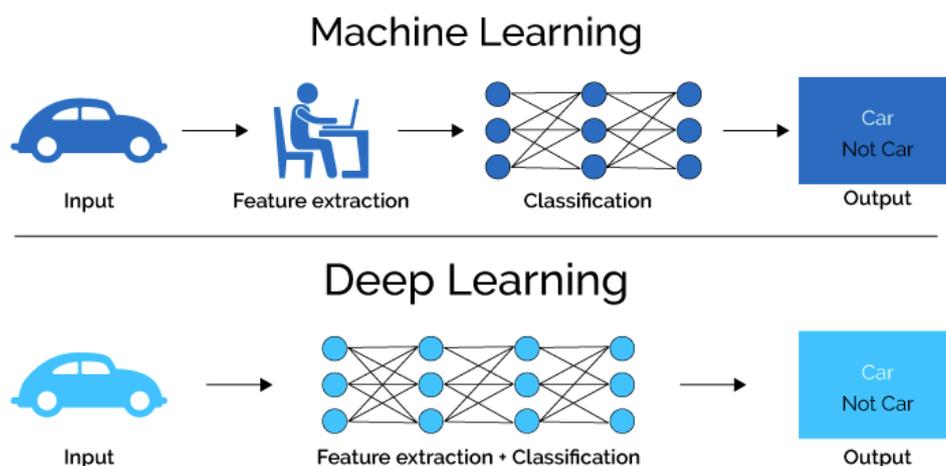


FIGURA 1.2.6: DIFFERENZA TRA ML E DL (LAVECCHIA, 2019)

1.3 Supervised Learning

Il supervised learning (apprendimento supervisionato) è un tipo di machine learning allenato con la supervisione umana. Per “supervisione” si intende la presenza, nel dataset, di dati etichettati, ovvero di dati che verranno utilizzati come input al sistema e di cui

si conosce già l'output. Basandosi sulla relazione nota input-output di questi dati etichettati, il sistema sarà in grado, durante la sua fase di allenamento, di modificare i suoi parametri interni al fine di avvicinarsi il più possibile alla relazione input-output nota. Così facendo, nel momento in cui si presenterà al sistema un nuovo dato, che non sarà quindi etichettato (ovvero di cui non si conosce a priori l'output), il sistema, sfruttando la relazione input-output ottimizzata durante la fase di allenamento, ne fornirà l'output.

A seconda della tipologia dell'output fornito dal sistema di ML, ovvero a seconda del compito che deve risolvere, il supervised learning può essere distinto in due gruppi principali:

- **Problema di classificazione:** lo scopo del sistema di ML è quello di modellare una funzione che associ ad una variabile di input X una variabile discreta di output Y . Questa variabile di output Y è spesso detta “categoria” o “etichetta”, e quindi compito del modello di ML sarà quello di assegnare la variabile X ad una delle possibili categorie. Allo stesso tempo, spesso un sistema di classificazione fornisce come output la probabilità, per ogni categoria, che la variabile d'ingresso X appartenga a quella categoria. Questi valori continui di probabilità potranno poi essere convertiti in un valore discreto di categoria, semplicemente scegliendo la categoria che ha avuto la probabilità più alta. Un classico esempio di ciò è un algoritmo il cui compito sia quello di separare le e-mail spam da quelle non spam: il modello potrà semplicemente decidere, per ogni e-mail, se quest'ultima è spam o no, oppure fornire dei valori di probabilità, per esempio dicendo che quell'e-mail è al 90% non spam e al 10% spam, e quindi in questo caso si classificherà quell'e-mail come non spam. Tipici esempi di algoritmi di classificazione sono: classificatore lineare, support vector machines (SVM), decision trees, k-nearest neighbor, random forest e naive bayes.
- **Problema di regressione:** lo scopo del sistema di ML è quello di modellare una funzione che associ ad una variabile di input X una variabile continua di output Y . La variabile d'ingresso X può essere

sia discreta che continua e questo tipo di sistema è spesso usato per fare previsioni di quantità (come, per esempio, il prezzo di una casa, o di un'auto). Tipici algoritmi di regressione sono la regressione lineare, la regressione logistica e la regressione polinomiale.

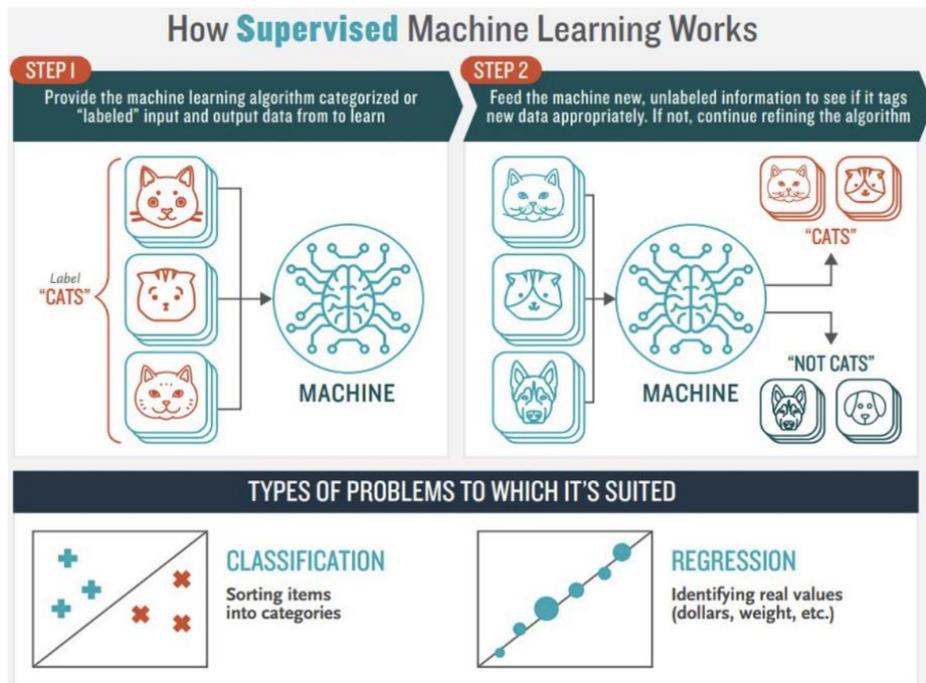


FIGURA 1.3.1: SUPERVISED LEARNING (*SUPERVISED LEARNING. IN MACHINE LEARNING, SUPERVISED... / BY JORGE LEONEL / MEDIUM, S.D.*)

1.3.1 Support Vector Machines (SVM)

Le support vector machines sono dei modelli di supervised machine learning, ideati da Vladimir Vapnik, usati sia per problemi di classificazione che di regressione, con più frequenza per i primi rispetto ai secondi. Questa tipologia di algoritmi è spesso usata per suddividere in due categorie un insieme di dati, di dimensione N , usando un iperpiano di dimensione $N-1$, che fungerà quindi da confine tra le due categorie. Per esempio, nel caso in cui si avessero dei dati a tre dimensioni, ciò che si userebbe sarebbe un iperpiano a

due dimensioni, ovvero un piano; se si avessero dei dati a due dimensioni, ciò che si userebbe sarebbe un iperpiano a una dimensione, cioè una retta. Tipicamente, l'iperpiano è costruito cercando di far sì che sia il più distante possibile dai dati noti delle due categorie, in modo tale che gli esempi appartenenti alle due diverse categorie siano chiaramente separati da uno spazio, detto margine, il più possibile ampio. Nel momento in cui si avrà un nuovo dato, semplicemente “mappandolo” nello spazio di dimensione N si osserverà in quale delle due categorie ricadrà.

Spesso, tuttavia, può succedere che i dati ad N dimensioni non siano separabili nello spazio a N dimensioni. Per ovviare a questo problema, i dati ad N dimensioni vengono mappati, usando una “funzione kernel”, in uno spazio di dimensione più grande, dove verosimilmente è possibile suddividere in due categorie i dati. A seconda dell'algoritmo SVM usato, si potranno avere diverse funzioni kernel, come per esempio quella lineare, non lineare, polinomiale o la funzione radiale di base.

Gli ambiti di utilizzo delle SVM sono molteplici, anche grazie alla versatilità che possiedono questi algoritmi per via delle diverse funzioni kernel utilizzabili; alcuni degli ambiti in cui le SVM sono più usate sono: elaborazione del linguaggio naturale, identificazione di facce in immagini, identificazione di pedoni e classificazione di testi.

1.3.2 Decision Tree e Random Forest

Il decision tree, o albero decisionale, è una tecnica di supervised machine learning usata sia per problemi di classificazione che di regressione. Quando l'albero decisionale è usato per classificare i dati all'interno di categoria, prende il nome di “classification tree”; quando è utilizzato invece per prevedere valori numerici, è detto “regression tree”. L'albero decisionale è un algoritmo

estremamente semplice da costruire, usare ed interpretare in quanto esso suddivide i dati noti in sottoinsiemi sempre più omogenei (ovvero sottoinsiemi i cui dati hanno sempre più caratteristiche uguali), fino ad arrivare al punto in cui non è più possibile procedere. L'albero si compone di due parti principali: i nodi, ovvero i punti dell'albero in cui i dati vengono suddivisi in base al criterio scritto all'interno del nodo, e le foglie, che invece sono i punti terminali dell'albero, oltre al quale non si può procedere. Il primo nodo dell'albero è anche detto "root node", ovvero è la radice dell'albero. Fornito un insieme di dati di allenamento, per costruire un albero decisionale esistono diversi metodi, come l'entropia o il metodo Gini. Infine, quando si avrà un nuovo dato, seguendo il percorso dell'albero dalla radice fino alle foglie basandosi sulle caratteristiche del nuovo dato, si arriverà a classificare quest'ultimo.

Un albero decisionale è dotato di diversi parametri, che vanno definiti prima di costruire l'albero. Alcuni di questi parametri sono:

- **Criterion:** è il parametro usato per determinare l'impurità, ovvero il valore predittivo di una caratteristica, e quindi di un nodo. Come citato precedentemente, per un "classification tree" i metodi più usati sono l'entropia e l'indice di Gini.
- **Profondità dell'albero:** indica di quanti nodi al massimo si può scendere in un albero. Più la profondità è maggiore, più si riuscirà a suddividere l'insieme dei dati di allenamento in gruppi omogenei e quindi maggiore sarà la precisione dell'albero decisionale sul set dei dati di allenamento. Tuttavia, il rischio che si corre nell'aumentare la profondità dell'albero è di andare incontro al fenomeno dell'overfitting: l'albero decisionale sarà molto preciso nel classificare i dati di allenamento, ma avrà una bassa accuratezza nei nuovi dati. In questo caso l'albero non è stato in grado di generalizzare: l'algoritmo si è adatto molto bene ai dati di allenamento, arrivando ad avere ottime prestazioni su di essi, ma non è in grado di applicare ciò che ha imparato su nuovi dati.

- Numero minimo di campioni per foglia: indica il numero minimo di campioni che si devono avere per definire una foglia dell'albero. Più è basso, più l'albero sarà preciso sui dati di allenamento; anche in questo caso si rischia di incorrere nel fenomeno dell'overfitting.

Nonostante l'albero decisionale sia un algoritmo molto semplice ed intuitivo, che richiede anche pochi dati per essere allenato, è un algoritmo poco usato; i motivi principali di ciò sono che l'albero decisionale è molto soggetto al fenomeno dell'overfitting ed è un algoritmo poco robusto: piccole variazioni nel set dei dati di allenamento possono far variare molto la struttura dell'albero, che quindi non sarà in grado di generalizzare e risulterà poco accurato quando si tratterà di classificare nuovi dati.

Per ovviare questi problemi, cercando però di mantenere gli aspetti positivi, è molto usato l'algoritmo Random Forest, o foresta casuale, che non è altro che un insieme di alberi decisionali. Per costruire questa "foresta" di alberi decisionali, si seguono pochi semplici passi:

- Si crea il "bootstrapped dataset": a partire dal set di dati di allenamento iniziale, si crea un altro dataset, di dimensione uguale al primo, scegliendo casualmente dal primo dataset i dati. L'aspetto importante è che in questa fase è consentito scegliere più volte uno stesso dato. Così facendo, se da un lato si potrà ripescare lo stesso dato più volte, dall'altro lato alcuni dati non verranno selezionati per la fase di allenamento (ovvero per la fase di costruzione della foresta). L'insieme di quest'ultimi dati, che prende il nome di "out-of-bag-dataset", verrà usata per verificare le performance della foresta casuale, mentre la porzione degli "out-of-bag-samples" che sarà classificata incorrettamente viene detta "out-of-bag-error".
- Successivamente, a partire da questo bootstrapped dataset, si costruisce un albero decisionale: a differenza dell'algoritmo "decision tree", in questo caso si utilizza solo un sottoinsieme delle caratteristiche del bootstrapped dataset. La dimensione di questo sottoinsieme può dipendere da algoritmo ad algoritmo, ma spesso

si sceglie il numero intero più vicino alla radice quadrata del numero totale di caratteristiche.

- Infine, si ripetono questi primi due passi, costruendo ogni volta un nuovo albero decisionale.

Così facendo, si costruisce un gran numero di alberi decisionali, indipendenti tra di loro; ciò rende più accurata la foresta casuale rispetto ad un singolo albero decisionale. In un problema di classificazione, nel momento in cui si avrà un nuovo dato, quest'ultimo verrà fatto processare da tutti gli alberi decisionali e la categoria a cui apparterrà il dato sarà quella scelta dalla maggior parte degli alberi. Nel caso invece di un problema di regressione, il dato verrà sempre fatto processare da tutti gli alberi, mentre come variabile di output si prenderà la media dei valori di tutti gli alberi. Questo procedimento di creare il bootstrapped dataset e di fare processare un nuovo dato da tutti gli alberi (detto "aggregate") prende il nome di "bagging".

Un algoritmo molto simile al random forest è l'"extra trees". Questo algoritmo opera in maniera quasi identica al random forest, con prestazioni leggermente inferiori ma con una velocità maggiore.

1.3.3 Naive Bayes

L'algoritmo naive bayes è una tecnica di supervised machine learning utilizzata principalmente per problemi di classificazione. Come suggerisce il termine stesso, questo algoritmo si basa sul teorema di Bayes, che calcola la probabilità di un evento y condizionata da un altro evento x :

$$P(y|x) = \frac{P(x|y) \times P(y)}{P(x)} \quad (1.3.1)$$

Dove $P(y|x)$ è la probabilità condizionata che accada l'evento y , sapendo che accade l'evento x ; $P(x|y)$ è la probabilità condizionata che accada l'evento x , sapendo che accade l'evento y ; $P(x)$ è la probabilità che accada l'evento x mentre $P(y)$ è la probabilità che accada l'evento y .

Per esempio, in un dataset di allenamento contenente N istanze, cioè dati, ognuno con M caratteristiche e due categorie in cui sono classificati i dati (e in cui si dovranno classificare nuove istanze), si avrà:

- $P(y)$: numero di dati appartenenti alla categoria y sul totale N delle istanze;
- $P(x|y)$: probabilità che un determinato valore (x) di una caratteristica sia presente nella categoria y ;
- $P(x)$: probabilità che il valore x di una caratteristica sia presente nelle N istanze;
- $P(y|x)$: probabilità che una istanza, avente in una caratteristica il valore x , appartenga alla categoria y .

Nel momento in cui, invece che considerare solo il valore di una caratteristica, si considerino i valori di tutto l'insieme delle caratteristiche, la formula (1.3.1) diventa:

$$P(y|x_1, \dots, x_N) = \frac{P(x_1, \dots, x_N|y) \times P(y)}{P(x_1, \dots, x_N)} \quad (1.3.2)$$

Facendo l'ipotesi "naive" (cioè ingenua) che i valori delle caratteristiche siano tra di loro indipendenti, ovvero che per la generica i -esima caratteristica valga:

$$P(x_i|y, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_N) = P(x_i|y) \quad (1.3.3)$$

¹ Ciò, in pratica, significa che la conoscenza dei valori delle $N-1$ caratteristiche non fornisce alcun "indizio" sul valore della rimanente caratteristica.

Allora la formula (1.3.2) può essere riscritta come:

$$P(y|x_1, \dots, x_N) = \frac{P(y) \times \prod_{i=1}^N P(x_i|y)}{P(x_1, \dots, x_N)} \quad (1.3.4)$$

Nel momento in cui si avrà un nuovo dato avente come caratteristiche i valori (x_1, \dots, x_N) , usando la (1.3.4) si potrà calcolare la probabilità che quell'istanza appartenga alla categoria y .

Sebbene l'algoritmo Naive Bayes sia semplice e risolva efficacemente alcuni problemi di classificazione, ha diversi problemi: prima di tutto, è un algoritmo "naive", ovvero ingenuo, perché ipotizza che tutte le caratteristiche di una istanza siano tra di loro indipendenti, che non sempre è vera come ipotesi; inoltre, richiede la conoscenza di tutti i dati del problema per poter calcolare le probabilità condizionate e semplici, e ciò non è sempre fattibile.

1.3.4 Regressione Lineare e Logistica

La regressione lineare è una tecnica di supervised machine learning usata principalmente per problemi di regressione. Questa tecnica permette di trovare la miglior relazione tra una variabile di input X e una variabile di output Y , nel caso della regressione lineare semplice, o più variabili di uscita Y , nel caso della regressione lineare multipla. È una metodologia usata soprattutto per fare previsioni di quantità, come può essere per esempio il prezzo di una casa sapendo la sua superficie. La relazione tra la variabile d'ingresso X e quella di uscita Y è spesso ottimizzata usando il metodo dei minimi quadrati, il quale consiste nel minimizzare la somma degli errori quadratici tra gli output della funzione da ottimizzare e le variabili di uscita Y note.

La regressione logistica è concettualmente simile alla regressione lineare, con la differenza che mentre la regressione lineare è utilizzata per valori continui della variabile d'uscita Y , la regressione logistica è invece usata quando la variabile dipendente Y è discreta e può assumere solo due valori, come “Sì”/”No” oppure “Vero”/”Falso”. Sebbene entrambi i metodi di regressione (lineare e logistica) siano utilizzati per trovare la miglior relazione tra le variabili X e Y , la regressione logistica è spesso usata per risolvere problemi di classificazione binaria. Il motivo per cui questo tipo di regressione è detta logistica è perché, al fine di risolvere problemi di classificazione binaria, utilizza la funzione logistica, la cui equazione è:

$$f(x) = \frac{L}{1 + e^{-k(x-x_0)}} \quad (1.3.5)$$

Dove:

- L è il valore massimo della curva;
- x_0 è il valore medio lungo l'asse x ;
- K definisce la pendenza della curva per $x = x_0$

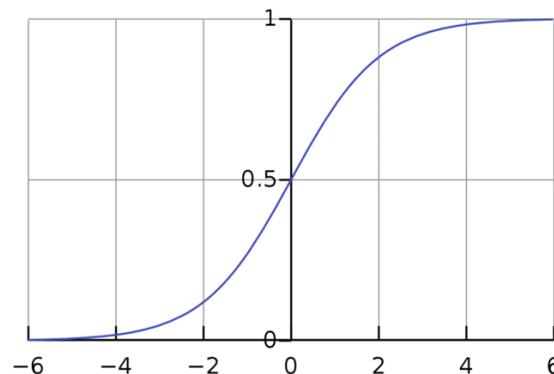


FIGURA 1.3.3: FUNZIONE LOGISTICA CON $L=1$, $K=1$ E $x_0=0$ (GUIDA RAPIDA ALLE FUNZIONI DI ATTIVAZIONE NEL DEEP LEARNING – NETAI, S.D.)

Definendo, per esempio, il valore “0” come “Falso” e il valore “1” come “Vero”, il valore fornito dalla curva logistica può essere visto come la probabilità che quella nuova istanza appartenga alla categoria “Vero”: più è alto il valore fornito dalla curva logistica,

più la probabilità è vicina ad “1”, cioè alla categoria “Vero”. In questo modo si riesce a sfruttare una curva continua per risolvere un problema di classificazione binaria.

1.3.5 Reti Neurali Artificiali

Le reti neurali artificiali sono un insieme di algoritmi di supervised learning, appartenenti sia alla famiglia del ML che del DL. Come suggerisce il termine stesso, questo algoritmo prende spunto dalla struttura del cervello, realizzando quella che è una rete composta di neuroni connessi tra di loro.

Uno tra i primi modelli di rete neurale artificiale fu il perceptrone, ideato da Frank Rosenblatt. La struttura del perceptrone era molto semplice: esso era composto da un singolo neurone che, ricevendo in input un vettore x di ingressi, forniva in uscita un valore scalare reale $f(x)$ così calcolato:

$$f(x) = g(\langle w, x \rangle + b) \quad (1.3.6)$$

Dove $\langle w, x \rangle$ è il prodotto scalare tra il vettore degli ingressi e w , il vettore dei pesi, b è il “bias” del neurone e $g(y)$ è la funzione di attivazione del neurone. Il ruolo del vettore w è quello di pesare i valori degli ingressi, dando ad ognuno una specifica rilevanza; il bias b , che è un parametro interno al neurone, ha lo scopo di settare la soglia di attivazione del neurone: spesso è un valore negativo, e più negativo esso risulta più la somma pesata degli ingressi dovrà essere grande per “attivare” il neurone. Infine, la funzione di attivazione $g(y)$ serve per scalare il valore dato dalla somma tra il bias e la somma pesata degli ingressi. Alcune tra le funzioni più usate inizialmente erano la funzione segno, molto usata per i classificatori binari e che restituisce solamente i valori -1 e +1 a

seconda che il valore assunto da y sia rispettivamente negativo o positivo, e la funzione a gradino unitario.

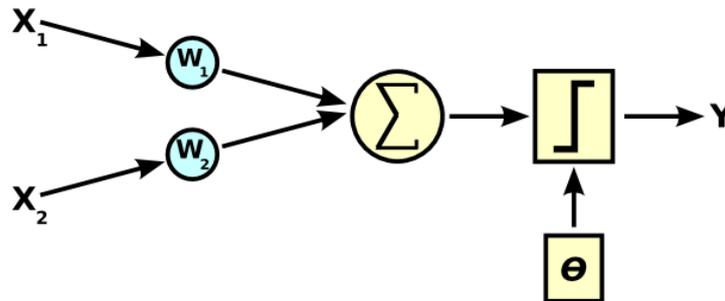


FIGURA 1.3.3: STRUTTURA BASE DEL PERCETTRONE
(USER: YEAROFTHE DRAGON, S.D.)

Il perceptrone può essere visto come l'antecedente delle reti neurali feed-forward, ovvero di quelle reti le cui connessioni tra neuroni non formano cicli e il flusso di informazioni procede direttamente dall'ingresso all'uscita senza mai tornare indietro. Il perceptrone utilizzava come regola d'apprendimento l'algoritmo di retropropagazione dell'errore; questo algoritmo, che verrà meglio discusso più avanti, permetteva al perceptrone di modificare i valori del vettore dei pesi w e del bias b , ottimizzando la funzione $f(x)$. Così facendo, il perceptrone poteva essere usato per prendere delle decisioni sulla base dei valori dell'ingresso, o per implementare operatori logici quali AND, OR e NOT, semplicemente variando l'importanza degli input e il suo valore di soglia. Un problema del perceptrone è che, da solo, non è in grado di imparare la risoluzione di problemi non linearmente separabili. Questo problema fu chiaramente esposto nel 1969 nel libro "Perceptrons" (Minsky & Papert, 2017) di Minsky e Papert, che mostrava come un perceptrone non fosse in grado di implementare la funzione logica XOR.

Tuttavia, questo problema è in realtà solo un problema parziale: infatti, una rete di perceptroni, detta "multilayer perceptron"

(MLP), è in grado di risolvere questa classe di problemi non linearmente separabili. Basti pensare che, per esempio, in un circuito logico la funzione XOR può essere implementata da un insieme di gate logici AND, OR e NOT; se un perceptrone può implementare ciascuna di queste funzioni logiche, una MLP sarà in grado di implementare lo XOR.

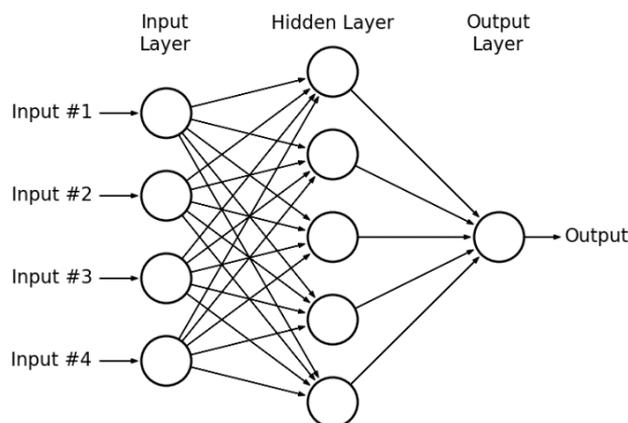


FIGURA 1.3.4: ESEMPIO DI UNA MLP (NAKHAWA, 2021)

Una MLP è tipicamente costituita da “layer” o strati di perceptroni che ricevono come input l’uscita dei perceptroni del layer precedente e che forniscono l’output ai perceptroni del layer successivo. Essendo una rete di tipo feed-forward, ogni layer riceve informazioni dal layer precedente e fornisce informazioni al layer successivo, senza che vi sia mai uno scambio di informazioni tra perceptroni all’interno di uno stesso layer o che l’informazione possa tornare indietro. In una MLP vi sono tre tipi di layer: l’”input layer”, che è il layer contenente il vettore di input alla rete; l’”hidden layer”, che è invece il layer che elabora le informazioni proveniente dall’input layer (o, nel caso in cui vi siano più hidden layer, dall’hidden layer precedente); l’”output layer”, che invece è il layer che fornisce l’output (uno o più, a seconda di quanti perceptroni sono presenti in questo layer) della rete.

Al pari di un singolo perceptrone, l’importanza della MLP risiede nel fatto che, tramite algoritmi di apprendimento automatico, essa

è in grado di modificare i valori dei pesi e dei bias al suo interno, cambiando di fatto quello che è il suo “comportamento”. La logica che sta alla base di ciò è semplice: ad un piccolo cambiamento di un parametro all’interno della MLP corrisponde un piccolo cambiamento dell’uscita; cambiando progressivamente i valori dei parametri interni della MLP grazie a particolari algoritmi, la funzione svolta dalla MLP sarà in grado di modellarsi autonomamente, rispondendo alle esigenze di un particolare problema. Per far ciò, tuttavia, bisogna che ad un piccolo cambiamento di un parametro di un perceptrone corrisponda una piccola variazione del suo output. Utilizzando funzioni di attivazione come la funzione segno o la funzione a gradino unitario ciò non è possibile: essendo funzioni che assumono solo valori discreti, vi è il rischio che, cambiando il valore di un parametro di un perceptrone, la sua uscita rimanga costante (rimanendo, per esempio, fissa al valore 0), o che muti radicalmente (passando, per esempio, dal valore 0 direttamente al valore 1). Per ovviare questo problema, al giorno d’oggi sono tipicamente usate come funzioni di attivazione le funzioni sigmoidee, ovvero funzioni aventi un andamento ad “S” di cui la più usata è la funzione logistica, e la funzione ReLU (Rectified Linear Unit), la cui equazione è:

$$f(x) = \max(0, x)$$

Spesso, nel momento in cui si utilizzano queste funzioni di attivazione diverse da quelle tipiche di un perceptrone, invece che di quest’ultimo termine si utilizza il termine più generale “neurone” (o “neurone artificiale”) e, invece che utilizzare il termine MLP, si utilizza il termine più generale “rete neurale artificiale” (ANN, artificial neural network). Il vantaggio di queste funzioni di attivazione è che ad una piccola variazione di un parametro del

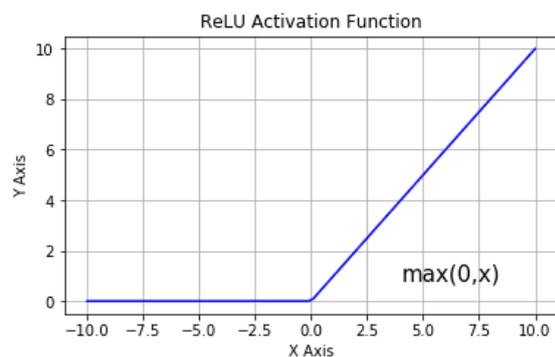


FIGURA 1.3.5: FUNZIONE RELU (NOMIDL, 2022)

neurone, corrisponde a sua volta una piccola variazione dell'uscita del neurone, dal momento che sono funzioni che possono assumere valori continui. L'importanza di queste funzioni risiede proprio nel loro andamento e non tanto nella loro particolare forma. Il motivo per cui, all'interno delle funzioni sigmoidee, è spesso utilizzata la funzione logistica è che la sua derivata è facile da calcolare; invece, il motivo per cui, soprattutto nelle reti neurali profonde (ovvero con molti hidden layer) è utilizzata la ReLU è per far fronte ad un problema detto "scomparsa del gradiente", che verrà trattato più avanti.

L'algoritmo più usato per allenare una ANN è la retropropagazione dell'errore. Questo algoritmo, facente parte degli algoritmi di apprendimento supervisionato, si basa sulla minimizzazione di una funzione errore, ovvero di una funzione che dipende dal vettore di uscita $y_{rete}(x)$ dato dalla ANN in corrispondenza di un vettore d'ingresso x alla rete e dal valore di uscita $y_{prev}(x)$ previsto per quell'ingresso x . Una funzione errore spesso usata è la funzione "errore quadratico medio" (MSE, mean square error):

$$E(w, b) = \frac{1}{2N} \sum_x \|y_{rete}(x) - y_{prev}(x)\|^2 \quad (1.3.7)$$

Dove N è il numero delle istanze di allenamento (ovvero il numero di tutti i vettori x di allenamento), w e b sono rispettivamente i pesi e i bias della rete e $\|y_{rete}(x) - y_{prev}(x)\|$ è la distanza tra il vettore $y_{rete}(x)$ e il vettore $y_{prev}(x)$, ovvero di fatto una misura dell'errore commesso dalla rete. La funzione costo $E(w, b)$ dipende dai pesi e dai bias della rete, e scopo dell'algoritmo di retropropagazione dell'errore è quello di trovare quell'insieme di valori dei pesi e dei bias tale per cui il valore della funzione costo sia il minimo possibile. In altre parole, lo scopo dell'algoritmo di retropropagazione è quello di trovare il minimo globale della funzione costo. Un metodo per far ciò è la via analitica: si può fare un sistema alle derivate parziali della funzione costo rispetto ad ogni peso e bias e grazie a ciò trovare il minimo. Tuttavia, spesso non è così semplice risolvere questo sistema e quindi si rende

necessaria un'altra strada, che è quella della “discesa del gradiente”. In questo caso, si calcolano sempre le derivate parziali della funzione costo, ottenendo il gradiente della funzione, per poi seguire altri progressivi passi:

- Si inizializzano casualmente i pesi e i bias, ottenendo un valore del vettore gradiente.
- Dal momento che il gradiente è un vettore che punta nella direzione e nel verso in cui la funzione cresce più velocemente, e dal momento che ciò che si cerca di fare è invece diminuire il valore della funzione, ci si sposterà nel verso dell'anti-gradiente, ovvero del gradiente cambiato di segno. Si può dimostrare matematicamente che spostarsi lungo questo verso garantisce al valore della funzione di diminuire. Spostarsi nel verso dell'anti-gradiente, matematicamente, significa aggiornare i valori dei pesi e dei bias così facendo: $v \rightarrow v' = v - \eta \nabla E$, dove v è il vettore iniziale dei pesi e bias, v' è il vettore aggiornato dei pesi e bias, $-\nabla E$ è l'anti-gradiente e η è il “learning rate”, ovvero un parametro (o, più correttamente, un iperparametro) che stabilisce di quanto muoversi nel verso dell'anti-gradiente. La scelta e l'ottimizzazione del learning rate è un passaggio fondamentale: un learning rate troppo piccolo significherebbe muoversi di passi troppo piccoli ad ogni iterazione, rendendo la discesa del gradiente troppo dispendiosa in termini di tempo; un learning rate troppo alto rischierebbe di oltrepassare sistematicamente il minimo della funzione, senza mai riuscire ad avvicinarsi abbastanza.
- Si ripete il secondo passaggio, calcolando il gradiente con il vettore aggiornato v' dei pesi e dei bias e spostandosi nuovamente nel verso del nuovo anti-gradiente.
- Il procedimento termina quando $\nabla E \approx 0$, idealmente quando $\nabla E = 0$.

Questo algoritmo, sebbene molto usato, non è esente da problemi: pur trovando $\nabla E = 0$, non è detto che ad esso corrisponda un minimo globale, bensì può essere anche un minimo relativo; nel caso in cui questo minimo relativo non soddisfi le esigenze della

ANN, un metodo per cercare di ovviare questo inconveniente è di ripetere la discesa del gradiente, inizializzando ancora una volta casualmente i pesi ed i bias. Un altro problema si presenta nel caso di ANN con molti layer: quando si calcola la derivata parziale della funzione costo rispetto un peso/bias di uno dei primi layer seguendo la regola della catena per le derivate, si ottiene una derivata parziale che è prodotto di numerose derivate parziali, che spesso assumono valori molto prossimi alla zero. Ciò che si rischia è che quindi la derivata parziale della funzione costo rispetto un peso/bias di uno dei primi layer sia estremamente vicina allo zero, e che quindi il valore di questo peso/bias venga aggiornato quasi per niente. Questo problema è noto come scomparsa del gradiente e un metodo per cercarlo, almeno in parte, è quello di scegliere funzioni di attivazione come la ReLU, che per suoi valori di input maggiori allo zero presenta una derivata costante. Esistono inoltre versioni simili alla discesa del gradiente, come per esempio la “discesa stocastica del gradiente”, che ne è una versione approssimata ma più veloce, che però non verranno trattate in questa tesi.

Nel campo del machine learning e deep learning esistono diverse tipologie di reti neurali artificiali, con nuovi modelli che vengono continuamente sviluppati e applicati a svariati domini di applicazione. Tra le macrocategorie di reti neurali più usate, si hanno: le reti neurali convoluzionali (CNN), che sono un particolare tipo di reti neurali feed-forward la cui struttura è ispirata dalla corteccia visiva animale e che utilizzano uno o più particolari layer, detti “convolutional layer”, dotati di filtri, detti “kernel”, per la detezione di precise strutture all’interno tipicamente di immagini; le reti neurali ricorrenti (RNN), che a differenza delle reti neurali feed-forward, presentano al loro interno connessioni tra neuroni che formano dei loop che permettono a dei layer della rete di fungere da memoria di stato, in modo che futuri ingressi della rete vengano processati anche in base alla sequenza degli ingressi precedenti.

1.4 Unsupervised e Reinforcement Learning

L'unsupervised learning (apprendimento non supervisionato) è una tecnica di machine learning che, a differenza dell'apprendimento supervisionato, durante la fase di allenamento utilizza dati non etichettati: sarà compito dell'algoritmo stesso cercare di imparare, senza alcun insegnante. È una tecnica tipicamente usata per analizzare dati e trovare al loro interno dei pattern, raggruppando assieme dati aventi caratteristiche simili. Per questo motivo, i suoi tipici domini di applicazione sono l'analisi dei dati, il customer segmentation nel campo del marketing, il riconoscimento di immagini, la riduzione della dimensione dei dati, etc... Così come per l'apprendimento supervisionato, anche per quello non supervisionato esistono diversi tipi di algoritmi, specifici per determinate applicazioni; alcuni esempi ne sono il k-Means, Hierarchical Cluster Analysis, Principal Component Analysis e Apriori.

Il reinforcement learning (apprendimento per rinforzo) è una tecnica di apprendimento automatico che mira alla creazione di agenti autonomi in grado di prendere decisioni all'interno di un ambiente dinamico, ovvero che muta nel tempo. A differenza degli altri tipi di apprendimento automatico, il reinforcement learning si occupa di problemi in cui vanno intraprese azioni sequenziali, in cui l'azione da compiere dipende dallo stato attuale del sistema e ne determina quello futuro. L'apprendimento per rinforzo è così detto perché si basa sul concetto di ricompensa come metodo di apprendimento: un'azione giusta frutterà una "ricompensa" all'agente, andando a rinforzare quel tipo di azione; viceversa, un'azione sbagliata comporterà una "penalità" all'agente. Scopo dell'agente sarà quello di massimizzare la quantità di premi ricevuti, senza essere esplicitamente programmato per tale

operazione e senza l'intervento dell'uomo. Al giorno d'oggi, il reinforcement learning trova applicazioni nel campo della robotica, della pianificazione, della guida autonoma e nei giochi/videogiochi.

1.5 Analisi delle Prestazioni

Quando si addestra un algoritmo di ML, molto importante è la suddivisione dei dati in tre grandi gruppi:

- **Training set:** insieme dei dati usato per allenare il modello. Il modello impara la relazione tra l'input fornitogli e l'output atteso proprio grazie a questo set di dati con la speranza che riesca a generalizzare tale relazione e che sia quindi in grado di fare delle buone previsioni su nuovi dati mai visti.
- **Validation set:** insieme dei dati, usati sempre durante la fase di allenamento, che serve per aiutare ad aggiustare il valore degli iper-parametri del modello. Esempi di iper-parametri sono il numero di layer di una rete neurale o la funzione costo, o la funzione kernel per quanto riguarda i modelli SVM.
- **Test set:** insieme dei dati usati per testare le performance del modello precedentemente allenato.

Nel caso in cui si abbia un grosso insieme di dati si possono creare i tre sottoinsiemi disgiunti di training, validation e test set mentre nel caso in cui si abbia un insieme di dati più piccolo, o comunque per avere una scelta più robusta degli iper-parametri, si può procedere con due tecniche dette “hold-out” e “k-fold cross validation”. In questo caso, l'insieme dei dati viene inizialmente suddiviso in due set, uno di allenamento e uno di test (fase di hold-out). Successivamente, il set di allenamento viene a sua volta suddiviso in k sottoinsiemi (fase di cross-validation): per ogni combinazione di iper-parametri che si vuole testare, si applicherà k

volte la procedura di allenamento, scegliendo ogni volta $k-1$ set come training set e il set rimanente come validation set; ognuno dei k set dovrà alla fine essere stato usato una volta come validation set. In seguito, per ogni combinazione di iper-parametri se ne calcola l'accuratezza media, scegliendo come combinazione di iper-parametri finale quella che ha avuto l'accuratezza media migliore. Infine, con questa combinazione di iper-parametri si riallrena il modello su tutto l'insieme di allenamento (formato dai k sottoinsiemi) e se ne verificano le prestazioni sul test set. Un caso estremo del “ k -fold cross validation” è il “leave-one-out”: in questo caso si ha $k=1$ e spesso viene usato, dato anche il costo computazionale che comporta, quando l'insieme dei dati di partenza è molto piccolo.

Ogni qual volta che si allena e successivamente si testa un modello di ML occorre riportare i risultati delle sue prestazioni. Tipicamente, le performance di un modello vengono riportate usando i seguenti parametri (o alcuni di essi): sensibilità, specificità, valore predittivo positivo/negativo, accuratezza, F1-score, curva ROC e area sotto la curva (AUC).

Prima di parlare di questi parametri, è importante introdurre la “matrice di confusione”, che è uno strumento per analizzare gli errori compiuti da un modello di machine learning. Un esempio di questa matrice è dato dalla (Fig. 1.5.1).

		Actual Condition		
		Actual Positive	Actual Negative	
Output of Classifier	Total Samples			
	Classify Positive	TP	FP	PPV (Precision)
Classify Negative	FN	TN		
		TPR (Recall)	TNR (Specificity)	ACC
				F-measure
				MCC

FIGURA 1.5.1: MATRICE DI CONFUSIONE (*MACHINE LEARNING-CONFUSION MATRIX* / I2TUTORIALS, S.D.)

In questa matrice, gli “actual condition” sono i veri valori di output delle istanze, che in questo esempio si suddividono in positivi e negativi, mentre gli “output of classifier” sono i valori di output restituiti dal modello di ML. TP è il numero dei “true positive” (veri positivi), ovvero il numero totale delle istanze che il modello ha classificato come positive e che, nella realtà, sono positive; TN è il numero dei “true negative” (veri negativi), ovvero il numero totale delle istanze che il modello ha classificato come negative e che, nella realtà, sono negative; FP è il numero dei “false positive” (falsi positivi), ovvero il numero totale delle istanze che il modello ha classificato come positive ma che, nella realtà, sono negative; FN è il numero dei “false negativi” (falsi negativi), ovvero il numero totale delle istanze che il modello ha classificato come negative ma che, nella realtà, sono positive. Questa matrice può essere estesa anche nel caso in cui si abbiano più classi, semplicemente considerando come valore correttamente previsto il valore in cui classe reale e prevista coincidono, mentre come valore erroneamente previsto il valore in cui classe reale e prevista non coincidono. A partire da questi dati, si possono calcolare i seguenti parametri:

- Sensibilità: è data da $\frac{TP}{TP+FN}$, ovvero è la percentuale delle previsioni positive corrette (TP) sul totale delle istanze positive (TP+FN); in altre parole, è la probabilità che una istanza positiva venga correttamente classificata come positiva. Più è alta la sensibilità, minori sono i FN.
- Specificità: è data da $\frac{TN}{TN+FP}$, ovvero è la percentuale delle previsioni negative corrette (TN) sul totale delle istanze negative (TN+FP); in altre parole, è la probabilità che una istanza negativa venga correttamente classificata come negativa. Più è alta la specificità, minori sono i FP.
- Valore predittivo positivo (PPV) e negativo (NPV): il primo è dato da $\frac{TP}{TP+FP}$, ovvero è la percentuale delle previsioni positive corrette (TP) sul totale delle previsioni positive (TP+FP); in altre parole, è la probabilità che una istanza classificata positiva dal modello di ML sia effettivamente positiva. Il secondo è dato da $\frac{TN}{TN+FN}$, ovvero è la percentuale delle previsioni negative corrette (TN) sul totale delle previsioni negative (TN+FN); in altre parole, è la probabilità che una istanza classificata negativa dal modello di ML sia effettivamente negativa.
- Accuratezza (ACC): è data da $\frac{TP+TN}{TP+FP+TN+FN}$, ovvero è la percentuale delle previsioni corrette effettuate dal modello (TP+TN) sul totale delle previsioni fatte (TP+TN+FP+FN). L'utilizzo dell'accuratezza da solo è in generale sconsigliato quando il numero di istanze per ogni categoria è sbilanciato, perché rischia di fornire un risultato falsificato.
- F1-score: è dato da $\frac{2 \times \text{specificità} \times \text{sensibilità}}{\text{specificità} + \text{sensibilità}}$, ovvero dalla media armonica dei valori di sensibilità e di specificità. Al pari di tutti i parametri sopra esposti, può variare da 0 (caso peggiore) a 1 (caso migliore).

Un altro parametro importante che descrive il modello di ML è la curva ROC (receiver operating characteristic). La curva ROC è costruita sul piano ROC, che è un piano avente come asse delle

ordinate la sensibilità e come asse delle ascisse il “tasso dei falsi positivi”; quest’ultimo è dato da $1 - \text{specificità} = \frac{FP}{TN+FP}$, ovvero è la percentuale delle istanze effettivamente negative che però il modello classifica come positive (o, in altre parole, è la probabilità che una istanza classificata positiva dal modello di ML sia in realtà negativa). Dal momento che sia la sensibilità che il tasso dei falsi positivi possono variare da 0 ad 1 (essendo, a tutti gli effetti, delle probabilità), il piano ROC sarà un quadrato come mostrato in (Fig. 1.5.2).

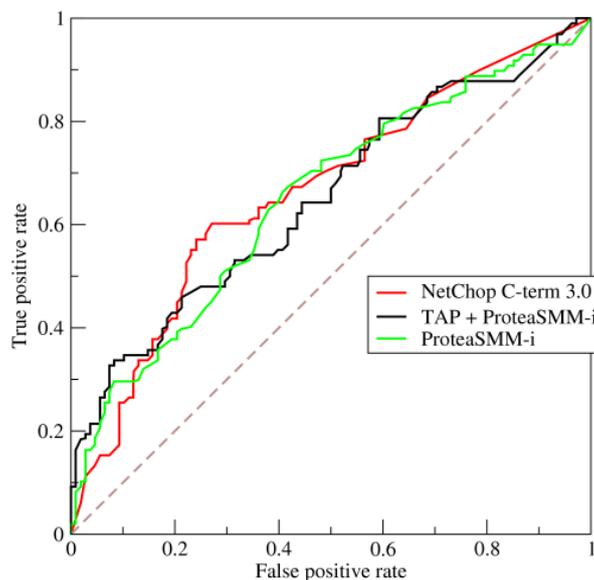


FIGURA 1.5.2: ESEMPIO DI CURVE E PIANO ROC (WIKIPEDIA, 2006)

All’interno del piano ROC vi sono tre punti principali: il punto (0;0), a cui corrisponde un classificatore che classifica tutte le istanze come negative (sensibilità minima e specificità massima); il punto (0;1), a cui corrisponde un classificatore perfetto, con sensibilità e specificità massime; il punto (1;1), che corrisponde ad un classificatore che classifica tutte le istanze come positive (sensibilità massima ma specificità minima). Ad ogni classificatore corrisponderà un punto nel piano ROC e, nel momento in cui si confronteranno tra di loro due classificatori, il “migliore” dei due sarà quello a cui corrisponde un punto più in alto e a sinistra.

Inoltre, ad un classificatore casuale, statisticamente parlando, corrisponderà un punto sulla diagonale del piano ROC che dipende dalla frequenza con la quale predice una risposta positiva: per esempio, un classificatore casuale che classifica casualmente l'80% delle istanze come positive, statisticamente classificherà l'80% delle istanze positive come positive (sensibilità = 0,8) e l'80% delle istanze negative come positive (tasso dei falsi positivi = 0,8). Un classificatore che si posizioni al di sotto della diagonale del piano ROC avrà quindi prestazioni inferiori di un classificatore casuale.

Molti classificatori, tuttavia, invece che fornire direttamente ad una istanza la sua classe di appartenenza prevista, forniscono per ogni classe una percentuale, che rappresenta la probabilità con cui quella istanza appartiene ad ogni classe. Per trasformare questi valori percentuali in una classe, si deve definire una soglia: per esempio, per un classificatore binario, si può pensare di definire una soglia percentuale dove se l'uscita del classificatore è sopra o uguale alla soglia, il classificatore classifica l'istanza positiva, altrimenti come negativa. Per ogni valore della soglia si può tracciare un punto sul piano ROC e, variando la soglia da 0% ad oltre 100%, si traccia la curva ROC di quel classificatore.

Un classificatore ideale, che cioè associa ad ogni istanza negativa una probabilità di essere positiva pari allo 0% e per ogni istanza positiva una probabilità di essere positiva pari al 100%, avrà una curva così fatta:

- Ad una soglia pari allo 0% (ovvero una soglia per cui per qualsiasi percentuale fornita dal classificatore, l'istanza viene classificata comunque positiva) corrisponderà il punto (1;1): questo punto sarà in comune con qualsiasi classificatore (standard, ideale, casuale, etc.), dal momento che con una soglia pari allo 0% tutte le istanze, indipendente dall'uscita del classificatore, sono classificate come positive. Ciò comporta una sensibilità massima, e una specificità minima (quindi un tasso di falsi positivi massimo);

- All'aumentare della soglia, per qualsiasi suo valore superiore allo 0% (escluso) e inferiore al 100% (incluso), al classificatore ideale corrisponderà il punto (0;1): dal momento che il classificatore ideale associa ad ogni istanza negativa una probabilità di essere positiva pari allo 0% e per ogni istanza positiva una probabilità di essere positiva pari al 100%, indipendentemente del valore che la soglia assume superiore allo 0% (escluso) e inferiore al 100% (incluso), le istanze negative verranno comunque giustamente classificate come negative e le istanze positive verranno comunque giustamente classificate come positive. Questa caratteristica è ciò che differenzierà un classificatore ideale da uno non ideale.
- Ad una soglia superiore al 100% (ovvero una soglia per cui per qualsiasi percentuale fornita dal classificatore, l'istanza viene classificata comunque negativa) corrisponderà il punto (0;0); anche questo punto sarà comune a tutti i classificatori.

Ciò che differenzia un classificatore ideale da uno reale è che quest'ultimo non fornirà alle istanze effettivamente positive una probabilità di essere positive pari al 100% (anche se, ovviamente, più alta sarà meglio è), così come non fornirà alle istanze effettivamente negative una probabilità di essere positive pari allo 0% (in questo caso, comunque, più bassa sarà meglio è). Di conseguenza, nel momento in cui la soglia aumenterà di valore, alcune delle istanze effettivamente negative continueranno comunque ad essere classificate positive e quindi il tasso dei falsi positivi non scenderà a zero, come invece succedeva in un classificatore ideale. Al tempo stesso, quando la soglia raggiungerà valori alti, vicini al 100%, alcune delle istanze effettivamente positive verranno classificate come negative, incrementando il numero dei falsi negativi e quindi diminuendo la sensibilità, cosa che invece non succedeva in un classificatore ideale.

Per poter comparare più agevolmente due classificatori a partire dalla curva ROC si usa spesso la AUC (area under the curve), ovvero l'area sottesa dalla curva ROC. Dal momento che il piano ROC è un quadrato di dimensioni 1x1, l'AUC potrà assumere

valori compresi tra 0 e 1, con un valore di 0,5 corrispondente ad un classificatore casuale; di conseguenza, un classificatore realistico dovrà avere un'AUC superiore a 0,5. Per confrontare due classificatori usando l'AUC, si avrà che:

- Se le due curve non si intersecano, il classificatore migliore sarà quello con l'AUC maggiore;
- Se le due curve si intersecano, si deve limitare il calcolo dell'area sottesa nell'intervallo di interesse per l'applicazione (per esempio ad elevata specificità (bassi valori di FP) oppure ad elevata sensibilità (alti valori di TP)).

Capitolo 2: L'Intelligenza Artificiale nelle Neuroscienze

Nel corso degli ultimi anni, l'utilizzo dell'IA, in particolar modo degli algoritmi di supervised machine learning, nell'ambito delle neuroscienze è notevolmente aumentato. I motivi di questo incremento sono principalmente due: in primo luogo, come spiegato nel primo capitolo, l'IA è un potente metodo di risoluzione di problemi, che permette spesso di raggiungere risultati migliori degli approcci tradizionali; questo motivo, combinato con il rapido incremento tecnologico a cui si sta assistendo, ha permesso all'IA di entrare in diversi domini, tra cui quello delle neuroscienze. In secondo luogo, le neuroscienze, a differenza di altri ambiti, ha il grosso vantaggio di poter essere utile all'IA stessa: nell'ambito delle reti neurali artificiali, per esempio, ispirarsi alla struttura del cervello, può essere un ottimo modo di migliorare ed innovare le reti neurali. Ciò crea un circolo vizioso in cui l'IA porta a nuove scoperte nel mondo delle neuroscienze e quest'ultime possono migliorare gli algoritmi di IA, permettendo di raggiungere ulteriori obiettivi nelle neuroscienze e così via. In questo e nei prossimi capitoli si illustrerà lo stato dell'arte relativo all'utilizzo dell'IA all'interno delle neuroscienze, citando a titolo di esempio alcune tra le più recenti ricerche in tale ambito e mostrando i vantaggi raggiungibili mediante questo approccio. Tuttavia, prima di parlare di questi vantaggi, è doveroso introdurre le neuroscienze, spiegando cosa sono, cosa studiano e in che modo e quali informazioni forniscono.

Per neuroscienze si intende quell'insieme di discipline e relativi studi scientifici che hanno per oggetto di ricerca lo studio morfofunzionale, ovvero sia strutturale che funzionale, del sistema nervoso centrale e periferico sotto l'aspetto normale e patologico, con particolare riferimento al cervello. All'interno di questo insieme convergono discipline quali fisiologia, biologia

molecolare, biologia cellulare, biochimica, anatomia, genetica, chimica, fisica, matematica, statistica e psicologia. Questo complesso di discipline rappresenta una scienza sempre più interdisciplinare, che va in senso opposto rispetto al confinamento specialistico dello studio del cervello e alla delimitazione del sapere tecnico degli anni passati.

La nascita del termine neuroscienze, per come è definito oggi, è attribuita al neurofisiologo americano Francis O. Schmitt e risale al 1962, anno nel quale Schmitt fondò il “Neuroscience Research Program”, un'organizzazione interuniversitaria e internazionale che segnò un momento chiave nella fondazione della neuroscienza come disciplina. Schmitt intuì che, per cercare di avere una piena comprensione degli aspetti morfofunzionali, sia normali che patologici, del sistema nervoso era necessario abbattere le barriere tra le diverse discipline scientifiche, unendone le risorse e gli sforzi.

Grazie soprattutto al progresso tecnologico che ha fornito strumenti sempre più complessi e sofisticati (TAC, EEG, MRI, PET...), le neuroscienze sono esplose a partire dalla seconda metà XX secolo, arrivando a studiare sempre più aspetti del sistema nervoso, anche su diverse dimensioni. Al fine di orientarsi all'interno di un campo di studio così vasto, sono stati definiti branche e ambiti applicativi all'interno delle neuroscienze tra cui:

- Neuroscienze cognitive: studiano substrati biologici, neurologici e cerebrali implicati nelle funzioni cognitive, quali attenzione, memoria, linguaggio e problem solving. Come molte altre branche, le neuroscienze cognitive hanno tratto un grosso vantaggio dall'introduzione e utilizzo di tecniche di neuroimaging (soprattutto funzionale) quali fMRI e PET, dal momento che grazie a quest'ultime si era in grado di studiare in tempo reale, senza alcuna invasività e senza alcuna interferenza, l'attivazione delle aree cerebrali durante lo svolgimento di un compito correlato alla funzione cognitiva d'interesse.

- **Neuroscienze affettive:** si occupano principalmente della ricerca di meccanismi cerebrali e neurali associati a processi affettivi, emotivi e motivazionali e alla regolazione degli stessi.
- **Neuropsicologia:** è una disciplina che nasce dall'unione della psicologia e delle neuroscienze. La neuropsicologia studia l'espressione comportamentale di una serie di deficit cerebrali. Si occupa, specificamente, di come il cervello possa influenzare cognizione e comportamenti in persone che mostrano lesioni o malattie cerebrali. Di particolare importanza in questo campo furono le ricerche svolte da Broca e da Wernicke per quanto riguarda lo studio del linguaggio.
- **Neuroscienze computazionali:** note anche come neuroscienze teoriche o neuroscienze matematiche, sono la branca delle neuroscienze che utilizza modelli matematici ed astrazioni del cervello per comprendere i principi che governano lo sviluppo, la struttura, la fisiologia e le funzioni cognitive del sistema nervoso.
- **Neuroscienze cellulari e molecolari:** studiano la biologia del sistema nervoso a più livelli: le neuroscienze cellulari studiano il sistema nervoso al livello della morfologia e del funzionamento dei neuroni. Le prime ricerche svolte in questa disciplina furono quelle di Santiago Ramón y Cajal per lo studio del neurone e di Luigi Galvani per lo studio dell'eccitabilità elettrica dei neuroni. Le neuroscienze molecolari invece si incentrano nello studio dei meccanismi con cui i neuroni si esprimono rispondendo ai segnali molecolari e di come gli assoni formino configurazioni complesse di connettività. Al fine di far ciò, questa branca di studio convoglia al suo interno discipline come biologia, biologia molecolare e genetica molecolare.
- **Neurolinguistica:** si occupa di studiare i correlati anatomico-fisiologici cerebrali del linguaggio e dei suoi deficit; nasce dall'integrazione tra linguistica, neuroscienze cognitive e psicologia dello sviluppo.
- **Neuroscienza cognitiva dello sviluppo:** è lo studio del meccanismo di sviluppo neuronale a livello cellulare; comprende lo studio di

come il cervello cresce e dei cambiamenti neurali durante lo sviluppo.

- **Neuroscienze sensoriali:** è il sottocampo delle neuroscienze che studia l'anatomia e la fisiologia dei neuroni che fanno parte dei sistemi sensoriali come la visione, l'udito e l'olfatto.

Gli ambiti di applicazione dell'IA all'interno delle neuroscienze sono molteplici ed eterogenei. Nonostante ciò, si possono individuare alcuni scopi comuni in essi: prima di tutto, molti studi richiedono un notevole quantitativo di tempo per essere svolti “a mano” e sebbene l'IA non sostituisca spesso completamente il lavoro umano, essa può essere decisamente utile per ridurre gli sforzi e il tempo richiesti. In secondo luogo, l'utilizzo dell'IA può portare ad una maggiore efficienza, riducendo la soggettività umana, i pregiudizi e gli eventuali errori commessi dall'uomo e quindi rendendo il risultato ottenuto più oggettivo, consistente e riproducibile.

Negli ultimi vent'anni, il numero di articoli riguardanti l'applicazione dell'IA nelle neuroscienze è aumentato in maniera evidente, come mostrato dal grafico in (Fig.2.1); quest'ultima riporta la proporzione di articoli, all'interno delle neuroscienze, che hanno usato il ML, dimostrando come la tendenza sia sempre più crescente.

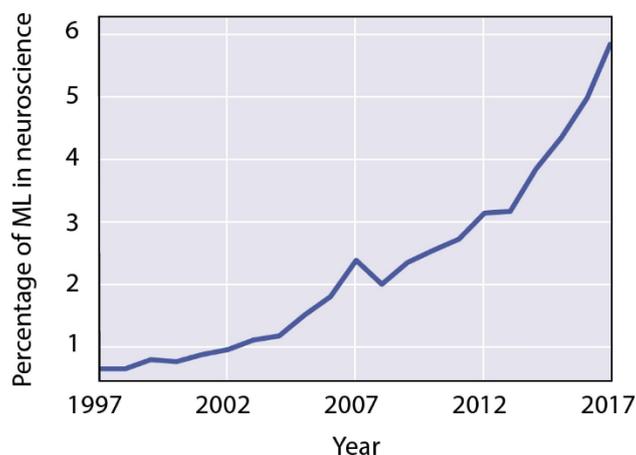


FIGURA 2.1: PROPORZIONE DEL ML NELLE NEUROSCIENZE (*THE ROLES OF SUPERVISED MACHINE LEARNING IN SYSTEMS NEUROSCIENCE / ELSEVIER ENHANCED READER, S.D.*)

I metodi di utilizzo dell'IA nelle neuroscienze possono essere suddivisi in tre macro-gruppi: utilizzo del supervised learning per la classificazione/regressione, selezione delle caratteristiche e modellistica del cervello. Nei seguenti paragrafi si approfondirà ciascuno di questi tre gruppi, con particolare attenzione al primo, riportando per ognuno di essi degli esempi.

2.1 Supervised Learning per le Neuroscienze

Un utilizzo dell'IA, e in particolare modo del ML, all'interno delle neuroscienze è quello di risolvere dei problemi in cui lo scopo è quello di stimare una variabile di uscita Y (continua o discreta) a partire da una variabile d'ingresso X misurata. Sono, più in generale, dei problemi di predizione che spesso sfruttano il supervised machine learning e dove ciò che interessa è solamente conoscere la variabile Y . Tuttavia, non è detto che la funzione che lega X ad Y sia nota, o che comunque sia semplice da calcolare. Questo approccio consente basandosi sui dati a disposizione di creare in modo automatico il modello senza dover necessariamente cercare di comprendere la relazione che esiste tra due variabili come avviene nell'approccio tradizionale. Nelle neuroscienze un vasto numero di problemi può essere visto come appartenente a questa classe di problemi, ed alcuni esempi saranno trattati nei seguenti sottoparagrafi. Sebbene l'uso dell'IA nel campo della salute mentale sia spesso riconducibile a questa tipologia di problemi, questo argomento non verrà trattato di seguito, dal momento che sarà maggiormente approfondito nel capitolo successivo.

2.1.1 Farmaci: Proprietà e Riutilizzo

Un ambito in cui l'IA può essere di grosso aiuto è quello riguardante i farmaci per il sistema nervoso centrale (CNS); infatti, nonostante il grande numero di malattie e disordini che affliggono il CNS, a cui segue di conseguenza un alto impatto sociale ed economico, lo sviluppo di farmaci per il CNS si sta rilevando lento e difficoltoso. I motivi alla base di ciò sono diversi: sicuramente, uno tra i motivi più importanti è la complessa fisiopatologia che sta alla base delle malattie e disordini del CNS, che ha come diretta conseguenza la non conoscenza dei bersagli su cui il farmaco deve agire e che, spesso, porta ad una bassa efficacia del farmaco nelle prime fasi del suo sviluppo; un altro ostacolo è la presenza della barriera emato-encefalica (BBB, blood-brain-barrier), una barriera semipermeabile altamente selettiva formata dalle cellule endoteliali sulla parete interna dei capillari sanguigni nel cervello e dagli astrociti (una forma di cellule gliali) che circondano i vasi, il cui principale compito è quello di proteggere il tessuto cerebrale, limitando il numero di sostanze che dal sangue possono passare al tessuto cerebrale. Il problema è che questa barriera, così come ostacola il passaggio di sostanze potenzialmente nocive, al contempo ostacola e spesso impedisce il passaggio anche dei farmaci, rendendoli di fatto completamente inefficaci. Se a questi problemi, infine, si aggiunge l'alto tasso di abbandono che è presente durante la sperimentazione clinica è facile comprendere come mai il tempo di sviluppo medio di un farmaco è di circa 15-19 anni, dalla sua scoperta alla sua approvazione.

Al tempo stesso, i progressi degli ultimi decenni nelle tecnologie omiche², nella sintesi chimica e nell'”high-throughput screening” (HTS), che è una tecnica che permette ai ricercatori di svolgere velocemente un alto numero di test farmacologici, chimici, etc, hanno portato ad un enorme aumento dei dati disponibili. Ci si

² Per -omica ci si riferisce a tutte quelle discipline biomolecolari terminanti in -omica, come genomica, metabolica etc...

ritrova quindi con lo stesso problema esposto nei paragrafi precedenti: i dati da analizzare sono presenti ma, data la loro quantità e la loro complessità, analizzarli “manualmente” sta diventando sempre meno efficiente. Ancora una volta l'utilizzo dell'IA può servire ad estrarre informazioni nuove ed utili a partire da questi dati, accelerando la ricerca farmaceutica.

All'interno della ricerca farmaceutica, l'IA può essere utile sotto vari aspetti; sicuramente, il più importante riguarda l'aspetto fisiopatologico di una malattia o disturbo: l'IA può servire per identificare quali molecole e quale comportamento di quest'ultime portano ad avere quella patologia, permettendo di conseguenza di avere uno o più bersagli su cui il farmaco dovrà agire. L'IA è stata inoltre impiegata per selezionare e ottimizzare i farmaci, o per scoprirne di nuovi, in modo da trovare il farmaco più efficiente possibile per un determinato bersaglio. In questo ambito vi sono due punti chiave: la relazione quantitativa struttura-attività e l'impronta digitale del farmaco. La relazione quantitativa struttura-attività, detta anche QSAR dall'inglese “quantitative structure-activity relationship”, è una relazione matematica che esprime in modo quantitativo l'attività biologica di un farmaco a partire dalle sue caratteristiche fisico-chimiche e strutturali. Questa relazione, abbinata all'utilizzo di algoritmo di ML o DL, è stata impiegata all'interno dello screening virtuale, ovvero di quella tecnica computazionale utilizzata per la ricerca, all'interno di grosse librerie di molecole, di composti la cui struttura potrebbe essere la più indicata per legarsi al bersaglio farmacologico. L'impronta digitale di una molecola è invece un metodo per codificare la struttura della molecola; il più comune tipo di impronta digitale è una serie di bits che rappresentano la presenza o l'assenza di particolari sottostrutture nella molecola. Anche l'impronta digitale, così come la QSAR, può essere abbinata ad algoritmi di ML o DL all'interno dello screening virtuale, o anche per creare la QSAR di un farmaco partendo dall'impronta di quel farmaco. Un esempio di utilizzo delle impronte digitali, della QSAR e dell'IA è dato dalla ricerca svolta da Fang et al. (Fang et al., 2015), i quali hanno usato

due impronte digitali (ECFP6 e MACCS), un particolare modello di QSAR (multitarget-QSAR, o mt-QSAR) e due algoritmi di ML (Naive Bayesian ed un modello simile al decision tree) per prevedere l'interazione tra un farmaco e più bersagli contro la Sindrome di Alzheimer (AD). Il motivo alla base di questo studio è che malattie complesse come l'Alzheimer sono spesso dovute non ad un singolo fattore, bensì alla concausa di più fattori; un nuovo approccio quindi tentato per far fronte a queste malattie è stato quello di progettare nuovi singoli composti chimici, detti "multitarget-directed ligands", in grado di interagire con più bersagli. Tuttavia, lo studio dell'interazione tra un composto ed un suo eventuale bersaglio rimane difficoltosa e richiedente molto tempo, motivo per cui approcci in silico, soprattutto basati sull'IA e sulla previsione, si stanno rendendo sempre più necessari. Nello studio svolto da Fang et al. si identificarono 25 molecole bersaglio relative all'AD con le corrispettive molecole attive e non attive, riportando ciascuno di essi con le proprie due impronte digitali e suddividendo, in parti eque, le molecole attive e non attive nel training e test set. Successivamente, per costruire il modello mt-QSAR (ovvero per prevedere quali molecole erano attive e quali no) si implementarono in totale cento modelli di ML: ogni bersaglio aveva cioè quattro modelli, ognuno dei quali si basava su una coppia algoritmo di ML e impronta digitale diversa. Inoltre, il metodo di allenamento di ogni modello consisteva in una 5-fold cross validation effettuata con il training set. Compito di ogni modello era quello di prevedere, datogli in ingresso una molecola, se quest'ultima fosse stata attiva o meno per la molecola bersaglio relativa a quell'algoritmo. Dando in ingresso ogni molecola ad ogni algoritmo, si riusciva quindi a prevedere per quali bersagli quella molecola era attiva e per quali no. Le prestazioni ottenute dai cento modelli nel test set mostrarono in generale ottimi risultati, con un AUC il cui intervallo era 0.741-1.0 e un valor medio di 0.965. I modelli bayesiani si rivelarono più performanti di quelli basati sul decision tree per quanto riguarda l'impronta ECFP6, mentre per l'impronta MACCS i risultati dei due modelli erano molto simili.

Di tutti questi modelli creati, tuttavia, risulta difficile dire quale sia il migliore: per ciascuna tipologia di algoritmo usato, cambiando le impronte digitali delle molecole (ovvero l'input all'algoritmo), le prestazioni possono cambiare; analogo discorso vale anche per le impronte digitali. Successivamente, per validare ulteriormente i modelli, essi vennero testati su sei farmaci AD già approvati, utilizzando sempre le 25 molecole bersaglio iniziali e definendo il farmaco attivo nei confronti di un bersaglio nel caso in cui almeno due dei quattro algoritmi di quel bersaglio lo avessero previsto come attivo. Così facendo, vennero trovate 44 coppie di interazione farmaco-bersaglio, 17 delle quali già validate in studi precedenti. Sebbene 17 su 44 possa sembrare un risultato non soddisfacente, va tenuto comunque conto del fatto che le rimanenti coppie non validate non sono a priori sbagliate, bensì urgeranno ulteriori studi per confermarle o smentirle. Questo studio, considerando che è uno studio pilota in questo ambito, ha mostrato degli ottimi risultati, mostrando la fattibilità di usare approcci di IA anche per la selezione di farmaci.

L'IA è stata utilizzata per altri due importanti compiti, ovvero lo studio e la predizione dell'attività dei farmaci e il riutilizzo (repurposing) dei farmaci. Lo scopo del primo compito è alquanto intuitivo: basandosi su dei descrittori molecolari, ovvero dei parametri che descrivono sinteticamente le proprietà chimico-fisiche della molecola (ovvero, in questo caso, del farmaco), e/o sulle impronte digitali della molecola, il modello di IA serve per prevedere il comportamento del farmaco, come per esempio a quali bersagli esso si possa legare, quali effetti possa avere o se riesca o meno ad oltrepassare la BBB. L'obiettivo del secondo compito è quello invece di cambiare l'utilizzo di un farmaco: conoscendo già le sue proprietà e il suo comportamento in una determinata parte del corpo umano, che è quella in cui veniva inizialmente usato il farmaco, lo scopo del modello di IA è quello di prevedere per quali altre malattie o disturbi quel farmaco potrebbe essere efficace.

Due esempi per quanto riguarda la predizione dell'attività e delle caratteristiche dei farmaci sono dati dai lavori svolti da Jiang et al. e Gao et al. Il primo gruppo si focalizzò sulla predizione della permeabilità dei farmaci alla BBB a partire dalle loro caratteristiche fisico-chimiche, come l'energia di ionizzazione e il volume di van der Waals, e dal loro fenotipo clinico, ovvero dai loro effetti collaterali e dalla loro indicazione, cioè per quali patologie viene tipicamente usato quel farmaco. Come spiega il gruppo stesso, lo studio della permeabilità alla BBB dei farmaci che agiscono nel CNS è un prerequisito molto importante per l'efficacia dei farmaci stessi, dal momento che la selettività della barriera è estremamente severa, bloccando circa il 98% dei composti esterni dall'entrare nel CNS. Ciò può, per esempio, essere un grosso problema che si trova ad affrontare il riutilizzo dei farmaci, dal momento che la BBB è un ostacolo che verosimilmente il farmaco non doveva superare nel suo primo utilizzo. Al tempo stesso, gli esperimenti in vivo della permeabilità dei farmaci alla BBB, sebbene rappresentino il gold standard, ovvero il meccanismo più affidabile e sicuro, sono troppo costosi e richiedono troppo tempo, diventando non riproducibili su larga scala. È per questo motivo che metodi in silico, come i modelli di IA, stanno ricevendo sempre più attenzioni. Inoltre, il motivo per cui nello studio svolto da Jiang et al., oltre alle caratteristiche fisico-chimiche del farmaco, furono impiegate anche le informazioni riguardanti il fenotipo clinico del farmaco fu quello di cercare di avere più informazioni possibili di varia natura del farmaco, per far fronte alla complessità della BBB ed estendere l'oggetto della ricerca non solo alle piccole molecole. L'algoritmo di ML usato in questa ricerca fu il SVM, basato su una funzione kernel polinomiale, il cui scopo era quello di prevedere la permeabilità o meno alla BBB dei farmaci. Lo studio fu diviso in tre parti principali: nella prima parte si allenò, tramite cross-validation, il SVM basandosi solo su dati relativi al fenotipo clinico dei farmaci, testando quindi il modello su un test set indipendente da quello di allenamento. I risultati di questa prima fase vennero

riportati mediante l'accuratezza, l'AUC e l'F1-score, ottenendo i valori di 68.3%, 0.692 e 0.676 rispettivamente. Partendo sempre dagli stessi dati, si provò ad allenare il modello SVM usando prima solo gli effetti collaterali e poi solo le indicazioni del farmaco: sebbene le prestazioni del modello basate solo sugli effetti collaterali fossero maggiori di quelle basate solo sulle indicazioni, le prestazioni di entrambi i casi risultarono inferiori rispetto al modello allenato usando entrambi i tipi di informazioni. Nella seconda fase della ricerca, si allenò un altro modello di SVM, sotto tre condizioni differenti: prima usando solo proprietà chimico-fisiche del farmaco, poi solo quelle relative al fenotipo clinico ed infine usando entrambe le informazioni. Mentre i risultati mostrarono come le caratteristiche relative al fenotipo clinico avessero un valore predittivo maggiore delle proprietà chimico-fisiche, le prestazioni migliori furono quelle relative al modello SVM allenato usando entrambe le caratteristiche, ottenendo un'accuratezza pari a 0.855, un AUC pari a 0.854 e un F1-score di 0.854. Vale la pena sottolineare come i risultati ottenuti unendo le informazioni chimico-fisiche e del fenotipo clinico siano migliori di molti degli studi condotti usando solo le caratteristiche chimico-fisiche. Nell'ultima fase venne applicato il modello migliore costruito nella prima fase ad un nuovo dataset di 1128 farmaci, di cui non si conosceva a priori la permeabilità della BBB. Di questo nuovo dataset, il modello SVM classificò 110 farmaci come permeabili e i restanti 1018 non permeabili. Anche in questo caso il modello ottenne buoni risultati: per esempio, seppur non si avesse una misura diretta della permeabilità dei farmaci, dei 110 farmaci classificati come permeabili alla BBB il 78,2% mostrò avere un'attività diretta nel CNS, suggerendo come verosimilmente questi farmaci fossero permeabili alla BBB. Questo studio ha quindi mostrato tre aspetti chiave: il primo è che gli studi in silico mediante utilizzo di IA sono molto promettenti per lo studio della permeabilità alla BBB dei farmaci, facendo risparmiare tempo e risorse economiche; il secondo risultato ottenuto è che integrare alle informazioni chimico-fisiche le caratteristiche relative al

fenotipo clinico conduce a delle prestazioni migliori rispetto all'utilizzo delle sole proprietà chimico-fisiche; infine, questo studio mostra la fattibilità di riutilizzare i farmaci ai disturbi o malattie inerenti al CNS.

Il secondo studio, svolto da Gao et al., si focalizzò invece sulla predizione della neurotossicità chimica dei composti usando un approccio basato sul ML. La neurotossicità è, ovviamente, un'importante caratteristica da valutare per un farmaco ed è, al tempo stesso, uno dei principali motivi della sospensione di un farmaco dal mercato. Anche in questo caso, metodi sperimentali biologici per rilevare la neurotossicità di un farmaco sono complessi e richiedono molto tempo e risorse, motivo per cui si sta sempre più puntando su metodi in silico utilizzando AI. La ricerca di Gao et al. utilizzò quindi un grosso numero di composti diversi (per rendere più generale possibile i modelli di ML) per allenare dei modelli di regressione per la predizione della neurotossicità dei composti. I modelli di ML usati furono otto, tra cui extra-trees regressor (ETR), Gaussian process regression (GPR) e random forest regressor (RF). Il dataset da 422 composti fu quindi diviso in training set e test set, in un rapporto 8:2 rispettivamente, e i modelli allenati mediante cross-validation. Come variabili d'ingresso e d'uscita ai modelli vennero usate rispettivamente dei descrittori molecolari dei composti e il pLD50, un indice della letalità di una sostanza, mentre i modelli vennero valutati mediante il coefficiente di determinazione R^2 , un coefficiente che misura la capacità predittiva di un modello e che varia tra 0 (caso peggiore) ed 1 (caso migliore). Dai risultati ottenuti basandosi sul test set, l'abilità degli otto algoritmi usati nel prevedere la neurotossicità dei composti risultò essere ovviamente diversa. Il risultato migliore fu quello ottenuto dal modello ETR, con un R^2 di 0,784. Sebbene il risultato ottenuto non sia il migliore rispetto ad altri studi simili, l'aspetto fondamentale di questa ricerca è il grosso dataset usato, che permette al modello di ML di poter essere impiegato su un più vasto dominio. In ogni caso, le prestazioni mostrate dall'algoritmo ETR lo rendono un modello quantitativo robusto ed affidabile, il

quale dimostra come gli algoritmi di AI possano essere utilizzati per un'analisi quantitativa della neurotossicità dei farmaci.

2.1.2 Studi Clinici

Un altro dominio in cui l'IA è molto utile è quello degli studi clinici. Questi ultimi, sebbene non facenti propriamente parte delle neuroscienze, possono comunque averne delle grosse ripercussioni, come per esempio l'immissione o meno sul mercato di un farmaco agente nel CNS. Un recente studio, svolto da Wouters et al., ha stimato che l'investimento medio per immettere un nuovo farmaco sul mercato sia di \$1335.9 milioni, mentre il tempo necessario varia da 10 a 15 anni mediamente. Di questo investimento e di questo tempo, circa la metà sono impiegati per la fase dello studio clinico all'interno del percorso di sviluppo di un farmaco. Tuttavia, nonostante i continui miglioramenti in campo tecnologico e i sempre più crescenti investimenti da parte delle aziende farmaceutiche e biotecnologiche nel settore ricerca e sviluppo, la tendenza attuale mostra come il numero di farmaci approvati, per miliardo di euro investiti, dimezzi ogni nove anni. Questa tendenza, in contrasto con la ben più nota legge di Moore, viene definita legge di Eroom, ed uno dei motivi alla base di essa è l'alto tasso di fallimento degli studi clinici. In particolar modo, meno di un terzo di tutti i composti chimici che arrivano alla fase II della sperimentazione clinica, dove sono realizzati gli studi clinici, riescono ad avanzare alla fase III.

I due principali motivi alla base dell'insuccesso di uno studio clinico sono la scelta di una coorte di pazienti poco adeguata allo studio clinico, dove per coorte si intende un gruppo di pazienti che condividono un insieme di caratteristiche desiderate per uno specifico studio, così come l'assenza o il non utilizzo di una tecnologia in grado di far fronte alla complessità di uno studio

clinico, ovvero di una strumentazione tecnologica che sia in grado di monitorare il paziente durante questa fase, di misurarne la non aderenza, dove per aderenza si intende la volontà e la capacità da parte del paziente a rispettare i piani dello studio clinico, e di fornire dei validi endpoint clinici, cioè misure di eventi quali presenza/assenza di sintomi o l'occorrenza di una malattia, che siano adeguati, ovvero che rappresentino una misura diretta dell'obiettivo dello studio clinico. L'IA può essere utile per risolvere questi problemi, aiutando a selezionare pazienti più adeguati per lo studio clinico e monitorandoli durante lo svolgimento di quest'ultimo.

La scelta della coorte di pazienti è un passo importante dello studio clinico, dal momento che quest'ultimo è progettato per dimostrare l'eventuale efficacia di un trattamento non in un campione casuale della popolazione, bensì in un gruppo di pazienti in cui l'effetto del farmaco, se è presente, possa essere più facilmente dimostrato. Per far ciò, ciascun studio clinico definisce in partenza criteri di inclusione ed esclusione, che descrivano le caratteristiche richieste per lo studio al fine di determinare in modo accurato l'efficacia del farmaco. Mentre alcuni di questi criteri sono facilmente identificabili all'interno della popolazione (come, per esempio, l'età), altri sono più complessi e spesso richiedono l'intervento dell'uomo, che manualmente dovrà analizzare numerosi records per identificare i pazienti qualificati. Tuttavia, a causa dei diversi modi con cui l'informazione viene registrata, degli errori nella codifica dell'informazione, di dettagli mancanti e di altri vari problemi, la selezione dei pazienti rimane uno degli aspetti più difficili, costosi e richiedenti molto tempo all'interno di uno studio clinico. Di conseguenza, vi è la necessità di trovare ed impiegare nuovi metodi, come l'IA, che possano selezionare i pazienti più adatti per ogni studio clinico, indipendentemente dai criteri usati e che richiedano un intervento umano che sia il minimo possibile.

Per verificare la fattibilità di impiegare l'IA nella selezione dei pazienti, Segura-Bedmar e Raez (Segura-Bedmar & Raez, 2019)

hanno usato un approccio basato sulle reti neurali per determinare quali criteri soddisfacesse ciascun record di ciascun paziente dato in ingresso alla rete. I ricercatori studiarono quattro diverse reti neurali, quali una CNN, una CNN profonda, una RNN e una rete ibrida formata da una CNN seguita in cascata da una RNN. Per ognuna di queste reti si valutarono altre tre versioni: ad una versione vi era l'aggiunta di uno strato completamente connesso (FFF, fully connected feedforward) immediatamente prima dell'output layer; in un'altra versione, invece che inizializzare una rete casualmente, usava una rete già pre-allenata su testi di carattere biomedico; la terza versione combinava assieme le altre due versioni. Il dataset era formato da 311 records di pazienti, etichettati manualmente da esperti per indicare quali dei 13 criteri scelti ciascun di essi soddisfacesse. Di questi 311 records, 203 vennero usati per la fase di training, implementata mediante cross-validation, e i restanti 108 per il test set. Ciascuna rete utilizzava nell'output layer una funzione di attivazione sigmoideale, che forniva per ognuno dei 13 neuroni (uno per ogni criterio) del layer la probabilità che quel records soddisfacesse quel criterio specifico. Lo scopo di questo studio, oltre a quello sopra esposto, era anche quello di valutare quale di queste reti era la migliore nello svolgere la selezione dei pazienti. Per valutare ciò, i parametri usati furono il micro-F1-score (calcolato partendo dalla somma dei TP, TN, FP e FN di tutti i criteri) e il macro-F1-score (calcolato facendo la media di ogni F1-score di ogni criterio). I risultati del test set mostrarono come il layer FFF migliorò le prestazioni di tutte le reti, tranne di quella ibrida, mentre solo la CNN profonda mostrò un miglioramento significativo quando pre-allenata. Sebbene tra tutte le reti non vi fosse una significativa differenza nelle prestazioni, tra di esse la migliore fu il modello ibrido CNN-RNN inizializzato in modo casuale e senza FFF, con un micro-F1 del valore di 78,27% ed un macro-F1 pari al 50.62%, seguito dalla RNN con FFF e inizializzata in modo casuale (micro-F1=78.43% e macro-F1=48.31%). Questi risultati mostrano come l'IA possa essere un ottimo strumento utilizzabile nella selezione dei pazienti, sia per

ridurre l'eterogeneità della popolazione, identificando quei pazienti con precise caratteristiche d'interesse, sia per scegliere pazienti che mostrino degli endpoint clinici facilmente misurabili (“prognostic enrichment”) e che siano più propensi a rispondere al trattamento (“predictive enrichment”).

Dal momento che reclutare i giusti pazienti in uno studio clinico è un grosso investimento in termini soprattutto di tempo, il ritorno di questo investimento lo si può ottenere solo tramite il completamento, da parte dei pazienti, dello studio. Ciò significa che i pazienti devono rimanere nello studio clinico, aderendo alle sue procedure e alle sue regole durante tutta la sua durata, e che tutte i dati che servono per monitorare l'impatto del farmaco testato siano collezionati in modo efficiente e affidabile. Tuttavia, solo il 15% degli studi clinici non subisce l'abbandono da parte dei pazienti e il tasso di abbandono medio negli studi clinici è del 30%. Il principale motivo alla base di questo abbandono è che, per rispettare i criteri di aderenza, i pazienti sono tenuti a tenere registri dettagliati della loro assunzione di farmaci e di una varietà di altri dati relativi come, per esempio, la risposta ai farmaci. Questo può essere un compito faticoso, che porta in media il 30% dei pazienti a diventare non aderente dopo 150 giorni dall'inizio di uno studio clinico, richiedendo successivamente un ulteriore reclutamento di pazienti che comporta costi aggiuntivi e un ritardo nello svolgimento dello studio clinico. I sensori indossabili e il monitoraggio video possono essere utilizzati per raccogliere automaticamente e continuamente i dati del paziente, agevolando così il paziente dallo svolgimento di questo compito, mentre i modelli di ML e DL possono essere utilizzati per analizzare tali dati in tempo reale per rilevare e registrare eventi di rilevanza, oppure per prevedere in modo dinamico il rischio di abbandono per uno specifico paziente, ovvero per individuare, lungo tutta la durata dello studio, quei comportamenti da parte del paziente che suggeriscono che egli stia facendo fatica ad aderire al protocollo dello studio.

Un esempio di ciò è dato dal lavoro di Koesmahargyo et al. (Koesmahargyo et al., 2020), dove fu esaminata l'accuratezza della previsione dinamica della non aderenza utilizzando due tipologie di dati: i dati "statici", ovvero che non variavano durante lo svolgimento dello studio, e i dati "dinamici", ovvero che potevano invece variare durante lo studio. Il dataset era formato da queste due tipologie di dati, provenienti da 4182 pazienti di numerosi studi clinici. Per monitorare quotidianamente i pazienti si usò la piattaforma software AiCure che, eseguita tramite cellulare, utilizza la visione artificiale per confermare, in tempo reale, l'aderenza del paziente, ovvero l'assunzione del giusto dosaggio del farmaco nelle cadenze prestabilite, ricordate a loro volta al paziente mediante una sveglia settata nel software. Il video dell'assunzione del farmaco, registrato mediante la fotocamera frontale del cellulare ed usato per confermare l'ingestione del farmaco, era successivamente controllato da degli esperti, al fine di verificare/confermare l'assunzione del farmaco. Nel momento in cui, durante lo svolgimento degli studi clinici, si sarebbe dovuto assumere il farmaco, i casi in cui si poteva andare incontro erano in totale quattro: il paziente non utilizzava il software AiCure, risultando quindi come una non-aderenza; il paziente utilizzava AiCure e il controllo umano confermava l'assunzione, risultando come una aderenza; il paziente utilizzava AiCure ma il controllo umano del video identificava una forte evidenza visiva di una non-aderenza intenzionale, marcando quindi il video con un avviso rosso; il paziente utilizzava AiCure ma, dal controllo del video, si identificavano comportamenti sospetti, senza tuttavia una chiara evidenza visiva di una non-aderenza, marcando quindi il video con un avviso arancione. Dal momento che ogni paziente poteva dover assumere il farmaco una o più volte al giorno, il parametro usato per monitorare l'aderenza giornaliera era dato da:

$$Aderenzagiornaliera = \frac{(Numero\ assunzioni\ monitorate\ da\ AiCure)}{(Numero\ assunzioni\ previste)} \quad (2.1.1)$$

Per tenere conto anche dei video marcati con avviso rosso o arancione, si introdusse l'aderenza giornaliera aggiustata, definita

come l'aderenza giornaliera dove però al “numero assunzioni monitorate da AiCure” veniva sottratto il numero di video con avviso rosso e quelli con avviso arancione.

Della tipologia dei dati statici facevano parte la lunghezza dello studio, i dati demografici, la diagnosi dei pazienti e i micro-rimborsi (ovvero degli incentivi, spesso monetari, forniti ai pazienti per adempiere appieno alle regole dello studio). Della categoria dei dati dinamici facevano invece parte il numero di interventi giornalieri dei coordinatori dello studio, dovuti per esempio per contattare un paziente che stava dimostrando una bassa aderenza, il ritardo dell'assunzione del farmaco da quando suonava l'allarme dell'AiCure, il tempo necessario per assumere il farmaco, l'aderenza giornaliera e l'aderenza giornaliera aggiustata. Tutti questi dati vennero usati come caratteristiche predittive per costruire il modello di ML, il quale forniva in uscita la previsione per la futura aderenza del paziente. L'aderenza giornaliera, l'aderenza giornaliera aggiustata e l'uscita del modello di ML erano tutte percentuali, che venivano trasformate in due classi: classe 0 se la percentuale fosse stata sotto l'80%, indicando non-aderenza, classe 1 se la percentuale fosse stata sopra l'80%, indicando aderenza. Il classificatore di ML usato fu XGBoost (Extreme Gradient Boosting), che è una tipologia degli algoritmi di ML basati sul decision tree, mentre il metodo di allenamento fu la 5-fold cross validation. Lo scopo principale dello studio, ovvero prevedere, mediante il classificatore, l'aderenza futura dei diversi pazienti, fu declinato in cinque obiettivi:

- Prevedere l'aderenza di tutto lo studio basandosi sui dati della sola prima settimana dello studio.
- Prevedere l'aderenza di tutto lo studio basandosi sui dati delle prime due settimane dello studio.
- Prevedere l'aderenza della settimana successiva basandosi su quella precedente.
- Prevedere l'aderenza del giorno successivo basandosi sulla settimana precedente.

- Individuare quale caratteristica aveva il valore predittivo migliore.

I risultati ottenuti si basavano sulla media delle prestazioni conseguite dal modello di ML sui cinque validation set, e sono riportati nella tabella 1.

Aderenza	Accuratezza	AUC	PPV	Sensibilità	FP
Tutto lo studio in base ai primi 7 giorni	72.2	0.80	0.76	0.74	0.25
Tutto lo studio in base ai primi 14 giorni	76.6	0.83	0.78	0.78	0.18
Settimana successiva in base a quella precedente	81.3	0.87	0.82	0.82	0.16
Giorno successivo in base alla settimana precedente	81.0	0.87	0.84	0.82	0.17

TABELLA 1: PRESTAZIONI XGBOOST

Le quattro curve ROC mostrarono tutti risultati comparabili (Fig.2.3.1)

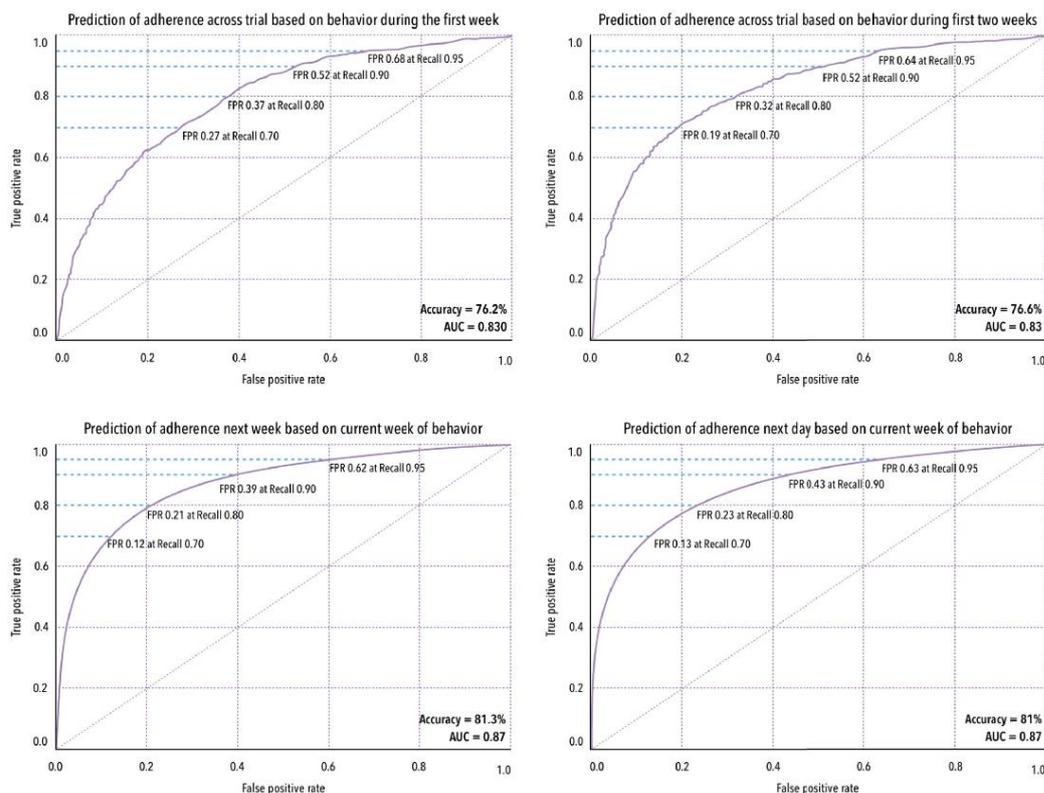


FIGURA 2.3.1: CURVE ROC DEI MODELLO XGBOOST (KOESMAHARGYO ET AL., 2020)

Infine, di tutte le caratteristiche usate come ingresso ai modelli di ML, quella che si rivelò avere più importanza nella previsione dell'aderenza fu la media dell'aderenza dei giorni precedenti, seguita dalla diagnosi del paziente.

I risultati di questo studio mostrano come l'IA possa aiutare gli studi clinici sotto diversi aspetti: utilizzando i dati relativi ai primi 7/14 giorni si può già prevedere quali pazienti siano più propensi ad essere non-aderenti, decidendo quindi o di escluderli dallo studio clinico o di monitorarli in maniera più assidua; durante lo svolgimento dello studio, riuscire a prevedere in modo dinamico i pazienti che potrebbero essere a rischio non-aderenza permette agli organizzatori dello studio clinico di agire tempestivamente, prima ancora che il paziente sia effettivamente non-aderente; inoltre, le prestazioni dei modelli mostrano che come la predizione diventa più dinamica (settimana successiva/giorno successivo), i dati di tipo dinamico diventano più rilevanti di quelli di tipo statico, mostrando quindi come gli strumenti tecnologici che consentano di monitorare dinamicamente il paziente siano di grande necessità. Tutto ciò garantisce un minor tasso di abbandono, a cui consegue un minor impiego di risorse, sia dal punto di vista temporale, evitando ritardi nello svolgimento dello studio clinico, sia in termini monetari.

2.1.3 Attività Neurale

Un tipico problema all'interno delle neuroscienze è quello di voler estrarre, a partire dalla misurazione dell'attività neurale, intenzioni, sensazioni, comportamento o malattie/disturbi. Tuttavia, al giorno d'oggi, il significato dell'attività neurale è spesso non conosciuto, rendendo particolarmente ostica la risoluzione di questo problema mediante approcci tradizionali. Il ML può quindi essere un'ottima strada alternativa, soprattutto nel momento in cui si abbia a

disposizione un grosso dataset di attività neurale relativa alle sensazioni, intenzioni, comportamenti o malattie d'interesse. Riuscire ad estrarre comportamenti ed intenzioni dall'attività neurale può ritrovare applicazione nell'interfaccia neurale (BCI, dall'inglese "Brain Computer Interface"), la quale mira ad usare i segnali neurali per controllare protesi o altri oggetti esterni. Numerosi gruppi di ricerca hanno utilizzato algoritmi di ML, come per esempio le reti neurali ricorsive, al fine di migliorare le BCI utilizzando la registrazione di potenziali d'azione, l'elettrocorticografia o l'EEG. Un esempio di ciò è dato dal lavoro svolto da Anumanchipalli et al. (Anumanchipalli et al., 2019) per quanto riguarda una BCI in grado di ripristinare la funzione vocale. I disturbi neurologici che portano alla perdita dell'abilità di comunicare possono essere devastanti e, spesso, i pazienti fanno affidamento a dispositivi di comunicazione che misurano i movimenti non-verbali residui della testa o degli occhi, o a BCI che permettono al paziente di controllare un cursore consentendo di selezionare le lettere su uno schermo al fine di produrre la parola. Sebbene questi sistemi possano migliorare la qualità della vita del paziente, spesso il numero massimo di parole trasmesse al minuto è di circa dieci, mentre nel linguaggio naturale la media è di 150 parole al minuto. Un'alternativa a questi sistemi è quella di sintetizzare direttamente il linguaggio dall'attività neurale mediante approcci di DL. Anumanchipalli et al. lavorarono con cinque volontari che dovevano sottoporsi ad una procedura detta monitoraggio intracranico, dove degli elettrodi sono impiegati per monitorare l'attività neurale come parte di un trattamento per l'epilessia. Per tracciare l'attività delle aree del cervello relative al controllo del linguaggio e degli articolatori del tratto vocale, ovvero delle strutture anatomiche coinvolte nella produzione del linguaggio (come labbra, lingua, laringe, mascella...) i ricercatori usarono una tecnica, chiamata elettrocorticografia ad alta densità, mentre i volontari pronunciavano diverse frasi. Al fine di ricostruire il linguaggio a partire dai segnali neurali, invece che trasformare direttamente quest'ultimi in un segnale vocale,

Anumanchipalli et al. utilizzarono un approccio di due fasi: nella prima fase trasformarono l'attività neurale in rappresentazioni motorie degli articolatori del tratto vocale, mentre nella seconda fase, a partire da queste rappresentazioni motorie, estrassero le caratteristiche acustiche (come, per esempio, la frequenza fondamentale) al fine di sintetizzare successivamente il linguaggio. In entrambe le fasi, per prevedere le rappresentazioni motorie e le caratteristiche acustiche si utilizzarono due distinte reti neurali ricorsive (bLSTM), dal momento che quest'ultime sono particolarmente adatte a processare segnali aventi una complessa struttura temporale. Il motivo alla base di questo approccio a due fasi è dato dalle osservazioni di studi precedenti che mostrano come l'attività neurale delle aree cerebrali connesse al linguaggio corrisponda maggiormente ai movimenti degli articolatori del tratto vocale piuttosto che al segnale acustico prodotto durante il linguaggio. Infatti, l'approccio di Anumanchipalli et al. risultò in un linguaggio con una distorsione acustica decisamente minore rispetto agli approcci diretti, mostrando come questo approccio possa esserne una valida alternativa. Nonostante ciò, l'intelligibilità del linguaggio ricostruito risultò comunque inferiore rispetto a quella del linguaggio naturale; per questo motivo, ulteriori studi saranno necessari per verificare se questo approccio possa essere migliorato, per esempio collezionando più dati, migliorando gli algoritmi che ne stanno alla base e utilizzando interfacce neurali che utilizzino dati di attività neurali più localizzati rispetto a quelli forniti dall'elettrocorticografia.

L'analisi dell'attività neurale tramite ML può essere utile anche per prevedere l'andamento di attività neurale futura a partire da quella passata. Questa applicazione è rilevante, per esempio, per l'epilessia, che è un disturbo neurologico caratterizzato dalla predisposizione all'insorgenza di crisi epilettiche, ovvero alterazioni dello stato di coscienza, movimenti involontari o convulsioni. Un recente studio, svolto da Talathi (Talathi, 2017), ha utilizzato un particolare tipo di rete neurale ricorsiva (GRU RNN) al fine di rilevare le crisi epilettiche a partire da registrazioni EEG.

Nel rilevamento di crisi epilettiche, il modello RNN ha raggiunto un'accuratezza del 99,6% sul validation set; nel rilevamento precoce delle crisi, il modello ha mostrato un'accuratezza di circa 98% nel prevedere l'evento a partire dai primi cinque secondi della crisi stessa. Modelli in grado di prevedere l'insorgenza di nuove crisi epilettiche potrebbero portare allo sviluppo di nuovi trattamenti in grado di agire tempestivamente sulla crisi stessa, controllandola, mitigandola e riducendone la durata.

L'IA, oltre a permettere di prevedere intenzioni o comportamenti a partire dall'attività neurale, è utile anche per prevedere l'attività neurale sulla base di stimoli esterni. Nell'ambito delle neuroscienze sensoriali, per esempio, una grande sfida è quella di costruire modelli di codifica neurale³ che riescano a prevedere l'attività neurale di un'area sensoriale in risposta a uno stimolo d'interesse. Un esempio di questi modelli è quello proposto da McIntosh et al. (McIntosh et al., 2016) per quanto riguarda la vista: il gruppo di ricerca costruì un modello basato sulla CNN, allenandolo a prevedere le risposte di cellule gangliari a delle immagini naturali o di rumore bianco presentate alla retina. Lo studio mostrò come i modelli basati sulle CNN siano migliori degli altri modelli lineari-non lineari (LN) o lineare generalizzato (GLM), precedentemente molto usati nella modellazione delle risposte retiniche. In particolar modo, lo studio dimostrò come le CNN si avvicinino all'affidabilità della retina sia per il rumore bianco che per le scene naturali, mentre i modelli LN e GLM falliscano nella risposta a scene naturali, come le CNN richiedano meno tempo di allenamento rispetto gli altri modelli, raggiungendo prestazioni migliori, e come le CNN siano più in grado di generalizzare, essendo inoltre meno inclini all'overfitting, rispetto agli altri modelli. Questo tipo di studi sono molto utili per migliorare le prestazioni degli occhi bionici, permettendo loro di eguagliare le capacità di un occhio naturale. In maniera analoga e più generale, le tecniche di ML possono aiutare le neuroprotesi, ovvero dispositivi artificiali in grado di sostituire

³ Per codifica neurale si intende lo studio di come i neuroni rappresentino l'informazione tramite attività elettrica (potenziali d'azione), a livello della singola cellula o della rete di neuroni.

e/o migliorare specifiche funzioni del sistema nervoso, a compiere in maniera più efficiente il loro lavoro.

2.1.4 Segmentazione dei Tumori

L'utilizzo dell'IA si sta rendendo sempre più necessario al fianco di tecniche di neuroimaging, data una sempre più grossa mole di dati proveniente da magnetoencefalografia (MEG), tomografia ad emissione di positroni (PET), MRI strutturale (sMRI), MRI funzionale (fMRI). Questo aumento di volume ingente di dati rende impossibile fare affidamento solo sul lavoro manuale dell'uomo, sia perché ciò richiederebbe troppo tempo sia perché l'insieme di variabili/caratteristiche presente diventerebbe troppo complesso per l'uomo da analizzare. In quest'ottica, si sta cercando di implementare tecniche di ML e DL sempre più abili, per esempio, nella segmentazione di immagini (ovvero nel ripartire un'immagine in regioni significative). Ciò può essere utile per diverse applicazioni cliniche, come la valutazione dell'atrofia cerebrale, la delimitazione di lesioni dovute alla sclerosi multipla o la localizzazione di tumori al cervello. In tutte queste applicazioni, l'algoritmo di IA più usato è quello basato sulle CNN, dal momento che, come accennato nel capitolo precedente, queste reti sono ottime per l'elaborazione di immagini. In particolar modo, nel caso di un tumore, l'IA può aiutare il radiologo nel segmentare l'immagine identificando correttamente l'esatta localizzazione spaziale del tumore.

Nel sistema nervoso centrale (SNC) tra i principali tumori primari, ovvero quelli originatisi nel SNC, vi sono i gliomi, che costituiscono circa il 40% di tutti i tumori del SNC. I gliomi sono dei tumori delle cellule della glia del SNC e ne esistono di diversi tipi, le cui caratteristiche dipendono, principalmente, dal tipo di cellula della glia colpita e dalla velocità di crescita della massa

tumorale. Si suddividono in due grandi gruppi: gliomi a basso grado (grado WHO⁴ I e II), tipicamente poco aggressivi e con una velocità di crescita lenta, e gliomi ad alto grado (grado WHO III e IV), ad alto grado di malignità e aggressività biologica e con crescita rapida. Conoscere le caratteristiche di un glioma come sede, dimensioni e gravità permette di avere una diagnosi più precisa, di impostare la terapia più adeguata e di seguire meglio il paziente durante il follow-up: molto importante è quindi la procedura di segmentazione del tumore, che richiede la distinzione del tessuto tumorale da quello sano circostante e che è normalmente effettuata da un radiologo esperto; al tempo stesso, tuttavia, svolgerla manualmente ha due principali inconvenienti: richiede molto tempo ed è soggetta ad alta variabilità inter-operatore ed intra-operatore. Quindi, metodi di segmentazione automatica sono ampiamente ricercati, dal momento che essi possono facilitare l'ottenimento di misure più consistenti e simultaneamente ridurre il tempo speso dai radiologi nello svolgere questa procedura. Caratteristica principale degli algoritmi di ML o, più spesso in questo compito, di DL deve essere la flessibilità, ovvero la capacità di generalizzare, dal momento che i gliomi possiedono un alto grado di variabilità: possono presentarsi ovunque nel cervello e possono avere forma e dimensione molto diverse tra di loro. Negli anni passati, grossi sviluppi nella segmentazione automatica sono stati fatti grazie all'annuale sfida Brain Tumor Segmentation (BraTS), il cui scopo principale è quello di valutare i metodi all'avanguardia per la segmentazione dei tumori cerebrali nelle scansioni di risonanza magnetica multimodale (MRI). All'interno di questa sfida, il comitato dell'organizzazione rilascia scansioni multimodali di volumi provenienti da un grosso numero di pazienti affetti da glioma, suddivisi in training, validation e test set e segmentati manualmente da neuroradiologi esperti, fornendo il gold standard. Successivamente, differenti gruppi di ricerca si sfidano per cercare di costruire il miglior algoritmo di segmentazione automatica,

⁴ WHO = World Health Organization

valutato nel test set, fornendo quindi di anno in anno algoritmi sempre migliori. Tra quest'ultimi, i più performanti si sono rilevati essere diverse reti neurali convoluzionali, sia 3D che 2D. Inoltre, nelle ultime edizioni della BraTS, per aumentare la rilevanza clinica della sfida, oltre alla valutazione della segmentazione automatica degli algoritmi, ci si è concentrati anche sulla previsione della sopravvivenza del paziente e si è valutata l'incertezza dell'algoritmo nella segmentazione del tumore, facendo scegliere alle squadre gareggianti se competere per tutti i compiti o focalizzarsi solo su uno. Per esempio, nella sfida del 2020, il gruppo vincitore della sezione "segmentazione tumore" (team MIC-DKFZ, Isensee et al., 2020) usò come modello di DL la rete neurale nnU-Net, un particolare tipo di U-net (a sua volta un tipo di rete neurale convoluzionale) in grado di adattarsi autonomamente a qualsiasi nuovo dataset di segmentazione, integrandola con alcune specifiche utili per ottimizzarla alla sfida BraTS. I risultati ottenuti da questa rete nel test set per quanto riguarda la segmentazione del tumore, del core del tumore e dell'"enhancing tumor"⁵ furono riportati con il coefficiente di Dice (un indice per la sovrapposizione dell'area predetta dal modello e quella del gold standard, che varia da un minimo di 0%, che è il caso peggiore, ad un massimo di 100%, che è il caso migliore) e furono di 88.95, 85.06 e 82.03 rispettivamente, dimostrando degli ottimi risultati. Lo stesso gruppo (Isensee et al., 2019), nel 2018, era arrivato secondo nella sfida BraTS, usando un modello U-net detto "No-New Net" (su cui ci si basò poi per arrivare alla nnU-Net) che ottenne nel test set come coefficiente di Dice i valori 77.88, 87.81 e 80.62 rispettivamente per la segmentazione del tumore, del core del tumore e dell'enhancing tumor. Questi risultati mostrano due aspetti chiave: il primo è che particolarmente adatte al compito della segmentazione sono le reti neurali convoluzionali, cosa che non stupisce troppo data la risaputa abilità delle CNN nell'elaborare immagini; il secondo aspetto chiave è la tendenza

⁵ Tumor, core tumor ed enhancing tumor sono alcune delle sotto-regioni in cui viene delineato un tumore.

nel miglioramento delle prestazioni di questi modelli, rendendoli sempre più promettenti per l'utilizzo clinico a supporto del radiologo. Tuttavia, al giorno d'oggi questi modelli di IA non trovano impiego nella pratica clinica. Una spiegazione di ciò è che non esiste un solo modello, ma ne esistono molti che, seppur sempre basati sulle CNN, hanno caratteristiche diverse e portano a prestazioni diverse. Inoltre, non esiste ancora una procedura standardizzata che spieghi nel dettaglio come usare questi strumenti. Infine, per rendere questi modelli effettivamente utili per la pratica clinica essi non devono aggiungere ulteriori compiti a cui il radiologo deve stare dietro; tuttavia, per il momento, la segmentazione automatica richiede ancora di essere supervisionata da un esperto.

2.1.5 Predizione della Diagnosi e della Prognosi

Due ambiti in cui l'IA è molto usata sono quelli della predizione della diagnosi e della prognosi. Per diagnosi si intende il processo di analisi che permette di determinare da quale patologia sia affetto un paziente analizzandone sintomi, segni e storia clinica; la prognosi, invece, è una previsione sull'evoluzione e sul decorso di una malattia. L'IA può aiutare i medici nella diagnosi di disturbi, come il disturbo depressivo maggiore o i disturbi dello spettro autistico, e malattie, come il cancro o il morbo di Alzheimer; nel far ciò, l'IA è tipicamente usata in due fasi temporali distinte: può essere usata per prevedere l'eventuale insorgenza della malattia/disturbo, aiutando i medici a prevenire quest'ultima, oppure può essere usata quando la malattia/disturbo è già presente. In quest'ultimo caso, l'IA può essere utile per determinare se il paziente ha effettivamente o meno la malattia/disturbo e, in caso di risposta positiva, per definirla meglio, come nel caso della segmentazione di un tumore. Nel caso della prognosi, l'IA può aiutare i medici a prevedere l'andamento della malattia/disturbo,

permettendo quindi di scegliere tempestivamente il percorso di cura più adatto ed efficiente per il paziente.

Un esempio di utilizzo dell'IA nella diagnosi è quello dato dal lavoro di Abedi et al. (Abedi et al., 2017) per quanto riguarda l'ischemia cerebrale acuta (ACI), ovvero una condizione nella quale il cervello non riceve il giusto apporto di sangue. L'ACI rappresenta un'emergenza medica che, se non trattata tempestivamente, può portare all'infarto cerebrale, cioè all'ictus, e quindi alla morte o a delle disabilità permanenti. Dato il breve lasso di tempo in cui si può intervenire prima che vi siano danni irreparabili, riuscire a diagnosticare in tempi brevi l'ACI è estremamente importante per prevenire danni al cervello e migliorare i sintomi. Altro aspetto importante dell'ACI è quello di riuscire a discriminarla dagli "stroke mimics", ovvero da tutte quelle condizioni mediche non dovute all'ACI ma in cui i sintomi sono simili; è essenziale riuscire a diagnosticare correttamente gli stroke mimics, evitando di confonderli con l'ACI, al fine di non utilizzare in modo inappropriato procedure mediche costose e potenzialmente dannose. Scopo del lavoro di Abedi et al. fu quindi quello di sviluppare un modello di apprendimento supervisionato al fine di riconoscere tempestivamente l'ACI e discriminarla dagli stroke mimics in condizioni di emergenza. Il modello di IA usato fu una rete neurale artificiale (ANN), allenata mediante la retropropagazione dell'errore. Il database iniziale era composto da pazienti consecutivi che si presentavano con sintomi connessi all'ictus, entro 4,5 ore dall'insorgenza dei sintomi stessi, in due centri per l'ictus di cure terziarie (ovvero dei centri in cui si erogano prestazioni altamente specializzate) durante un periodo di tre anni. La diagnosi effettiva, che avrebbe dovuto fare da gold standard per il modello di IA, era ottenuta mediante diversi esami, quali TAC alla testa, ECG e diversi tipi di RM e da caratteristiche quali età, sesso, presenza di ipertensione e diabete; tutti i pazienti a cui non era possibile effettuare uno o più di questi esami o a cui non era possibile risalire ad una o più caratteristiche venivano esclusi dal database. Al fine di avere un database bilanciato, tolti dal database

iniziale i pazienti con dati mancanti, vennero selezionati in modo casuale 130 pazienti con l'ACI e 130 con stroke mimic, per un totale di 260 pazienti. Vennero creati dieci modelli di ANN, allenati con una 10-fold cross validation; ogni training set e validation set conteneva 13 pazienti con l'ACI e 13 con lo stroke mimic. Basandosi sui risultati ottenuti mediante la cross validation, la sensibilità media e la specificità media dei modelli di ANN per quanto riguarda la diagnosi dell'ACI furono rispettivamente dell'80.0% e dell'86.2%. Il valore predittivo positivo medio per la diagnosi dell'ACI e degli stroke mimics fu invece dell'85.2% e dell'81.1% rispettivamente. I risultati di questo studio pilota dimostrano come l'utilizzo delle ANN al fine di diagnosticare l'ACI e per discriminare l'ACI dagli stroke mimics sia un valido strumento, soprattutto in situazioni di emergenza. Al tempo stesso lo studio ha dei limiti: per esempio, non venivano distinti i pazienti con ischemia alla circolazione anteriore del cervello da quelli con ischemia alla circolazione posteriore, così come non venivano prese in considerazione le varie cause dell'ischemia; per questo motivo, e per verificare se un maggior numero di pazienti e di caratteristiche possano incrementare le prestazioni del modello di IA, ulteriori studi sono necessari.

Nel caso della prognosi, una delle malattie più studiate è il morbo di Alzheimer (AD, Alzheimer's disease). Quest'ultimo è caratterizzato da un processo dinamico di cambiamento neurocognitivo che va da una cognizione normale, per poi proseguire in un "Disturbo Cognitivo Lieve" (MCI, mild cognitive impairment), caratterizzato da problemi di memoria, linguaggio, pensiero e ragionamento, fino ad arrivare alla condizione di demenza. Tuttavia, non tutte le forme di MCI sono dei precursori del morbo di Alzheimer, tendendo infatti a rimanere stabili nel tempo senza sfociare nella demenza. Al fine quindi di individuare e scegliere le migliori strategie personalizzate per prevenire o rallentare la progressione all'AD, è condizione necessaria riuscire a distinguere i pazienti affetti da MCI a rischio di AD da quelli affetti da MCI ma che resteranno stabili nel tempo. È in questo

contesto che si inserisce il lavoro svolto da Spasov et al. (Spasov et al., 2019), nel quale fu sviluppato una particolare architettura di rete neurale di deep learning il cui scopo era quello di identificare, all'interno del gruppo di pazienti affetti da MCI, quelli che avevano una maggior probabilità di degenerare nell'AD nell'arco di tre anni. Il modello di deep learning riceveva in ingresso diversi dati, tra cui dati genetici, demografici, neuropsicologici ed immagini strutturali MRI, provenienti da un sottoinsieme di partecipanti appartenenti al database ADNI (Alzheimer's Disease Neuroimaging Initiative). Il totale dei partecipanti era di 785, di cui 192 pazienti affetti da AD, 409 pazienti affetti da MCI (sia pazienti stabili, sia pazienti che sarebbero peggiorati nell'AD) e 184 persone sane di controllo. L'allenamento e la valutazione della rete neurale vennero implementati mediante la suddivisione del database nei tre gruppi di training, validation e test set e applicando una 10-fold cross validation, mentre i parametri usati per valutare il modello furono l'accuratezza, la sensibilità, la specificità, la curva ROC e l'area AUC. L'algoritmo usato fu in grado di distinguere i pazienti MCI che avrebbero sviluppato l'AD entro tre anni dai pazienti MCI stabili nello stesso arco temporale con un'AUC pari a 0.925, un'accuratezza dell'86%, una sensibilità dell'87.5% e una specificità dell'85%. La stessa rete fu in grado di discriminare i pazienti affetti da AD da quelli sani con un'AUC pari a 1 ed un valore dell'accuratezza, sensibilità e specificità del 100%. Le novità introdotte da questo modello rispetto ai precedenti modelli furono diverse: in primo luogo, era un modello di rete neurale multi-tasking, ovvero in grado di discriminare simultaneamente sia i pazienti MCI stabili da quelli che avrebbero poi sviluppato l'AD, sia i pazienti sani da quelli affetti da AD; inoltre, usava molti meno parametri rispetto alle altre architetture di deep learning, con un conseguente minor rischio di overfitting. Al tempo stesso era un modello molto flessibile, in grado di integrare altre modalità d'imaging, come la PET, ed altri dati clinici, ed era un modello convoluzionale generico, applicabile quindi a qualsiasi dataset di immagini 3D e facilmente utilizzabile

come sistema CAD (computer aided diagnostic) al fine di prevedere e classificare problemi in qualsiasi ambito medico utilizzando immagini multimodali ed altri dati clinici.

2.3 Selezione delle Caratteristiche

Un altro utilizzo dell'IA è quello della “selezione delle caratteristiche”, che è una forma di riduzione della dimensionalità di un determinato dataset, che mira ad individuare le variabili (le caratteristiche) rilevanti per l'abilità predittiva di un modello. Oltre a quello appena citato, i motivi per cui questa tecnica viene utilizzati sono: semplificazione dei modelli per renderli più facili da interpretare, tempi di addestramento minori e minor rischio di overfitting. Nell'ambito delle neuroscienze, ciò può servire per capire quali variabili del mondo esterno siano più responsabili di determinate attività neurali. In maniera speculare, questa tecnica può servire per capire, partendo dall'attività neurale registrata, quali aree cerebrali siano più rilevanti per lo svolgimento di alcuni compiti; oppure, può servire per determinare quali variabili siano più importanti per il comportamento. Per esempio, Coen et al. (Coen et al., 2016) studiarono come le caratteristiche ambientali prevedessero l'intensità della canzone prodotta dalla *Drosophila* durante il corteggiamento, trovando come variabile più rilevante la distanza tra il maschio e la femmina e permettendo quindi il successivo studio della rete neurale responsabile della modulazione dell'intensità della canzone basata sulla distanza.

Al tempo stesso, nell'ambito clinico, la selezione delle caratteristiche è importante per riuscire a capire quali siano le caratteristiche predittive per determinate malattie/disturbi. L'approccio generale utilizzato per tale compito si suddivide in due fasi: nella prima fase si costruisce un algoritmo di ML o DL, che serve per determinare se un soggetto ha la malattia/disturbo; successivamente, dopo aver testato l'efficacia dell'algoritmo, si

applica il processo di selezione delle caratteristiche, andando a valutare quale siano le caratteristiche più rilevanti che permettono al classificatore di avere quelle prestazioni. Un esempio di ciò è dato dal lavoro svolto da Deshpande et al. (Deshpande et al., 2015): il gruppo di lavoro usò un particolare tipo di rete neurale, la fully connected cascade (FCC), per distinguere, a partire da dati di fMRI, pazienti soggetti da Disturbo da Deficit di Attenzione/Iperattività (ADHD) da soggetti sani, raggiungendo un'accuratezza di quasi il 90%, e per riuscire a distinguere tra di loro diversi sottotipi di ADHD, arrivando ad un'accuratezza di circa il 95%. A partire da questi risultati, infine, andarono alla ricerca della caratteristica più rilevante per la corretta categorizzazione, dimostratasi essere un'alterata e ridotta connettività nella corteccia orbitofrontale sinistra e in varie regioni cerebellari presente nei pazienti ADHD. Uno studio analogo fu svolto da Kim et al. (Kim et al., 2016), usando però come algoritmo una rete neurale profonda (DNN) e avente come oggetto di ricerca la schizofrenia, basandosi su dati di connettività funzionale ottenuti mediante “resting-state” fMRI. Lavori simili forniscono “indizi” su quello che è la fisiopatologia, ovvero le modificazioni presentate dalle funzioni organiche, di numerose malattie/disturbi ad oggi ancora poco compresi, gettando le basi per progetti futuri atti a permettere di capirle meglio.

2.3 Modellistica del Cervello

Un ulteriore utilizzo dell'IA all'interno delle neuroscienze è quello di fungere da modello per il cervello stesso, permettendo di comprendere meglio come esso funzioni. L'idea che le reti neurali profonde (DNN) siano un buon modello per il cervello, sebbene ad un alto livello di astrazione, è diventata particolarmente popolare nell'ultimo decennio, grazie ai recenti progressi ottenuti dalle DNN nei compiti comportamentali, come nel riconoscimento di immagini. All'interno del sistema visivo, numerose sono le

somiglianze evidenziate da altrettanto numerosi studi tra le DNN e l'apparato visivo umano: dal punto di vista strutturale, sia la corteccia visiva che i modelli DNN presentano un'organizzazione multistrato e gerarchica, con strati che man mano estraggono caratteristiche sempre più astratte; inoltre, come mostrato da due diversi lavori di Kheradpisheh et al. (Kheradpisheh et al., 2016), le DNN commettono gli stessi tipi di errori nel riconoscere le immagini e hanno l'abilità di raggiungere l'accuratezza umana nel riconoscimento invariante degli oggetti (ovvero sono in grado di riconoscere uno stesso oggetto anche in condizioni diverse, come da diversi punti di vista, con un differente sfondo, una luminosità diversa, etc.), arrivando anche a superarla nel caso di reti molto profonde.

Le analogie tra le reti neurali e il cervello non si limitano solo all'apparato visivo: come mostrato da Kell et al. (Kell et al., 2018), reti neurali allenate nel riconoscimento vocale e nel prevedere il genere musicale di brani presentano un'attività simile alla corteccia uditiva; Sussillo et al. (Sussillo et al., 2015) hanno dimostrato come reti neurali ricorsive allenate per riprodurre movimenti motori delle scimmie presentino un'attività molto simile a quella neuronale registrata nella corteccia motoria delle scimmie stesse. Tutte queste similarità tra le reti neurali e il cervello sono un segno di come questi modelli potrebbero catturare importanti aspetti dell'elaborazione fatta dal cervello, fornendo degli ottimi strumenti che permettano di comprendere sempre meglio il funzionamento del cervello stesso.

Capitolo 3: L'Intelligenza Artificiale per la Salute Mentale

Un campo all'interno del quale l'IA si sta rivelando di grande successo è quello della salute mentale, ovvero di tutti quegli aspetti che, in assenza di patologie e disturbi, rappresentano una condizione di sanità, benessere e/o equilibrio di tipo psicologico ed emotivo. Mentre l'IA è stata adottata quasi subito all'interno della medicina per la salute fisica, la disciplina della salute mentale è stata più lenta ad integrare l'IA. Il motivo alla base di ciò sono diversi: in primo luogo, i professionisti che lavorano in questa disciplina fanno molto affidamento a soft-skills, come creazione di relazioni con i pazienti e l'osservazione diretta dei comportamenti e delle emozioni dei pazienti; inoltre, i dati clinici sulla salute mentale sono spesso sotto forma di affermazioni soggettive e qualitative del paziente e note scritte, rendendo l'integrazione con l'IA più complessa. Al tempo stesso, però, l'utilizzo di tecnologie basate sull'IA può portare a grossi vantaggi. In particolar modo, un grosso aiuto è quello fornito dall'IA nell'aspetto patologico della salute mentale, ovvero per quanto riguarda i disturbi mentali, quali disturbo depressivo maggiore, schizofrenia, disturbo bipolare, disturbi dello spettro autistico ma anche ideazione e tentativi di suicidio. Le fasi in cui l'IA può essere impiegata sono diverse: inizialmente, l'IA può servire per definire in maniera più oggettiva i disturbi mentali rispetto a quanto fa il DSM-5 (ovvero il “Manuale diagnostico e statistico dei disturbi mentali”, uno dei sistemi nosografici per i disturbi mentali o psicopatologici più utilizzati da psichiatri, psicologi e medici di tutto il mondo), identificando questi disturbi in maniera tempestiva fin dai primi sintomi ed adattando alle caratteristiche uniche del paziente il trattamento, rendendo quest'ultimo più efficiente. Per esempio, sfruttare le tecniche di intelligenza artificiale offre la capacità di sviluppare strumenti di screening prediagnosi migliori e di formulare modelli di rischio per determinare la predisposizione o il rischio di

sviluppare malattie mentali di un individuo (Graham et al., 2019). Al tempo stesso, l'IA può essere d'aiuto nel prevedere l'efficacia di un trattamento, argomento che sarà meglio discusso successivamente. In questa tesi ci si è concentrati sull'utilizzo dell'IA nell'ambito dei disturbi mentali e, più nello specifico, per quanto riguarda il disturbo depressivo maggiore (DDM) e i disturbi dello spettro autistico (DSA).

3.1 L'IA per il Disturbo Depressivo Maggiore

Il disturbo depressivo maggiore è uno tra i più comuni disturbi mentali a livello mondiale, con una prevalenza nel corso della vita del 10.8%⁶. Il DDM rientra all'interno della depressione, un disturbo dell'umore che può presentarsi sia come singolo episodio, prendendo il nome di episodio depressivo, sia come vero e proprio disturbo, chiamato disturbo depressivo. A loro volta, sia il l'episodio che il disturbo depressivo possono avere una minore o maggiore gravità; nel caso di disturbo depressivo con maggiore gravità, che è quello più comune, si parla di DDM. Il DDM è tipicamente caratterizzato da un basso umore, tristezza, perdita di interesse e/o piacere, sensi di colpa e scarsa autostima, sonno e appetito disturbati, bassi livelli di energia, bassa concentrazione e sensazioni di stanchezza. Nelle forme più severe, il DDM può arrivare fino a pensieri di suicidio e un incremento nel rischio di mortalità. Affinché si possa parlare propriamente di DDM, devono essere presenti almeno cinque dei sopracitati sintomi, presenti quasi ogni giorno per due settimane. Per questo motivo, il disturbo depressivo maggiore è un disturbo invalidante, che coinvolge spesso sia la sfera affettiva che quella cognitiva della persona,

⁶ Ovvero il 10.8% della popolazione va incontro ad almeno un episodio di depressione nel corso della sua vita.

influenza negativamente sulla qualità della vita, sia familiare che lavorativa, sullo studio, sulle abitudini alimentari, sul sonno e sulla salute fisica, con forte impatto dunque sullo stile di vita e la qualità della vita in generale. Tipicamente la sintomatologia del DDM ha un decorso lento ed insidioso, che può variare da paziente a paziente e che può nel tempo o aggravarsi, portando anche a cronicità, o andare incontro a remissione. In accordo con i report dell'Organizzazione Mondiale della Sanità, ci sono oltre 300 milioni di persone che vivono con il DDM, e ogni anno circa 800 mila pazienti muoiono a causa del suicidio derivante dal DDM. Sebbene ad oggi esistano oltre 20 farmaci antidepressivi, suddivisi in diverse classi, e diversi trattamenti per il DDM, l'efficacia di quest'ultimi è piuttosto bassa. Ciò è dovuto principalmente a due fattori: il primo è che non si sa di preciso a cosa sia dovuto il DDM; il secondo è che esistono diversi tipi di DDM e la diagnosi di quest'ultimi è, come scritto in precedenza, fortemente basata su affermazioni soggettive e qualitative del paziente, quindi spesso non troppo oggettiva. L'approccio tradizionale consta quindi nell'individuare quella che si presume essere la forma di DDM che affligge il paziente e, sulla base di quest'ultima, scegliere la classe del farmaco antidepressivo più adatta. Tuttavia, proprio per la difficoltà nell'individuare precisamente la forma di DDM, questo approccio risulta essere per molti pazienti un metodo "trial-and-error", ovvero dove si provano diversi farmaci (errati) finché non si trova quello giusto. Mediamente, due pazienti su tre non rispondono al primo trattamento e per molti soggetti sono necessari oltre quattro differenti trattamenti antidepressivi (Khodayari-Rostamabad et al., 2013). Ovviamente ciò ha diversi svantaggi, dal momento che richiede molto tempo (ogni farmaco va testato tipicamente per circa un mese o anche più, prima di poter effettivamente dire se funziona o meno), è molto costoso e abbatte moralmente ancor di più il paziente, aumentando il rischio di suicidio, oltre ad esporlo ad un trattamento inefficace.

Per questo motivo, ad oggi, uno dei maggiori impieghi dell'IA nell'ambito del DDM è quello di prevedere, basandosi su diversi

dati quali questionari, dati clinici, demografici, genetici e imaging, se un determinato paziente risponderà o meno ad uno specifico trattamento. Un esempio di ciò è quello del lavoro svolto da Al-Kaysi et al. (Al-Kaysi et al., 2017), nel quale si usò un approccio basato sul machine learning al fine di prevedere, basandosi principalmente su dati derivanti dall'elettroencefalografia, quei pazienti affetti da DDM che avrebbe risposto positivamente alla stimolazione transcranica a corrente continua (tDCS). Quest'ultima è un metodo sicuro e non invasivo utilizzato per trattare disturbi neuropsichiatrici e modulare l'attività corticale, mediante l'applicazione di una corrente, tipicamente da 1-2 mA, attraverso il cervello tra due o più elettrodi posizionati sullo scalpo. Oltre ad avere diversi vantaggi rispetto ad altri trattamenti, come un buon rapporto costo-efficacia e pochi effetti avversi, diversi studi hanno dimostrato come la tDCS possa ridurre i sintomi della DDM e migliorare le funzioni cognitive nei pazienti affetti da DDM. Tuttavia, oltre al fatto che il meccanismo alla base del suo effetto antidepressivo sia ad oggi ancora non ben compreso, la stimolazione tDCS risulta essere inefficace nell'80% dei pazienti. Al fine quindi di evitare di sottoporre inutilmente un paziente alla tDCS, Al-Kaysi et al. cercarono di prevedere, utilizzando tre algoritmi di machine learning (support vector machine, extreme learning machine e linear discriminant analysis) e basandosi sulle caratteristiche dell'EEG a riposo registrato prima che il paziente iniziasse la terapia, i pazienti affetti da DDM che avrebbero risposto alla tDCS da quelli non responsivi. In particolar modo, compito degli algoritmi era quello di prevedere l'eventuale miglioramento dell'umore e delle funzioni cognitive nei pazienti affetti da DDM, a partire dalla potenza spettrale dell'EEG. L'EEG venne valutato sia prima del trattamento, in condizioni di riposo e durante lo svolgimento di un compito cognitivo, sia durante lo svolgimento del trattamento; di queste registrazioni, solo l'EEG a riposo prima del trattamento fu usato per allenare i modelli. Per lo studio vennero selezionati dieci pazienti, tutti aventi avuto almeno un episodio di DDM e con una "Montgomery-Åsberg Depression

Rating Scale (MADRS)⁷ superiore o uguale a 20 all’inizio dello studio. Il trattamento tDCS era formato da un totale di 30 sessioni, e il punteggio MADRS fu valutato alle sessioni 8, 15, 23, 30 e dopo una settimana e un mese dalla fine del trattamento. Per allenare i modelli di machine learning venne usata la densità spettrale di potenza (PSD) media delle frequenze delta (0.5–4 Hz), theta (4–8 Hz), alpha (8–12 Hz), beta (13–30 Hz) e gamma (30–100 Hz), calcolata a partire da 62 canali EEG. Oltre a ciò, venne calcolata l’asimmetria alpha (ovvero un parametro indicativo della differenza di potenza nella frequenza alpha tra i due emisferi) nella corteccia frontale, centrale e parietale prima e durante lo svolgimento della TDCS

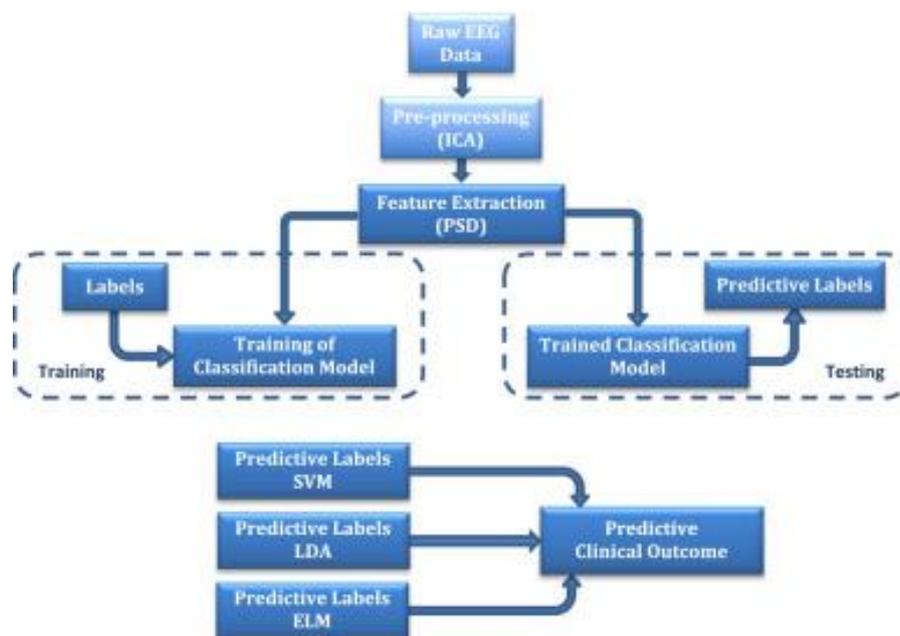


FIGURA 3.1.1: SCHEMA DI ALLENAMENTO E VALUTAZIONE DEI TRE MODELLI DI MACHINE LEARNING. I LABELS SONO GLI OUTPUT DEI MODELLI, OVVERO VALORE 0 PER UNO SCARSO O NULLO MIGLIORAMENTO E VALORE 1 PER UN BUON MIGLIORAMENTO. (AL-KAYSI ET AL., 2017)

Il motivo per cui vennero usati tre modelli basati su differenti meccanismi di apprendimento fu per minimizzare i bias associati a

⁷ La scala MADRS è comunemente utilizzata per valutare la gravità dei disturbi dell'umore, della concentrazione, della condizione fisica, disturbi del sonno riscontrati negli stati depressivi

ciascuno di essi; per lo stesso motivo, dei risultati della classificazione degli algoritmi, allenati e testati mediante la leave-one-out cross validation (perché il numero di soggetti era molto basso), ne venne fatta una media unica. Alla PSD (input dell'algoritmo), al fine di allenare il modello, venne associato come output un valore 1 se il punteggio MADRS del paziente, calcolato alle sessioni 15 e 23, fosse stato almeno in uno dei due casi minore della metà del punteggio MADRS calcolato prima dell'inizio del trattamento (identificando in questo modo i pazienti responsivi), e un valore 0 in tutti gli altri casi. Al tempo stesso, le funzioni cognitive vennero valutate mediante il "Symbol Digit Modalities Test" (SDMT), calcolato prima e dopo la sessione 1 del trattamento. In questo caso, venne assegnato un valore di 1 ai pazienti che mostravano un incremento di 5 punti o più nel SDMT (mostrando un buon miglioramento) e un valore di 0 negli altri casi. Mentre il punteggio MADRS serviva per valutare l'evoluzione dell'umore del paziente, il punteggio SDMT serviva per valutare le funzioni cognitive del paziente. I due punteggi non sempre mostrarono lo stesso andamento: infatti, mentre due pazienti mostrarono un basso miglioramento, ovvero valore 0, in entrambe le scale e tre pazienti mostrarono un grosso miglioramento, ovvero valore 1, in entrambe le scale, i restanti cinque partecipanti avevano valori discordanti nelle due scale. I tre modelli di machine learning vennero allenati e testati singolarmente per i punteggi MADRS e SDMT, utilizzando sia dati provenienti da un solo canale EEG, che da coppie di canali. Per quanto riguarda i singoli canali, i migliori furono il canale Tp9, con un'accuratezza del $71 \pm 11\%$, e il canale Pz, con un'accuratezza dell' $87 \pm 5\%$, rispettivamente per la previsione dell'umore e dell'aspetto cognitivo. In generale, i canali frontali fornirono il miglior risultato medio per l'umore, mentre per la cognizione il risultato medio migliore fu quello dei canali parietali. Nel caso delle coppie di canali, i risultati migliori furono quelli delle coppie FC4-AF8, con un'accuratezza del $76 \pm 11\%$, e CPz-CP2, con un'accuratezza del $92 \pm 4\%$, rispettivamente per l'umore e la cognizione. Le prestazioni medie migliori furono

ottenute dalle coppie di canali nella regione frontale per l'umore e nella regione parietale per la cognizione. I risultati dei singoli algoritmi furono tra loro molto simili e sono illustrati nella figura (Fig. 3.1.2).

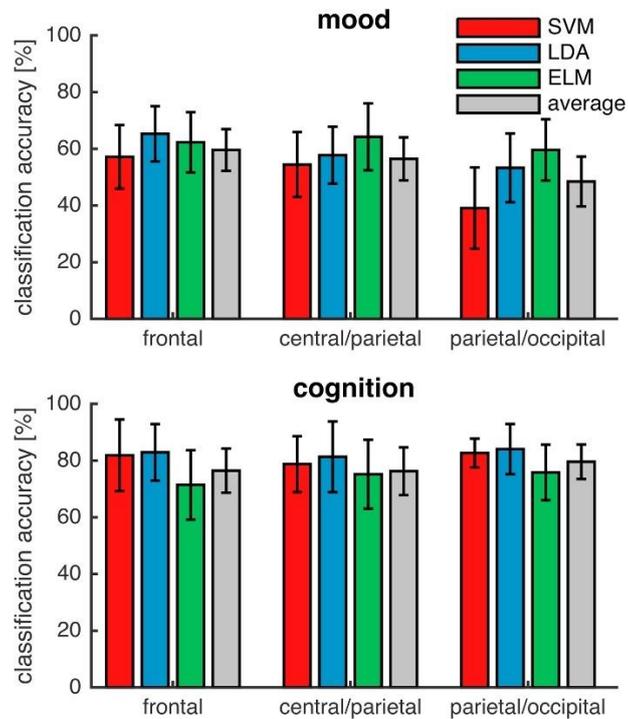


FIGURA 3.1.2: MEDIA E SINGOLI VALORI DELLE PRESTAZIONI DEI TRE MODELLI DI ML (AL-KAYSI ET AL., 2017)

Infine, venne valutato l'andamento della asimmetria alpha: inizialmente, l'indice era negativo nelle regioni frontali e centrali del cervello, indicando una maggior attività nell'emisfero destro di queste zone, mentre era positivo nel lobo parietale, indicando viceversa una maggior attività nel lobo parietale sinistro. Durante il trattamento, l'indice di asimmetria negativo sembrò accentuarsi maggiormente nei pazienti responsivi rispetto ai non-responsivi. Sebbene tale risultato non sia completamente affidabile dato il basso numero di pazienti, analogo risultato fu ottenuto in studi precedenti, dove il valore dell'asimmetria alpha si è rivelato essere un possibile indice di discriminazione della risposta ai trattamenti farmacologici per la depressione, con i pazienti responsivi al

trattamento che mostrarono un maggiore potere alfa sull'emisfero destro rispetto all'emisfero sinistro (Bruder et al., 2001). Nel lavoro di Al-Kaysi et al., i pazienti responsivi mostrarono una maggiore asimmetria alpha destra e un valore MADRS prima del trattamento leggermente più alto dei pazienti non-responsivi. Tuttavia, dal momento che i due gruppi erano composti entrambi da solo cinque partecipanti, ulteriori e più grandi studi sono necessari per valutare il valore predittivo dell'asimmetria alpha. Sebbene i risultati raggiunti dallo studio di Al-Kaysi et al. mostrino come l'EEG a riposo valutato prima del trattamento possa avere un buon valore predittivo nei confronti del risultato del trattamento tDCS, è importante notare come, a causa del basso numero di partecipanti, alcune coppie di canali produssero dei risultati simili a dei tentativi casuali (ovvero con un errore del 50%). Il motivo alla base di ciò è che l'algoritmo allenato su questi canali non fu in grado di generalizzare e discriminare i due output (valore 0 e 1) durante l'allenamento, proprio per il basso numero di pazienti. In definitiva, questi validi ed incoraggianti risultati potranno aver un importante risvolto per l'adozione del trattamento tDCS per la depressione, permettendo di individuare fin da subito i pazienti responsivi ad esso ed escludendo invece quelli non-responsivi, facendo risparmiare tempo e risorse. Al tempo stesso, dato l'esiguo numero di partecipanti, al fine di validare questi risultati saranno necessari altri studi, con una coorte di pazienti maggiore.

Un altro esempio, basato sempre sullo studio dell'EEG, volto a prevedere l'efficacia di un trattamento antidepressivo è quello dato dalla ricerca di Ahmad Khodayari-Rostamabad et al. (Khodayari-Rostamabad et al., 2013): in questo caso il trattamento usato non fu più il tDCS, bensì la classe di “inibitori selettivi della ricaptazione della serotonina” (SSRI, selective serotonin reuptake inhibitor). In questo studio vennero selezionati 22 partecipanti, tutti affetti da DDM, resistenti ai trattamenti⁸ e con una severità della depressione, misurata mediante la scala “17 item Hamilton

⁸ Esistono diverse definizioni di paziente resistente ai trattamenti; in questo studio, con ciò si faceva riferimento a pazienti che non avevano risposto ad almeno due trattamenti precedenti.

depression rating scale (HamD-17), maggiore o uguale a 18 (quindi con una depressione almeno moderatamente severa). Tutti i partecipanti furono trattati con una dose standard di SSRI per sei settimane, sotto la supervisione di uno psichiatra esperto, e alla fine del trattamento venne loro misurato il punteggio HamD-17. Il segnale EEG venne acquisito sia sei settimane prima dall'inizio del trattamento, che dopo due settimane dall'inizio del trattamento e anche dopo circa dieci giorni dalla sua fine. Per ogni paziente si usarono sei registrazioni EEG di 60 secondi ciascuna, tutte ottenute in una condizione di riposo e opportunamente filtrate e trattate, dalle quali si prelevarono successivamente tutte le caratteristiche usate come input al modello di machine learning. Un paziente veniva definito responsivo (R) qualora ottenesse un miglioramento di almeno il 30% tra il punteggio pre- e post-trattamento HamD-17; in caso contrario, il paziente veniva definito non responsivo (NR). Da ogni registrazione EEG vennero estratte N_c caratteristiche: di queste, usando una procedura di selezione di caratteristiche (nello specifico, venne usata la "Fisher discriminant ratio (FDR)"), ne vennero selezionate N_r (con $N_r \ll N_c$), ovvero quelle più statisticamente rilevanti⁹, al fine di evitare l'overfitting. Ad ognuna di queste N_r caratteristiche, usate come input dell'algoritmo di ML, venne associato il corrispettivo output, ovvero classe R o classe NR; inoltre, l'input dell'algoritmo di ML poteva essere 1-D, ovvero composto da una sola caratteristica, oppure M-D, quindi composto da M caratteristiche. L'implementazione del caso 1-D era più semplice del caso M-D, ma al tempo stesso forniva delle prestazioni peggiori. Tra le caratteristiche inizialmente estratte vi furono il logaritmo della PSD di ogni elettrodo e di tutte le frequenze d'interesse, la "magnitude squared spectral coherence (MSC)¹⁰" tra tutte le coppie di elettrodi in tutte le frequenze d'interesse e l'asimmetria di potenza tra i due emisferi. Il modello di ML impiegato fu il

⁹ In questo caso, per statisticamente rilevanti si fa riferimento a quelle caratteristiche che differiscono maggiormente tra i pazienti R e i pazienti NR.

¹⁰ La MSC è una misura di quanto sono simili tra di loro due segnali

“mixture of factor analysis (MFA)”, un particolare tipo di classificatore che restituiva come output la probabilità che il dato fornitogli come input appartenesse alla classe R: se la probabilità era sopra una determinata soglia, l’input veniva classificato come R, viceversa come NR. Dal momento che per ogni partecipante venivano fornite in ingresso all’MFA diverse caratteristiche delle varie registrazioni EEG, la probabilità finale veniva calcolata come media degli output dell’MFA e successivamente convertita nella classe R o NR. Al fine di allenare e valutare il MFA venne usata una 11-fold cross validation (o leave-2-out cross validation), con ogni sottoinsieme formato dalle caratteristiche relative a due partecipanti, e i risultati disposti in una matrice di confusione, calcolandone la specificità, la sensibilità e la media delle due. Nel caso M-D, utilizzando cinque tra le caratteristiche più significative dell’EEG registrato prima dell’inizio del trattamento si ottennero una specificità dell’80.93% e una sensibilità del 94.86%, con una media di 87.9%. Usando sempre lo stesso MFA allenato con le caratteristiche derivanti dall’EEG registrato prima del trattamento, ma utilizzando come input all’algoritmo cinque caratteristiche derivanti dall’EEG acquisito dopo due settimane dall’inizio del trattamento si ottennero una specificità del 78.6% e una sensibilità del 100%, con una media del 89.29%; ciò dimostrò come la previsione dell’efficacia del trattamento potesse essere valutata non solo prima del trattamento, ma anche mentre il trattamento era in corso, riuscendo quindi a proseguirlo o a bloccarlo in corso d’opera a seconda delle previsioni del modello. In generale, nel caso 1-D le prestazioni ottenute furono peggiori di quelle del caso M-D. I risultati ottenuti da questo studio mostrarono inoltre come la caratteristica MSC, che è una misura della connettività del cervello, fosse tra le più significative, in particolar modo tra gli elettrodi T3 e T5: sia la MSC che le regioni del cervello relative agli elettrodi T3 e T5 furono già studiate in precedenza da altre ricerche infatti, anche se principalmente per distinguere le persone affette da vari disturbi mentali dalle persone sane, piuttosto che per la predizione dell’efficacia di un trattamento. Inoltre, l’alto potere

predittivo delle caratteristiche selezionate mostrò come l'utilizzo dell'FDR sia un valido approccio per l'estrazione delle caratteristiche, permettendo di individuare quelle più significative anche partendo da un grosso insieme di caratteristiche, riducendo i tempi ed evitando il fenomeno dell'overfitting. Le prestazioni ottenute dal modello MFA suggerirono come, basandosi sulle sole caratteristiche ottenute dalle registrazioni EEG a riposo precedenti al trattamento, fosse possibile prevedere fin da subito quali pazienti saranno responsivi ad un determinato trattamento, in questo caso all'SSRI; allo stesso tempo, dal momento che l'output dell'MFA è un valore continuo e non direttamente una classe, è possibile estendere l'utilizzo dell'MFA a più trattamenti in contemporanea, riuscendo non solo a prevedere se un paziente sarà responsivo o meno ad un determinato trattamento, ma anche a selezionare il trattamento in cui la probabilità che il paziente sia responsivo è la più alta, ordinando i vari trattamenti in termini di probabilità della loro efficacia. In questo modo si aumenterà la probabilità che il primo trattamento a cui va incontro un paziente sia già quello efficace, evitando il lungo e costoso processo trial-and-error. Infine, nulla vieta di introdurre, oltre alle caratteristiche derivanti dall'EEG, anche dati di carattere genetico, clinico o demografico, andando quindi alla ricerca di quei dati con il più alto valore predittivo possibile.

Gli approcci di predizione dell'efficacia del trattamento basati sulle registrazioni EEG non sono gli unici ad essere stati studiati: diverse ricerche si sono focalizzate infatti anche su dati clinici, genetici e/o demografici, così come sull'imaging. Un esempio di ciò è dato dallo studio svolto da Kautzky et al. (Kautzky et al., 2015) nel quale venne sviluppato un modello di IA in grado di analizzare il ruolo interattivo di variabili cliniche e genetiche nei pazienti affetti da depressione resistente ai trattamenti (TRD, Treatment-resistant depression), al fine di riuscire a discriminare i pazienti responsivi al trattamento da quelli resistenti. Come dati genetici vennero usati i "polimorfismi a singolo nucleotide" (SNP), ovvero una delle più comuni variazioni del materiale genico a carico di un unico

nucleotide. La scelta di studiare simultaneamente più variabili genetiche e cliniche fu dettata dalla logica che, data la complessa natura del DDM, è plausibile ipotizzare che la maggior parte delle variabili, prese singolarmente, non siano in grado di determinare l'efficacia del trattamento; viceversa, interagendo tra di loro, più variabili studiate contemporaneamente hanno maggior probabilità di prevedere l'efficacia del trattamento. Tutti i partecipanti scelti per questo studio, derivanti da un precedente studio del "Group for the Study of Resistant Depression (GSRD)", dovevano essere diagnosticati con il DDM e aver ricevuto in precedenza almeno un trattamento antidepressivo. Basandosi su precedenti studi condotti sempre dal GSRD, vennero scelte come variabili genetiche 12 SNPs, tra cui i polimorfismi rs6265, rs7430 e rs6313, e 8 parametri clinici, quali disturbo d'ansia, disturbo da attacchi di panico, malinconia e severità del DDM. Per valutare la resistenza o la risposta al trattamento venne usata l'HamD-17, definendo un paziente resistente se, a seguito di uno o più trattamenti durante l'ultimo episodio depressivo, presentava un punteggio superiore a 17 e responsivo se presentava un punteggio minore o uguale a 17 successivamente a uno o due trattamenti. Il modello utilizzato constava di due algoritmi: il primo era il random forest (RF), i cui compiti erano quelli di discriminare, partendo dalle 12 variabili genetiche e dalle 8 cliniche, i pazienti resistenti da quelli responsivi e, allo stesso tempo, valutare l'importanza, ovvero il valore predittivo, delle variabili di input, basandosi sulla accuratezza ottenuta dall'out-of-bag-dataset. Per allenare questo modello, tutti i partecipanti non dovevano avere nessuna variabile mancante; per questo motivo dei 1371 partecipanti iniziali ne vennero selezionati 225. Il secondo algoritmo era il k-means clustering, ovvero un algoritmo che, basandosi sui dati ricavati dall'RF, aveva il compito di suddividere i partecipanti in insiemi il più possibile omogenei in termini di risultato del trattamento. Così facendo, si ottennero cinque sottoinsiemi: di questi, quattro presentavano una media dei pazienti responsivi tra il 25.7% e il 34.1%; un gruppo invece presentava una media dei pazienti responsivi del 62.5%.

Concentrandosi sulle caratteristiche dei pazienti di questo ultimo gruppo, tutti i pazienti erano omozigoti per l'allele G dell'rs6265 e per l'allele T dell'rs6313 e la maggior parte dei pazienti erano omozigoti per l'allele G dell'rs7430 (62.5%) e non presentavano malinconia (75%). Ripescando dai 1371 partecipanti iniziali quei soggetti aventi queste caratteristiche, tra cui anche quelli inizialmente scartati perché aventi altri valori mancanti, il 62% di essi erano responsivi al loro trattamento antidepressivo. Infine, partendo sempre dai 1371 partecipanti iniziali, vennero selezionati 74 pazienti aventi informazioni disponibili per i quattro parametri rs6265, rs6313, rs7430 e malinconia, anche se con informazioni mancanti per alcuni dei restanti 16 dati. Questi 74 pazienti servirono per formare il test set, usato per verificare il modello RF precedentemente costruito. Per testare il modello RF si usarono tutte le possibili combinazioni di questi quattro parametri, mostrando come l'incremento del numero di fattori portasse ad un maggior tasso di pazienti responsivi correttamente classificati. Usando un solo fattore, la sensibilità era dello 0%, che arrivava al 21.9% quando usati due o tre fattori fino ad arrivare al 25% usando tutti e quattro i parametri. Questi risultati, seppur non proprio ottimali, mostrano comunque un effetto cumulativo delle variabili cliniche e genetiche nella predizione di pazienti responsivi o resistenti, fornendo indizi su quali possano essere i fattori, o la combinazione di fattori, che discriminano le due categorie di pazienti e quindi permettendo di agire direttamente su questi fattori. Al tempo stesso, lo studio aveva diversi limiti: alcuni partecipanti avevano ricevuto solo un trattamento antidepressivo, mentre altri due; i trattamenti a cui erano stati sottoposti i partecipanti erano di varia natura ed utilizzavano differenti farmaci, cosa che ovviamente può falsificare il risultato, dal momento che, un paziente può essere resistente ad un determinato trattamento, ma responsivo ad un altro. Inoltre, il numero di variabili studiate era limitato, avendone studiate solo 20 inizialmente e poi avendone approfondite solo quattro. Per questi motivi, al fine di approfondire

meglio l'impatto delle variabili cliniche e genetiche nei pazienti affetti da TRD, saranno necessari ulteriori studi.

3.2 L'IA per i Disturbi dello Spettro Autistico

Un altro ambito in cui l'IA si sta rivelando molto utile è quello dei disturbi dello spettro autistico (DSA). I DSA sono dei disturbi del neurosviluppo, ovvero del processo attraverso il quale si forma il sistema nervoso, caratterizzato da difficoltà nell'interazione e comunicazione sociale e dalla presenza di interessi ristretti e comportamenti ripetitivi e stereotipati. I DSA alterano la salute fisica e mentale dei bambini, gravando sulla loro interazione sociale, sull'apprendimento, sulla vita, sul lavoro, sulla famiglia e anche sulla società. Secondo l'OMS, la prevalenza dei DSA è in graduale aumento in tutto il mondo, con una prevalenza media globale del 0,62%, diventando sempre più un grave problema di salute pubblica globale, le cui cause sono ad oggi ancora non ben comprese, e quindi attraendo un grosso numero di ricercatori. La diagnosi precoce e, di conseguenza, l'intervento precoce per i bambini con DSA possono migliorare notevolmente la vita delle persone colpite; tuttavia, l'assenza di chiari ed identificabili biomarcatori relativi ai DSA rende la diagnosi difficoltosa. La procedura diagnostica seguita oggi consiste in osservazioni comportamentali eseguite da professionisti, supportate da strumenti quali "Autism Diagnostic Observation Schedule (ADOS)" e "Autism Diagnostic Interview-Revised (ADI-R)" dove diverse domande ed attività sono valutate dal professionista durante il processo di diagnosi. Nonostante i buoni livelli di validità e di accuratezza di questa procedura, vi sono alcuni aspetti negativi, dal momento che è una procedura che richiede molto tempo, con un numero molto elevato di domande e che richiede clinici specializzati per la gestione del processo. Questi limiti rendono necessario lo sviluppo di nuovi metodi che forniscano delle

valutazioni più rapide, accurate, oggettive, meno dispendiose e che al tempo stesso siano in grado di adattarsi all'eterogeneità dei pazienti affetti da DSA. In questo contesto, l'intelligenza artificiale si sta dimostrando un ottimo approccio: estraendo informazioni utili a partire da varie tipologie di dati, come dati genetici, comportamentali o derivanti da tecniche di neuroimaging, e costruendo modelli complessi che superano le prestazioni umane nell'analisi di grandi set di dati, i modelli di IA possono aiutare a capire quali di questi dati siano più influenti, permettendo una maggior comprensione dei DSA e, al tempo stesso, aiutando a costruire una base più solida ed oggettiva per screening e diagnosi migliori, permettendo di discriminare in maniera più efficiente e tempestiva i pazienti affetti da DSA da quelli sani.

Gli approcci con cui si è implementata l'IA nell'ambito dei DSA sono diversi; tra questi, vi sono, per esempio, l'utilizzo dell'IA al fianco degli attuali metodi di valutazione dei DSA, con l'obiettivo di coadiuvarli e migliorarli, e l'utilizzo dell'IA con lo scopo di investigare nuovi aspetti del comportamento da utilizzare negli screening o nelle diagnosi. I modi con cui l'IA può migliorare gli attuali metodi di valutazione dei DSA sono molteplici, come, per esempio, per migliorarne il potere predittivo diagnostico. Infatti, sebbene i sintomi dei DSA emergano presto nella vita, una diagnosi affidabile tipicamente non può essere fatta prima dei tre anni. I motivi alla base di ciò sono dati dal fatto che gli aspetti caratteristici dei DSA emergono più tardi e dalla grossa eterogeneità nell'eziologia, fenotipo, neurobiologia e percorso di sviluppo che caratterizzano i disturbi, facendo sì che ci siano più processi sottostanti che agendo insieme portano ai disturbi, piuttosto che un singolo processo, rendendone più complessa la diagnosi. Al tempo stesso, però, la prognosi dei DSA può essere migliorata agendo il più tempestivamente possibile, modificando i percorsi di sviluppo atipici. Dal momento che erogare un percorso di cura a tutti i bambini che presentano un sintomo che potrebbe essere riconducibile ai DSA è troppo costoso e poco efficace, si rendono necessari degli strumenti in grado di analizzare differenti tipi di dati

per far fronte alla grossa eterogeneità dei DSA. Il machine learning ha il potenziale di fornire un robusto algoritmo per la previsione dei risultati clinici, combinando informazioni complementari provenienti da diverse fonti in modo efficiente e consentendo l'identificazione della combinazione di dati più predittiva. Un esempio di ciò è dato dal lavoro svolto da Bussu et al. (Bussu et al., 2018), nel quale fu usato l'algoritmo SVM per inizialmente suddividere i bambini ad alto rischio¹¹ DSA (HR-DSA) da quelli ad alto rischio che poi però o non avrebbero sviluppato alcun disturbo (HR-Typical), o che avrebbero sviluppato un disturbo diverso dal DSA (HR-Atypical); successivamente, lo stesso algoritmo fu usato per suddividere gli HR-DSA e HR-Atypical da quelli HR-Typical. I dati di input all'algoritmo furono i punteggi MSEL, VABS e AOSI, valutati quando il bambino HR aveva sia 8 che 14 mesi, mentre lo scopo del modello SVM era prevedere l'eventuale sviluppo DSA o atipico al 36° mese, il cui gold standard era fornito dalla diagnosi di clinici esperti basandosi sui punteggi MSEL, VABS, ADOS e ADI-R. Il punteggio MSEL (Mullen Scales of Early Learning) è una misura standardizzata per valutare le funzioni cognitive verbali e non-verbali dalla nascita fino a 68 mesi; il VABS (Vineland Adaptive Behavior Scales) è una scala di valutazione del comportamento adattivo, che prevede una serie di domande da rivolgere ai genitori, in base alle quali viene definito il livello di sviluppo del bambino nella comunicazione, socializzazione, motricità e attività quotidiane; l'AOSI (Autism Observational Scale for Infants) è una misura impiegata per identificare i primi segni di autismo nei bambini ad alto rischio. Al fine di valutare quali parametri fossero più influenti, vennero creati più classificatori SVM, usando per ognuno diverse misure acquisite in momenti diversi (all'ottavo mese, al quattordicesimo o entrambi). Infine, per allenare e valutare il modello SVM venne usata la 40% hold-out cross validation, un particolare tipo di cross validation in cui si suddividono casualmente i dati nei due set di

¹¹ Per bambino ad alto rischio (high-risk infant) si intende un bambino che, perché nato con un peso troppo basso, o prematuramente, o con serie complicazioni mediche... è più incline a sviluppare ritardi nello sviluppo o una cognizione più lenta o dei deficit motori o una combinazione di essi.

training e validation, ripetuta 10 volte. Nel caso della distinzione tra HR-DSA e l'insieme dato da HR-Atypical e HR-Typical, considerando tutti i classificatori con i dati relativi all'ottavo mese, il valore AUC variava tra il 48% e il 65%; con i dati relativi al quattordicesimo mese, la prestazione migliore fu ottenuta usando il parametro "daily living" della scala VABS, ottenendo un AUC del 71.3%, una sensibilità del 79.6% e una specificità pari a 52.2%. Integrando i dati dell'8° e del 14° mese, le prestazioni mediamente non aumentarono significativamente. Nel caso della distinzione tra HR-Typical e l'insieme dato da HR-Atypical e HR-DSA, considerando tutti i classificatori con i dati relativi all'ottavo mese, il valore dell'AUC era compreso tra il 52% e il 69%; valori analoghi vennero ottenuti per i modelli usanti i dati relativi al quattordicesimo mese, mentre integrando i dati dei due diversi intervalli temporali i risultati furono leggermente migliori. Il miglior modello in questo caso fu quello relativo al quattordicesimo mese e che usava i dati provenienti dalle scale VABS e AOSI, con un'AUC del 71.8%, una sensibilità del 62.2% e una specificità del 69.5%. In definitiva, nello studio svolto da Bussu et al. vennero integrate informazioni provenienti da diversi domini e diversi intervalli temporali, con l'idea che così facendo si sarebbero migliorate le prestazioni in termini di previsione dei DSA rispetto alle prestazioni ottenute da un singolo dominio o intervallo temporale. Questo ragionamento fu solo in parte confermato, in particolar modo nel caso della distinzione tra HR-Typical e l'insieme dato da HR-Atypical e HR-DSA, dove l'integrazione delle informazioni proveniente da diversi intervalli temporali migliorò, anche se non eccessivamente, le prestazioni, cosa che non successe invece nella distinzione tra HR-DSA e il gruppo HR-Typical e HR-Atypical. Comunque, sebbene la previsione individuale dei DSA usando differenti misure provenienti da diversi istanti temporali fosse lontana dall'essere ottimale, questo studio ha mostrato come integrando diverse informazioni si sia in grado di discriminare uno sviluppo atipico o DSA da uno sviluppo "normale" al terzo anno di età già all'ottavo

e quattordicesimo mese, permettendo quindi di agire tempestivamente e, di conseguenza, più efficacemente.

Un altro modo con cui l'IA può aiutare gli attuali metodi di diagnosi dei DSA è riducendone i tempi. Infatti, sebbene l'autismo abbia una considerevole componente genetica, la sua valutazione viene tipicamente fatta mediante strumenti che analizzano il comportamento del bambino. Uno di questi strumenti è l'ADI-R, un'intervista comprensiva che raccoglie informazioni sui comportamenti importanti per la diagnosi di autismo, composta da 93 domande sulle abilità del soggetto nelle aree funzionali del linguaggio e comunicazione, interazioni sociali reciproche, comportamenti ed interessi ripetitivi, stereotipati e ristretti. Sebbene questo strumento sia molto affidabile, consistente tra i vari esaminatori (ovvero gode di poca variabilità inter-operatore) e fornisca un quadro chiaro dell'individuo affetto da DSA, la sua durata, che può arrivare ad essere anche di due ore e mezza, è proibitiva. Inoltre, nonostante molta attenzione sia stata data alla creazione di strumenti in grado di fornire una diagnosi più rapidamente, come l'"Autism Screening Questionnaire (ASQ)" o il "Modified Checklist for Autism in Toddlers (MCHAT)", molti di questi sono stati prevalentemente usati come strumenti per lo screening piuttosto che per la diagnosi, spesso seguiti dall'ADI-R o dall'ADOS. Per questo motivo si rende necessario un esame che preservi l'affidabilità dell'ADI-R ma che al tempo stesso possa essere compiuto nell'arco di minuti piuttosto che di ore, garantendo una più rapida diagnosi e quindi un'erogazione tempestiva e verosimilmente più incisiva della terapia. Una possibilità è quella di utilizzare algoritmi di machine learning per analizzare le 93 domande dell'ADI-R al fine di individuare quali di quest'ultime siano più significative per una corretta diagnosi e quindi usare solo quest'ultime come strumento diagnostico, riducendo il tempo necessario per lo svolgimento dell'esame. Questo tipo di approccio venne usato nella ricerca condotta da Wall et al. (Wall et al., 2012), dove, basandosi su 966 pazienti tra i 5 e i 17 anni, provenienti dal database dell'"Autism Genetic Resource Exchange (AGRE)" e già

sottoposti all'ADI-R, di cui 891 classificati con l'autismo e i restanti 75 senza autismo, si allenarono e testarono quindici algoritmi di ML alla ricerca delle domande con un maggior potere predittivo tra le 93 presenti nell'ADI-R. Ognuno di questi modelli venne allenato usando una 10-fold cross validation e, successivamente, testato usando un set indipendente di dati provenienti dai database del "Simons Simplex Collection (SSC)" e del "Boston Autism Consortium (AC)". Dei quindici modelli allenati e testati, il migliore fu l'"Alternating Decision Tree (ADTree)", ottenendo nel database AGRE una sensibilità del 100%, un tasso dei falsi positivi pari al 0.013 e un'accuratezza del 99.90%. Questo modello usava solo 7 delle 93 domande presenti nell'ADI-R, presentando quindi una riduzione del 93% circa. L'ADTree è un classificatore simile al decision tree, dove ogni domanda può andare ad aumentare o diminuire un punteggio, detto ADTree score; in questo studio, un punteggio negativo era relativo alla classificazione "autismo" mentre un punteggio positivo portava alla classificazione "sano". Il valore assoluto del punteggio era indice della confidenza della classificazione, con un punteggio assoluto maggiore indicativo di una maggiore confidenza. Nello studio svolto da Wall et al., la maggior parte degli output del modello furono il massimo del punteggio, o un valore vicino ad esso, per entrambi le classi, dimostrando come il classificatore fosse molto robusto (Fig. 3.2.1).

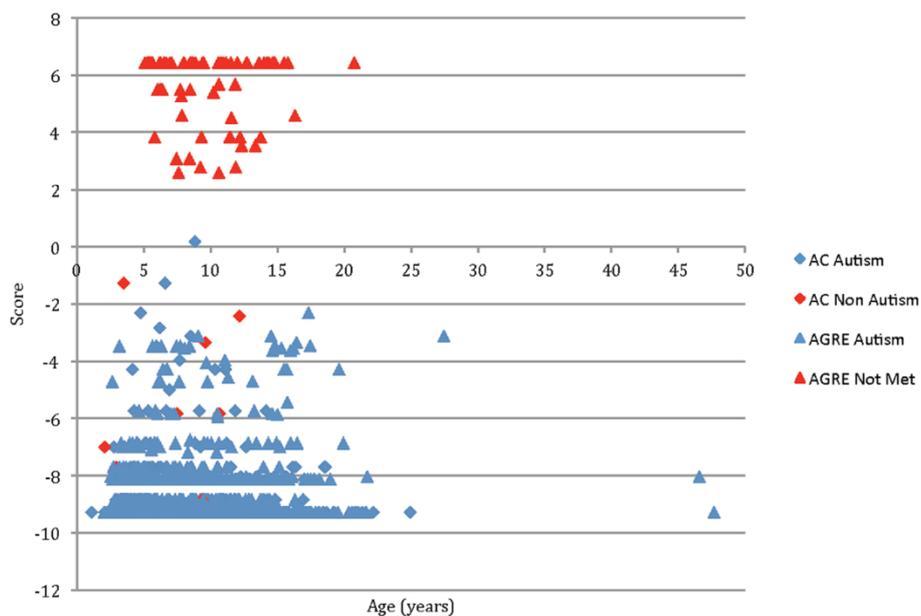


FIGURA 3.2.1: RISULTATI DELL'ADTREE. NOTARE COME LA MAGGIOR PARTE DEI VALORI SIANO, IN VALORE ASSOLUTO, ELEVATI (WALL ET AL., 2012).

Per validare in modo indipendente il modello ADTree vennero usati i punteggi ADI-R provenienti dai database SSC e AC. In entrambi i casi, il modello ottenne ottimi risultati: nel SSC, tutti i soggetti precedentemente diagnosticati con autismo vennero classificati correttamente; nell'AC, solo un soggetto dei 322 pazienti diagnosticati con autismo venne classificato erroneamente, anche se con un ADTree score davvero molto basso (0.179). Per quanto riguarda i soggetti diagnosticati senza autismo, il modello classificò correttamente 3 dei 5 soggetti nel SSC e 6 dei 12 nell'AC, anche se a tutti gli 8 pazienti erroneamente classificati dal modello venne in precedenza diagnosticato l'autismo usando l'ADOS, dimostrando sia come ulteriori esami vadano effettuati su questi pazienti, sia come il modello ADTree non avesse quindi completamente sbagliato a classificarli. Inoltre, dato il grosso sbilanciamento presente tra i soggetti di controllo (ovvero quelli sani) e quelli affetti da autismo, vennero simulati 2000 soggetti sani, 1000 dei quali simulati campionando casualmente le risposte ADI-R date da 92 dei soggetti sani, mentre i restanti 1000 creati mediante delle permutazioni casuali delle risposte ADI-R che in

generale non corrispondono alla diagnosi dell'autismo. Con riferimento al primo gruppo di 1000 soggetti, il modello ADTree ottenne una specificità del 99.9%, classificando in modo errato solo un soggetto; nel caso del secondo gruppo, il modello sbagliò a classificare solo 62 dei 1000 soggetti, raggiungendo una specificità pari al 93.8%. Infine, data l'importanza di diagnosticare l'autismo già dalla tenera età, si testò il modello ADTree anche con bambini al di sotto dei cinque anni di età. Anche in questo caso vennero usati 1589 soggetti diagnosticati con autismo e 88 soggetti senza autismo provenienti dai database AGRE e AC. Tutti i bambini con autismo tranne 1 vennero correttamente classificati dal modello, ottenendo una sensibilità quasi perfetta pari al 99.9%, mentre degli 88 soggetti sani solo 12 vennero classificati in modo sbagliato, corrispondendo ad una specificità dell'86%. Anche in questo caso, però, a tutti i 12 bambini venne diagnosticato l'autismo usando l'ADOS, per cui valgono le stesse considerazioni riportate precedentemente. In conclusione, gli ottimi risultati ottenuti dallo studio di Wall et al. mostrano come sia possibile ottenere uno strumento diagnostico che abbia un'affidabilità paragonabile all'ADI-R ma che, al tempo stesso, sia molto più rapido. La velocità di questo classificatore di ML e la sua ottima accuratezza per la classificazione di soggetti sia sani che affetti da DSA suggerisce la possibilità che possa essere utile nel rilevamento precoce e rapido dell'autismo, fornendo un valido strumento diagnostico che potrebbe richiedere una formazione meno specialistica del personale medico e che potrebbe andare incontro alle esigenze della popolazione a rischio, in particolare se erogato attraverso canali mobili, come internet e dispositivi portatili. Un lavoro analogo allo studio svolto da Wall et al. è quello di Kosmicki et al. (Kosmicki et al., 2015), focalizzato però sull'ADOS. Così come l'ADI-R, l'ADOS è uno degli strumenti diagnostici più usati per l'autismo, composto da quattro moduli, ognuno dei quali specifico per un diverso livello di sviluppo del bambino per garantire la copertura di diverse manifestazioni comportamentali. Ogni modulo è composto da una serie di attività semi-strutturate in

grado di elicitare specifici comportamenti relativi al linguaggio e comunicazione, interazione sociale, immaginazione e creatività, comportamenti stereotipati ed interessi ristretti. Il problema, ancora una volta, è che ogni esame richiede diverso tempo, tipicamente dai 30 ai 60 minuti fino ad arrivare ai 90 minuti; se a ciò si aggiunge il fatto che un esame possa dover essere ripetuto più volte da diversi professionisti per ridurre al minimo la variabilità inter-operatore e che negli ultimi anni la percentuale di bambini affetti da DSA sta aumentando, il conseguente collo di bottiglia nel sistema sanitario è inevitabile. Il lavoro svolto da Kosmicki et al. fu quindi finalizzato ad individuare, all'interno delle varie attività dei moduli 2 e 3 dell'ADOS e mediante l'uso di modelli di ML, l'insieme dato dal minor numero di attività in grado di fornire delle prestazioni comunque accettabili ma con tempi ridotti. Per far ciò, il gruppo di lavoro utilizzò i dati del modulo 2 e 3 di soggetti provenienti da cinque diversi database, quali l'AC, l'SSC e l'AGRE, il "National Database of Autism Research (NDAR)" e il "Simons Variation in Individuals Project (SVIP)"; sulla base di questi dati, per allenare i modelli di ML ogni soggetto venne classificato come "sano" o affetto da DSA mediante la diagnosi fornita dall'algoritmo dell'ADOS. Per ognuno dei due moduli vennero sviluppati otto algoritmi di ML, tra cui ADTree, Naive Bayes, logistic regression e random forest, ciascuno dei quali allenato mediante la "stepwise backward feature selection" su uno dei quattro database (il NDAR per gli algoritmi di ML del modulo 2 e l'AGRE per il modulo 3) e testato sugli altri quattro database indipendenti. In particolar modo, ogni algoritmo di ML veniva inizialmente allenato usando 28 caratteristiche elicitate tramite le attività dei moduli 2/3 mediante la 10-fold cross-validation; durante ciò, ognuna delle 28 caratteristiche dei moduli 2 e 3 veniva ordinata usando l'algoritmo SVM in base alla sua abilità di discriminare i soggetti con DSA da quelli senza. Una volta ordinate le caratteristiche, si rimuoveva quella in ultima posizione e si ripeteva il procedimento, usando quindi di volta in volta come input agli algoritmi sempre meno caratteristiche, finché non si arrivava ad allenare e testare gli

algoritmi usando come input una sola caratteristica. In questo modo si ottennero 28 classificatori per ognuno dei modelli di ML impiegati, ciascuno con le proprie prestazioni. Mappando per ognuno di questi classificatori la propria sensibilità, specificità ed accuratezza in relazione al numero di caratteristiche usate, il migliore venne individuato come quello con le miglior prestazioni e il minor numero di caratteristiche impiegate. Così facendo, per il modulo 2 i migliori algoritmi furono la regressione logistica e il “logistic model tree (LMT)”, ovvero un algoritmo supervisionato che incorpora al suo interno gli alberi decisionali e la regressione logistica, entrambi usanti le stesse nove caratteristiche e con prestazioni molto simili, anche se per i successivi test venne usata solo la regressione logistica. Testandola sui quattro dataset indipendenti (AC, AGRE, SSC e SVIP), la regressione logistica sbagliò a classificare solo 13 dei 1089 soggetti affetti da DSA, ottenendo una sensibilità del 98.81%, e 7 dei 66 soggetti senza DSA, raggiungendo una specificità del 89.39% e in totale un’accuratezza del 98.27%. Così come successe nel lavoro svolto da Wall et al., anche in questo caso l’errata classificazione della maggior parte dei soggetti non era in realtà completamente sbagliata: per esempio, sei soggetti erroneamente classificati con l’autismo erano stati in precedenza diagnosticati (ma non usando l’algoritmo dell’ADOS) con l’autismo, e tre degli individui erroneamente classificati senza l’autismo erano stati in precedenza diagnosticati senza l’autismo. Nel modulo 3, il miglior algoritmo fu il radial kernel SVM con dodici caratteristiche. Questo algoritmo venne testato nei restanti quattro database non utilizzati durante l’allenamento (ovvero AC, NDAR, SSC e SVIP), classificando erroneamente 44 dei 1924 pazienti con DSA e 6 dei 214 individui senza DSA, ottenendo quindi una sensibilità del 97.71%, una specificità del 97.20% e un’accuratezza pari al 97.66%. Questi risultati dimostrano come non sia necessario, durante un esame diagnostico implementato mediante l’utilizzo dell’ADOS, che il medico usi tutte le 14 attività dei moduli 2 e 3; bensì, egli si può concentrare solo su quelle attività in grado di elicitarle

caratteristiche più significative individuate dallo studio di Kosmicki et al. Ciò si tradurrebbe, come già visto con la ricerca di Wall et al., in una riduzione dei tempi necessari per la diagnosi, pur mantenendo elevate prestazioni quali sensibilità, specificità ed accuratezza, ed incoraggerebbe lo sviluppo di strumenti basati sullo screening per il rilevamento dei DSA e approcci di salute mobile che consentirebbero alle persone di ricevere cure più convenienti di quanto sia possibile con i metodi attuali.

Come già citato ad inizio paragrafo, l'IA, oltre a poter servire per migliorare gli attuali strumenti diagnostici, può aiutare anche ad investigare nuovi aspetti del comportamento che siano caratteristici dei soggetti affetti da DSA, e che quindi possono essere utili come valide informazioni per discriminare i soggetti sani da quelli affetti da DSA. Un esempio di ciò è dato dal lavoro svolto da Crippa et al. (Crippa et al., 2015), finalizzato a determinare se un semplice movimento degli arti superiori potesse essere d'aiuto per classificare accuratamente i bambini, di età compresa tra i due e i quattro anni, affetti da DSA. Infatti, sebbene le caratteristiche principali dei DSA siano dei deficit nella comunicazione ed interazione sociali e dei comportamenti, attività ed interessi ristretti e ripetitivi, è comunque importante non sottovalutare i deficit motori associati ai DSA, dal momento che essi hanno un'alta prevalenza del 79%, possono presentarsi molto presto durante lo sviluppo e possono avere un grosso impatto sulla qualità di vita del bambino. Per valutare quindi se un'attività motoria potesse essere d'aiuto nel classificare i bambini affetti da DSA, Crippa et al. svilupparono un modello di ML supervisionato per identificare i bambini, dai due ai quattro anni di età, affetti da DSA e discriminarli dai bambini sani, mediante l'analisi cinematica di un semplice gesto di raggiungere, afferrare e rilasciare un oggetto. Nello studio vennero coinvolti quindici bambini sani e quindici bambini a cui venne precedentemente diagnosticato un DSA, prima da parte di un medico, in accordo con i criteri del DSM, e successivamente confermato in modo indipendente da uno psicologo. I bambini sani vennero scelti in modo tale che la loro

età mentale¹² corrispondesse a quella dei bambini affetti da DSA, in modo tale che entrambi i gruppi di bambini avessero avuto la stessa abilità nel comprendere le istruzioni del compito, usare strategie appropriate ed inibire comportamenti non appropriati. Il compito dei bambini era quello di raggiungere con il braccio una pallina posizionata su un supporto, afferrarla e rilasciarla in una scatola. Tutto il movimento venne catturato mediante un sistema optoelettronico e l'utilizzo di marker passivi posizionati in punti strategici del braccio del bambino e sulla pallina. A partire da queste registrazioni vennero successivamente estratti i 17 parametri cinematici di input al modello di ML, tra cui la durata del movimento, la velocità massima, l'accelerazione massima e la decelerazione massima. L'algoritmo di ML constava di due parti: la prima parte era relativa alla selezione delle caratteristiche, ovvero il processo in cui veniva usata il "Fisher discriminant ratio (FDR)" per selezionare un sottoinsieme di caratteristiche rilevanti da usare per la classificazione; nella seconda parte, usando come input le caratteristiche precedentemente selezionate, venne usato un algoritmo SVM per discriminare i pazienti affetti da DSA da quelli non affetti, allenato e testato usando la leave-one-out cross validation. I risultati migliori vennero ottenuti usando sette caratteristiche selezionate dal FDR, raggiungendo un'accuratezza del 96.7%, una specificità del 93.8% e una sensibilità del 100%. Questi ottimi risultati dimostrano come sia fattibile ed affidabile usare un modello SVM al fine di classificare correttamente i bambini con DSA usando un semplice compito motorio. Ciò è di particolare rilevanza soprattutto perché, come già fatto notare in precedenza, la diagnosi dei DSA nei bambini piccoli è alquanto difficile, pure usando gli strumenti standard come l'ADOS o l'ADI-R; l'utilizzo di compiti motori, associati a modelli di ML, può quindi avere un valido potenziale clinico in questi casi, fornendo informazioni utili ai medici a supporto della diagnosi. Un altro aspetto importante dello studio svolto da Crippa et al. è l'aver

¹² L'età mentale è un concetto legato all'intelligenza, e rappresenta la capacità mentale di una persona espressa come l'età in cui una persona media raggiunge la stessa capacità.

utilizzato, a fini predittivi, un semplice compito motorio, che quindi può essere facilmente valutato e può essere compreso e svolto da quasi tutti i bambini, data la sua facilità e la sua natura auto esplicativa. In aggiunta a ciò, l'utilizzo di un'analisi cinematica fornisce uno strumento diagnostico privo di vincoli e non invadente, che sono aspetti molto importanti quando si deve andare a valutare dei pazienti "difficili", come appunto i bambini con DSA. Infine, l'analisi cinematica è anche una tecnologia più conveniente e meno costosa rispetto, per esempio, all'MRI, essendo inoltre più facilmente amministrabile da qualsiasi professionista che lavori con i bambini e di durata ridotta (circa 15 minuti per lo svolgimento del compito e 30 minuti per l'analisi dei dati acquisiti). Tuttavia, nonostante tutti questi aspetti positivi, lo studio di Crippa et al. ha un grosso limite, dovuto al numero esiguo di partecipanti (30 in totale), rendendo necessari quindi ulteriori studi al fine di confermare o meno i risultati ottenuti da Crippa et al.

Conclusioni

Grazie ad una quantità sempre più grande di dati ed una potenza di calcolo in continua crescita, l'intelligenza artificiale sta compiendo passi da gigante, con il miglioramento e lo sviluppo di algoritmi sempre più sofisticati ed accurati, e di conseguenza venendo implementata in un vasto numero di ambiti, tra cui quello delle neuroscienze. All'interno delle neuroscienze, data la loro vastità, l'utilizzo dell'IA è estremamente variegato e, in questa tesi, sono stati illustrati solo alcuni esempi di quale sia il ruolo dell'IA nelle neuroscienze. Tra questi, è stata data maggior importanza all'utilizzo dell'IA nel campo della salute mentale, approfondendo in particolar modo gli aspetti patologici del DDM e del DSA.

Nel DDM uno dei maggiori problemi che ci si ritrova ad affrontare al giorno d'oggi è l'erogazione della giusta terapia al giusto paziente, cercando di evitare il più possibile l'approccio "trial-and-error". L'IA si è rivelata essere un ottimo strumento che, grazie all'analisi di diverse tipologie di dati, come dati clinici, demografici e/o derivanti da registrazioni EEG, è in grado, con un'elevata accuratezza, di prevedere l'efficacia di uno o più trattamenti per ogni singolo paziente. Ciò permette di evitare l'approccio trial-and-error, erogando fin da subito un farmaco efficace e quindi facendo risparmiare tempo e denaro sia al sistema sanitario che al paziente, con ovviamente anche effetti positivi sulla salute del paziente stesso.

Mentre nel DDM ci si è focalizzati principalmente sul lato terapeutico della patologia, analizzando l'utilizzo dell'IA come strumento per prevedere l'efficacia di un trattamento, nei DSA ci si è invece concentrati sull'aspetto della diagnosi della patologia, indagando la fattibilità di implementare l'IA come strumento diagnostico. Infatti, sebbene i DSA abbiano un'importante componente genetica, ad oggi la sua diagnosi viene principalmente fatta mediante l'analisi comportamentale, utilizzando strumenti come l'ADOS o l'ADI-R che, nonostante l'ottima affidabilità,

hanno diversi aspetti negativi, come l'eccessivo tempo necessario per l'esame diagnostico. Anche in questo caso, l'utilizzo dell'IA si è rivelato un ottimo approccio, in grado sia di ridurre la durata di questi esami, estraendo dall'ADOS e dall'ADI-R solo quegli aspetti con maggior poter predittivo, ovvero più importanti ai fini diagnostici, sia permettendo di prevedere fin dalla tenera età l'insorgenza o meno dei DSA, permettendo di agire in maniera più tempestiva e quindi più efficiente, ma anche indagando nuovi aspetti dei DSA che ne facilitino la diagnosi.

Complessivamente, l'IA fornisce un approccio alternativo più che valido per la risoluzione di molti problemi attuali nell'ambito delle neuroscienze, riducendo le risorse necessarie, in termini di tempo, soldi e anche di personale, e fornendo risultati oggettivi, replicabili e molto accurati.

Bibliografia

- Abedi, V., Goyal, N., Tsivgoulis, G., Hosseinichimeh, N., Hontecillas, R., Bassaganya-Riera, J., Elijovich, L., Metter, J. E., Alexandrov, A. W., Liebeskind, D. S., Alexandrov, A. V., & Zand, R. (2017). Novel Screening Tool for Stroke Using Artificial Neural Network. *Stroke*, *48*(6), 1678–1681. <https://doi.org/10.1161/STROKEAHA.117.017033>
- Al-Kaysi, A. M., Al-Ani, A., Loo, C. K., Powell, T. Y., Martin, D. M., Breakspear, M., & Boonstra, T. W. (2017). Predicting tDCS treatment outcomes of patients with major depressive disorder using automated EEG classification. *Journal of Affective Disorders*, *208*, 597–603. <https://doi.org/10.1016/j.jad.2016.10.021>
- Anumanchipalli, G. K., Chartier, J., & Chang, E. F. (2019). Speech synthesis from neural decoding of spoken sentences. *Nature*, *568*(7753), 493–498. <https://doi.org/10.1038/s41586-019-1119-1>
- Bruder, G. E., Stewart, J. W., Tenke, C. E., McGrath, P. J., Leite, P., Bhattacharya, N., & Quitkin, F. M. (2001). Electroencephalographic and perceptual asymmetry differences between responders and nonresponders to an SSRI antidepressant. *Biological Psychiatry*, *49*(5), 416–425. [https://doi.org/10.1016/S0006-3223\(00\)01016-7](https://doi.org/10.1016/S0006-3223(00)01016-7)
- Bussu, G., Jones, E. J. H., Charman, T., Johnson, M. H., Buitelaar, J. K., Baron-Cohen, S., Bedford, R., Bolton, P., Blasi, A., Chandler, S., Charman, T., Cheung, C., Davies, K., Elsabbagh, M., Fernandes, J., Gammer, I., Garwood, H., Gliga, T., Guiraud, J., ... the BASIS Team. (2018). Prediction of Autism at 3 Years from Behavioural and Developmental Measures in High-Risk Infants: A Longitudinal Cross-Domain Classifier Analysis. *Journal of Autism and Developmental Disorders*, *48*(7), 2418–2433. <https://doi.org/10.1007/s10803-018-3509-x>

- Coen, P., Xie, M., Clemens, J., & Murthy, M. (2016). Sensorimotor Transformations Underlying Variability in Song Intensity during *Drosophila* Courtship. *Neuron*, 89(3), 629–644.
<https://doi.org/10.1016/j.neuron.2015.12.035>
- Crippa, A., Salvatore, C., Perego, P., Forti, S., Nobile, M., Molteni, M., & Castiglioni, I. (2015). Use of Machine Learning to Identify Children with Autism and Their Motor Abnormalities. *Journal of Autism and Developmental Disorders*, 45(7), 2146–2156.
<https://doi.org/10.1007/s10803-015-2379-8>
- Deshpande, G., Wang, P., Rangaprakash, D., & Wilamowski, B. (2015). Fully Connected Cascade Artificial Neural Network Architecture for Attention Deficit Hyperactivity Disorder Classification From Functional Magnetic Resonance Imaging Data. *IEEE Transactions on Cybernetics*, 45(12), 2668–2679. <https://doi.org/10.1109/TCYB.2014.2379621>
- Fang, J., Li, Y., Liu, R., Pang, X., Li, C., Yang, R., He, Y., Lian, W., Liu, A.-L., & Du, G.-H. (2015). Discovery of multitarget-directed ligands against Alzheimer’s disease through systematic prediction of chemical-protein interactions. *Journal of Chemical Information and Modeling*, 55(1), 149–164. <https://doi.org/10.1021/ci500574n>
- Géron, A. Hands-On Machine Learning with Scikit-Learn and TensorFlow. Second edition
- Graham, S., Depp, C., Lee, E. E., Nebeker, C., Tu, X., Kim, H.-C., & Jeste, D. V. (2019). Artificial Intelligence for Mental Health and Mental Illnesses: An Overview. *Current Psychiatry Reports*, 21(11), 116. <https://doi.org/10.1007/s11920-019-1094-0>
- Isensee, F., Jaeger, P. F., Full, P. M., Vollmuth, P., & Maier-Hein, K. H. (2020). *NnU-Net for Brain Tumor Segmentation* (arXiv:2011.00848). arXiv. <https://doi.org/10.48550/arXiv.2011.00848>
- Isensee, F., Kickingereder, P., Wick, W., Bendszus, M., & Maier-Hein, K. H. (2019). *No New-Net* (arXiv:1809.10483). arXiv. <http://arxiv.org/abs/1809.10483>
- Kautzky, A., Baldinger, P., Souery, D., Montgomery, S., Mendlewicz, J., Zohar, J., Serretti, A., Lanzenberger, R., & Kasper, S. (2015). The combined effect of genetic polymorphisms and

clinical parameters on treatment outcome in treatment-resistant depression. *European Neuropsychopharmacology*, 25(4), 441–453.

<https://doi.org/10.1016/j.euroneuro.2015.01.001>

Kell, A. J. E., Yamins, D. L. K., Shook, E. N., Norman-Haignere, S. V., & McDermott, J. H.

(2018). A Task-Optimized Neural Network Replicates Human Auditory Behavior, Predicts Brain Responses, and Reveals a Cortical Processing Hierarchy. *Neuron*, 98(3), 630-644.e16.

<https://doi.org/10.1016/j.neuron.2018.03.044>

Kheradpisheh, S. R., Ghodrati, M., Ganjtabesh, M., & Masquelier, T. (2016). Humans and Deep Networks Largely Agree on Which Kinds of Variation Make Object Recognition Harder.

Frontiers in Computational Neuroscience, 10, 92.

<https://doi.org/10.3389/fncom.2016.00092>

Khodayari-Rostamabad, A., Reilly, J. P., Hasey, G. M., de Bruin, H., & MacCrimmon, D. J. (2013).

A machine learning approach using EEG data to predict response to SSRI treatment for major depressive disorder. *Clinical Neurophysiology*, 124(10), 1975–1985.

<https://doi.org/10.1016/j.clinph.2013.04.010>

Kim, J., Calhoun, V. D., Shim, E., & Lee, J.-H. (2016). Deep neural network with weight sparsity control and pre-training extracts hierarchical features and enhances classification performance: Evidence from whole-brain resting-state functional connectivity patterns of schizophrenia. *NeuroImage*, 124, 127–146.

<https://doi.org/10.1016/j.neuroimage.2015.05.018>

Koesmahargyo, V., Abbas, A., Zhang, L., Guan, L., Feng, S., Yadav, V., & Galatzer-Levy, I. R.

(2020). Accuracy of machine learning-based prediction of medication adherence in clinical research. *Psychiatry Research*, 294, 113558. <https://doi.org/10.1016/j.psychres.2020.113558>

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The*

Bulletin of Mathematical Biophysics, 5(4), 115–133. <https://doi.org/10.1007/BF02478259>

- McIntosh, L. T., Maheswaranathan, N., Nayebi, A., Ganguli, S., & Baccus, S. A. (2016). Deep Learning Models of the Retinal Response to Natural Scenes. *Advances in neural information processing systems*, 29, 1369–1377.
- Minsky, M., & Papert, S. A. (2017). *Perceptrons, Reissue of the 1988 Expanded Edition with a new foreword by Léon Bottou: An Introduction to Computational Geometry*. MIT Press.
- Nakhawa, A. (2021, marzo 19). A General Overview of Artificial Intelligence. *MLearning.Ai*.
<https://medium.com/mllearning-ai/a-general-overview-of-artificial-intelligence-85437b4eabd8>
- Nomidl. (2022, aprile 20). What is ReLU and Sigmoid activation function? *Nomidl*.
<https://www.nomidl.com/deep-learning/what-is-relu-and-sigmoid-activation-function/>
- Segura-Bedmar, I., & Raez, P. (2019). Cohort selection for clinical trials using deep learning models. *Journal of the American Medical Informatics Association: JAMIA*, 26(11), 1181–1188. <https://doi.org/10.1093/jamia/ocz139>
- Spasov, S., Passamonti, L., Duggento, A., Liò, P., Toschi, N., & Alzheimer’s Disease Neuroimaging Initiative. (2019). A parameter-efficient deep learning approach to predict conversion from mild cognitive impairment to Alzheimer’s disease. *NeuroImage*, 189, 276–287. <https://doi.org/10.1016/j.neuroimage.2019.01.031>
- Sussillo, D., Churchland, M. M., Kaufman, M. T., & Shenoy, K. V. (2015). A neural network that finds a naturalistic solution for the production of muscle activity. *Nature Neuroscience*, 18(7), 1025–1033. <https://doi.org/10.1038/nn.4042>
- Talathi, S. S. (2017). Deep Recurrent Neural Networks for seizure detection and early seizure detection systems. *arXiv:1706.03283 [cs, q-bio]*. <http://arxiv.org/abs/1706.03283>
- The roles of supervised machine learning in systems neuroscience | Elsevier Enhanced Reader*. (s.d.). <https://doi.org/10.1016/j.pneurobio.2019.01.008>

Turing, A. M. (1937). On Computable Numbers, with an Application to the Entscheidungsproblem.

Proceedings of the London Mathematical Society, s2-42(1), 230–265.

<https://doi.org/10.1112/plms/s2-42.1.230>

Turing, A. M. (2009). Computing Machinery and Intelligence. In R. Epstein, G. Roberts, & G.

Beber (A c. Di), *Parsing the Turing Test: Philosophical and Methodological Issues in the Quest for the Thinking Computer* (pagg. 23–65). Springer Netherlands.

https://doi.org/10.1007/978-1-4020-6710-5_3

Wall, D. P., Dally, R., Luyster, R., Jung, J.-Y., & DeLuca, T. F. (2012). Use of Artificial

Intelligence to Shorten the Behavioral Diagnosis of Autism. *PLOS ONE*, 7(8), e43855.

<https://doi.org/10.1371/journal.pone.0043855>

Sitografia

1.9. *Naive Bayes*. (s.d.). Scikit-Learn. Recuperato 9 maggio 2022, da

https://scikit-learn/stable/modules/naive_bayes.html

Aiello, L. C. (2004, giugno) *INTELLIGENZA ARTIFICIALE: I PRIMI 50 ANNI* (s.d.) Recuperato 4 maggio

2022, da <http://web.dfc.unibo.it/buzzetti/IUcorso2007-08/mdidattici/ajello.pdf>

Brownlee, J. (2019, aprile 16). How Do Convolutional Layers Work in Deep Learning Neural Networks?

Machine Learning Mastery. Recuperato 10 maggio 2022, da

<https://machinelearningmastery.com/convolutional-layers-for-deep-learning-neural-networks/>

Cannata, G. P. (2020, settembre 30). Vanishing gradient in Deep Neural Networks. *Medium*.

Recuperato 10 maggio 2022, da <https://cannydatascience.medium.com/vanishing-gradient-in-deep-neural-networks-24d887047bf4>

Che cos'è il Reinforcement Learning? (s.d.). Recuperato 5 maggio 2022, da

<https://it.mathworks.com/discovery/reinforcement-learning.html>

CPU e GPU: Qual è la differenza? (s.d.). Intel. Recuperato 4 maggio 2022, da

<https://www.intel.com/content/www/it/it/products/docs/processors/cpu-vs-gpu.html>

Decision Tree—Algoritmi di Machine Learning. (2021, febbraio 10). *Pulp Learning - Tutto sul Machine*

Learning. Recuperato 6 maggio, da <https://pulplearning.altervista.org/algoritmi-di-machine-learning-decision-tree/>

Epilessia. (s.d.). *Humanitas*. Recuperato 15 maggio 2022, da

<https://www.humanitas.it/malattie/epilessia/>

Glioma. (2021). In *Wikipedia*. Recuperato 15 maggio 2022, da

<https://it.wikipedia.org/w/index.php?title=Glioma&oldid=120892945>

Glioma e Glioblastoma. (2019, agosto 7). *Glioblastoma Multiforme*. Recuperato 15 maggio 2022, da

<https://www.glioblastomamultiforme.it/glioma-e-glioblastoma/>

Guida rapida alle funzioni di attivazione nel Deep Learning – NetAi. (s.d.). Recuperato 7 maggio 2022, da

<https://netai.it/guida-rapida-alle-funzioni-di-attivazione-nel-deep-learning/>

History of neuroscience. (2022). In *Wikipedia*. Recuperato 12 maggio 2022, da

https://en.wikipedia.org/w/index.php?title=History_of_neuroscience&oldid=1080817674

L'algoritmo Naive Bayes, cos'è e come funziona (machine learning)—Andrea Minini. (s.d.). Recuperato

7 maggio 2022, da <https://www.andreaminini.com/ai/machine-learning/algoritmo-naive-bayes>

La grande scienza. Intelligenza artificiale in «Storia della Scienza». (s.d.). Recuperato 4 maggio 2022, da

[https://www.treccani.it/enciclopedia/la-grande-scienza-intelligenza-artificiale_\(Storia-della-Scienza\)](https://www.treccani.it/enciclopedia/la-grande-scienza-intelligenza-artificiale_(Storia-della-Scienza))

Lavecchia, V. (2019, giugno 11). Caratteristiche e funzionamento del Deep Learning in informatica.

Informatica e Ingegneria Online. Recuperato 5 maggio 2022, da

<https://vitolavecchia.altervista.org/caratteristiche-e-funzionamento-del-deep-learning-in-informatica/>

LE NEUROSCIENZE: PER INIZIARE. (s.d.). Recuperato 12 maggio 2022, da

<https://www.sinapsyche.it/neuroscienze/>

Logistic function. (2022). In *Wikipedia*. Recuperato 8 maggio 2022, da

https://en.wikipedia.org/w/index.php?title=Logistic_function&oldid=1094100764

Macchine a vettori di supporto. (2022). In *Wikipedia*. Recuperato 8 maggio 2022, da

https://it.wikipedia.org/w/index.php?title=Macchine_a_vettori_di_supporto&oldid=125370033

Machine Learning- Confusion Matrix | i2tutorials. (s.d.). Recuperato 10 maggio 2022, da

<https://www.i2tutorials.com/machine-learning-tutorial/machine-learning-confusion-matrix/>

Maltoni, D. (s.d.). *Tipi di Learning Supervisionato, Non supervisionato Batch, Incrementale, Naturale Reinforcement Learning*. 38. Recuperato 5 maggio 2022

Multimodal Brain Tumor Segmentation Challenge 2019 | CBICA | Perelman School of Medicine at the University of Pennsylvania. (s.d.). Recuperato 15 maggio 2022, da <https://www.med.upenn.edu/cbica/brats-2019/>

Neural networks and deep learning. (s.d.). Recuperato 9 maggio 2022, da <http://neuralnetworksanddeeplearning.com/chap1.html#eqtn3>

Neuroscienze. (2021). In *Wikipedia*. Recuperato 12 maggio 2022, da <https://it.wikipedia.org/w/index.php?title=Neuroscienze&oldid=120624032>

Neuroscienze: Dalle principali branche alle attuali tecniche di neuroimaging. (s.d.). Recuperato 13 maggio 2022, da <https://www.stateofmind.it/neuroscienze/>

Neuroscienze nell'Enciclopedia Treccani. (s.d.). Recuperato 12 maggio 2022, da <https://www.treccani.it/enciclopedia/neuroscienze>

Receiver operating characteristic. (2022). In *Wikipedia*. Recuperato 11 maggio 2022, da https://en.wikipedia.org/w/index.php?title=Receiver_operating_characteristic&oldid=1094148194

Reinforcement learning. (2022). In *Wikipedia*. Recuperato 5 maggio 2022, da https://en.wikipedia.org/w/index.php?title=Reinforcement_learning&oldid=1095185224

Relazione quantitativa struttura-attività. (2019). In *Wikipedia*. Recuperato 13 maggio 2022, da https://it.wikipedia.org/w/index.php?title=Relazione_quantitativa_struttura-attivit%C3%A0&oldid=104578423

Rete neurale ricorrente. (2022). In *Wikipedia*. Recuperato 9 maggio 2022, da https://it.wikipedia.org/w/index.php?title=Rete_neurale_ricorrente&oldid=126465408

StatQuest with Josh Starmer. (2018, marzo 5). *StatQuest: Logistic Regression*. Recuperato 7 maggio 2022, da <https://www.youtube.com/watch?v=yIYKR4sgzI8>

StatQuest with Josh Starmer. (2021, aprile 26). *Decision and Classification Trees, Clearly Explained!!!*
Recuperato 6 maggio 2022, da <https://www.youtube.com/watch?v=L39rN6gz7Y>

Supervised Learning. In machine learning, Supervised... | by Jorge Leonel | Medium. (s.d.). Recuperato 5 maggio 2022, da <https://medium.com/@jorgesleonel/supervised-learning-c16823b00c13>

Touger, E. (2018, agosto 3). What's the Difference Between Artificial Intelligence (AI), Machine Learning, and Deep Learning? *Prowess Consulting*. Recuperato 4 maggio 2022, da <https://www.prowesscorp.com/whats-the-difference-between-artificial-intelligence-ai-machine-learning-and-deep-learning/>

User:Yearofthedragon. (s.d.). *English: Artificial neuron (Computer science)*. Own work. Recuperato 10 maggio 2022, da <https://commons.wikimedia.org/wiki/File:Computer.Science.AI.Neuron.svg>

Virtual screening. (2022). In *Wikipedia*. Recuperato 15 maggio 2022, da https://en.wikipedia.org/w/index.php?title=Virtual_screening&oldid=1079914803

What is Supervised Learning? | IBM. (s.d.). Recuperato 6 maggio 2022, da <https://www.ibm.com/cloud/learn/supervised-learning>

What is the difference between Extra Trees and Random Forest? | Quantdare. (s.d.). Recuperato 6 maggio 2022, da <https://quantdare.com/what-is-the-difference-between-extra-trees-and-random-forest/>

Wikipedia, B. at E. (2006). *English: The picture was created by myself during my PhD for evaluating three different HIV epitope predictors. The methods were tested on only a small protein sample, so the results in this picture should not be used to evaluate the strength of the predictors. There're enough papers online (pubmed) where they're compared using large datasets*. Transferred from

en.wikipedia to Commons. Recuperato 9 maggio 2022, da

<https://commons.wikimedia.org/wiki/File:Roccurves.png>