Politecnico di Torino

Metodi di machine learning per la refertazione automatica di EEG in pazienti con gravi cerebrolesioni acquisite

Relatore:
Filippo Molinari

Correlatori:
Piergiuseppe Liuzzi
Andrea Mannini

Candidato:
Giorgio Chiesa

-

Dedicata a:

Abstract

Negli ultimi anni, i miglioramenti nel trattamento dei pazienti con grave cerebrolesione acquisita (GCA) hanno notevolmente aumentato le loro possibilità di sopravvivenza. Tra questi, un numero cospicuo di pazienti può sviluppare un disturbo della coscienza (DoC). La Coma Recovery Scale-Revised (CRS-R) è stata identificata come punto di riferimento standard per la valutazione clinica della coscienza in questi pazienti. Tuttavia, un'alta variabilità inter-operatore, fluttuazioni circadiane dei livelli di vigilanza e problemi di comunicazione sono notoriamente associati a errori nella diagnosi. L'analisi strumentale, ad esempio mediante elettroencefalografia costituisce una possibile strada per porre rimedio a questo problema. Hirsch e colleghi hanno proposto una terminologia standard per la refertazione di elettroencefalogrammi (EEG) durante la Critical Care. Ciononostante, la refertazione dei tracciati secondo tale terminologia richiede un'analisi attenta e accurata di una registrazione EEG di almeno 15 minuti. Questo lavoro di tesi si propone di studiare, sviluppare e validare un software per la refertazione automatica dei descrittori di Hirsch (frequenza, voltaggio, simmetria, gradiente antero-posteriore, anomalie lente delta, grafo-elementi epilettici) basato su metodi di machine learning.

A partire da un dataset contenente 621 registrazioni EEG di pazienti con GCA, sono state estratte features nel dominio del tempo e della frequenza. Tramite quest'ultime, 9 tipologie di modelli sono state allenate, validate e testate con un approccio di cross-validazione annidata. Sono state applicate quando necessario tecniche di i) ricampionamento, ii) selezione di features, iii) riduzione della dimensionalità e la soluzione più performante per ogni descrittore ha raggiunto un'accuratezza sul test superiore all'80%. È stata quindi realizzata un'interfaccia grafica per testare nuovi pazienti utilizzando i migliori modelli ottenuti. Questa consiste in una web-app che guida l'utente nel caricamento di un nuovo segnale, viene analizzato e infine classificato in modo autonomo. I risultati vengono espressi sia tramite testo, sia rappresentando la probabilità a posteriori del risultato di classificazione, sia tramite immagini topografiche del voltaggio e della potenza relativa nelle bande delta, teta e alpha.

Il sistema sviluppato dimostra come algoritmi di machine learning possano fornire un supporto fondamentale nella diagnosi automatica in pazienti con GCA, riducendo i costi e tempi di refertazione, mantenendo una qualità di cura elevata. Le accuratezze riscontrate nella soluzione proposta rientrano nell'intervallo della variabilità inter-operatore per la refertazione dei descrittori sopramenzionati, confermando la validità e l'usabilità dell'algoritmo in relazione a un parere neurofisiopatologico. Un ulteriore vantaggio della soluzione proposta consiste nel fatto che i segnali provengono da un ambiente clinico di routine, registrate con il sistema 10-20, senza utilizzo di canali ECG/EOG e senza l'analisi delle componenti indipendenti, rendendo il set-up clinico/sperimentale facilmente riproducibile.

In ultimo, è importante sottolineare che tali soluzioni automatiche potranno in futuro estendere l'accessibilità di questo tipo di refertazione avanzata dei tracciati EEG, permettendone l'uso anche in contesti quali gli ospedali in paesi in via di sviluppo che spesso non possono avvalersi di personale specializzato e formato.

Sommario

Introduzione

Popolazione - Chi sono i GCA

Le Gravi Cerebrolesioni Acquisite (GCA) sono definite come danni cerebrali causati da qualsiasi tipo di evento, traumatico, post-anossico, vascolare o emorragico, che può portare ad un'alterazione della coscienza per almeno 24 ore o un Coma Glasgow Scale (GCS) variabile tra 3 e 8 dopo 24 ore [1]. Negli ultimi anni, i miglioramenti nel trattamento dei pazienti con GCA durante la terapia intensiva hanno notevolmente aumentato le loro possibilità di sopravvivenza, sebbene spesso possano rimanere con una disabilità permanente. Inevitabilmente, un numero cospicuo di pazienti possono anche sviluppare un disturbo della coscienza (DoC) [2]. A seguito di una condizione di coma, i pazienti possono persistere in una sindrome di veglia non responsiva (UWS), in cui hanno cicli sonno-veglia e sono in grado di mantenere le funzioni vitali senza assistenza, possono mostrare risposte riflesse a stimoli tattili, dolorosi, uditivi o visivi, ma non mostrare risposta ai comandi. Un'altra condizione è quella di uno stato di coscienza minima (MCS), uno stato di paralisi corporea quasi completa in cui la coscienza è completamente recuperata ma le possibilità di comunicazione sono generalmente limitate. Quest'ultima è stata recentemente stratificata in MCS+ (più) e MCS− (meno), in base alla complessità delle risposte comportamentali. MCS− si riferisce a pazienti che mostrano solo livelli minimi di interazione comportamentale e movimenti non riflessivi. Possono mostrare orientamento a stimoli nocivi o ricerca visiva di stimoli in movimento o salienti. Gli stimoli ambientali possono suscitare risposte affettuose appropriate, come il pianto o il sorriso innescato da voci o volti familiari. La MCS+ è caratterizzata da comportamenti più complessi come seguire i comandi, apprensione del linguaggio, verbalizzazione intelligibile o risposte sì/no verbali o gestuali [3], [4], [5]. Infine, alcuni pazienti possono emergere da questo stato in uno stato E-MCS che consente il recupero di facoltà come la proprietà di linguaggio funzionale e accurata, stato di lieve confusione, capacità di performare movimenti e riconoscimento di oggetti. Chiaramente, la diagnosi differenziale dei livelli di coscienza nei pazienti con GCA è fondamentale per la definizione di un buon piano riabilitativo e di una prognosi accurata. Queste ragioni hanno portato all'introduzione di strumenti

diagnostici specifici per la valutazione del livello di coscienza e la Coma Recovery Scale-Revised (CRS-R) è stata identificata come punto di riferimento standard per la valutazione clinica della coscienza [6]. Tuttavia, data la presenza di un rilevante rischio di diagnosi errata quando la diagnosi si basa solo sulla valutazione clinica, le ultime linee guida internazionali hanno approvato l'introduzione di valutazioni strumentali come il neuroimaging funzionale o l'elettroencefalografia (EEG), in combinazione con le valutazioni cliniche [5], [7]–[9].

EEG nei GCA

Tra le varie tecniche strumentali, l'EEG è stato ampiamente introdotto nella pratica clinica nei pazienti con GCA negli ultimi anni [10]–[12]. Nel complesso, l'EEG è una valutazione affidabile, ripetibile, facile da usare, non invasiva e a basso costo. Tutti questi aspetti testimoniano la flessibilità di questo strumento, utilizzabile anche in paesi con un sistema sanitario sottosviluppato o durante valutazioni domiciliari in setting ecologici. Inoltre, utilizzando una procedura combinata della PET eseguita con il fluorodesossiglucosio (FDG-PET) con la classificazione dello stato di coscienza derivato su EEG non solo ottimizza le prestazioni diagnostiche, ma consente anche di rilevare l'attivazione corticale residua e di prevedere il recupero a 6 mesi [11], [13]. Al contrario, l'ampia eterogeneità e complessità di tali patologie sono la causa di una grande variabilità inter-operatore sull'interpretazione dei segnali e l'estrazione del significato clinico.

Nel tentativo di standardizzare le linee guida per la refertazione medica dell'EEG, Lawrence J. Hirsch e colleghi[7] hanno proposto una serie di pattern con attributi qualitativi per la refertazione EEG. Ancora oggi la terminologia di riferimento rimane quella secondo dell'American Clinical Neurophysiology Society (ACNS), che ha creato regole oggettive utilizzate per leggere i pattern EEG. Anche se estratta con mezzi qualitativi, tale terminologia EEG di terapia intensiva ha dimostrato di diminuire la variabilità inter-operatore e di aumentare la precisione nella valutazione di specifiche condizioni cliniche. Lo studio di N. Gaspard e i colleghi [14] hanno valutato che la maggior parte degli operatori, con anni variabili di esperienza nell'interpretazione dei segnali EEG, ha mostrato un accordo da sostanziale o

quasi perfetto tra le loro valutazioni, ad eccezione della morfologia trifasica e dell'evoluzione, che hanno mostrato rispettivamente un accordo moderato ed equo. Attualmente, tuttavia, l'elaborazione quantitativa dei dati EEG (qEEG) apre la strada a un'ulteriore riduzione della variabilità inter-operatore nella diagnosi neurofisiopatologia nei pazienti con GCA [9], [15]. In questo contesto, si mira a utilizzare tecniche di analisi del segnale per automatizzare l'identificazione delle caratteristiche dell'ACNS dai segnali EEG acquisiti nella routine clinica.

Utilizzando la terminologia ACNS, i descrittori EEG presi in considerazione in questo studio sono stati: per i dell’attività di background: il voltaggio, la frequenza, la simmetria, il gradiente antero-posteriore; per i pattern ritmico/periodici: le anomalie lente delta; per le scariche epilettiformi: i grafoelementi epilettici.

Stato dell’arte di diagnosi automatica

L’analisi della letteratura rileva come diversi studi hanno affrontato il problema della refertazione automatica di EEG, molti dei quali utilizzano modelli di machine learning (ML) per il rilevamento di caratteristiche (*feature*) e la classificazione. Allo stesso tempo, la maggior parte dei lavori si concentrano su una singola specifica patologia o su uno specifico pattern neurofisiologico.

Ad esempio, U. Rajendra Acharya e colleghi [16] hanno fatto uno studio per lo screening della depressione basato su EEG creando un modello di Convolutional Neural Network (CNN) per differenziare gli EEG ottenuti da soggetti depressivi e normali. Ha raggiunto precisioni del 93,5% e del 96,0% utilizzando segnali provenienti rispettivamente dall'emisfero sinistro e destro ottenuti da 15 pazienti depressi e 15 sani. La tecnica proposta non richiede che un insieme di feature semi-selezionato manualmente venga inserito in un classificatore per la classificazione, ma apprende automaticamente e deriva le features all’interno degli strati della rete.

Jin Jing e colleghi [17] hanno cross-validato un algoritmo mirato all’identificazione di scariche epilettiformi inter-ictali (IED) in modo affidabile. Ha utilizzato un totale di 9571 record EEG del cuoio capelluto con

e senza IED per addestrare una deep neural network (SpikeNet) per eseguire la classificazione. L'intera classificazione EEG ha ottenuto un risultato di AUC (vedere paragrafo metriche di valutazione) di 0,847 (IC 95%, 0,830-0,865).

Forrest Sheng Bao e colleghi [18] hanno presentato uno studio che mira a sviluppare un sistema diagnostico automatizzato in grado di utilizzare i dati EEG inter-ictali per riconoscere i segmenti EEG epilettici. Per sviluppare un tale sistema ha costruito una rete neurale probabilistica (PNN) convalidata con il metodo leave-one-out (LOO-CV) raggiungendo una precisione del 99,3% su cinque set, ciascuno contenente 100 segmenti EEG a singolo canale.

A. Harati e i colleghi [19] invece si pongono come target una classificazione a 6 gruppi: (1) Spike e/o Sharp Wave (SPSW): transitori epilettiformi che si osservano tipicamente nei pazienti con epilessia. (2) Scariche epilettiformi laterali periodiche (PLED): anomalie dell'EEG costituite da picchi ripetuti o scariche di onde acute, che sono focali o lateralizzate su un emisfero e che si ripetono a intervalli di tempo quasi fissi. (3) Scariche epilettiformi periodiche generalizzate (GPED): scariche periodiche diffuse a breve intervallo, scariche periodiche diffuse a lungo intervallo e schemi di soppressione-burst in base all'intervallo tra le scariche. Le onde trifasiche (picchi diffusi e sincroni bilaterali con predominanza bifrontale, tipicamente periodici a una frequenza di 1-2 Hz) sono inclusi in questa classe. (4) Artefatti (ARTF): attività elettrica registrata che non è di origine cerebrale, come quelle dovute ad apparecchiature o ambiente. (5) Eye Blinks (EYEBL): eventi comuni che spesso possono essere confusi per un picco. (6) Sfondo (BCKG): tutti gli altri segnali. Il metodo include estrazione manuale di features con successivo addestramento di un modello di Markov Nascosto standard [20] per ogni classe su un dataset di 3.762 segnali trovando un accuratezza multi-classe superiore al 50% .

R Agarwal e i colleghi [21] hanno creato un software per l'analisi di segnali EEG di lunga durata in grado di analizzare il segnale estrarne delle feature e segmentare il segnale in epoche e raggrupparle per similarità. Questo permette a un clinico di evitare di analizzare tutto il segnale di un paziente ma controllare solo alcuni pezzi significativi. Si tratta di un allenamento non

supervisionato che non mira a una classificazione, ma a un raggruppamento delle epoche. Il modello è stato testato su 41 record EEG di 6 ore ciascuno con una percentuale di concordanza col clinico del 43.9%, del 73.2% entro mezzo livello e del 100% entro 1 livello di agreement.

V. Jahmunah e i colleghi [22] hanno sviluppato uno strumento diagnostico per indagare e classificare i modelli di segnale EEG in classi normali e schizofreniche. La soluzione proposta utilizza 19 canali EEG di 14 soggetti, sette maschi e 7 femmine della durata di 15 minuti per l'addestramento di diversi modelli. Dai segnali sono stati estratti un totale di 157 feature tra cui ne sono state selezionate 14 con il t test di Student. Infine, è stata implementata una pratica di classificazione del segnale tramite Decision-Tree (DT), Linear-Discriminant analysis (LDA), k-Nearest-Neighbour (KNN), Probabilistic-Neural-Network (PNN) e Support-Vector-Machine (SVM) con vari kernel. Il risultato sperimentale ha mostrato che l'SVM-rbf offriva un valore medio di accuratezza pari a 92,9% sul dataset considerato.

Obiettivi

Per colmare il gap nella letteratura riguardante la refertazione automatica nei pazienti con GCA, si propone un metodo per il rilevamento automatico dei descrittori della terminologia ACNS per la terminologia del *Critical Care*. Il lavoro parte dal calcolo di features numeriche estratte da EEG prelevati con il sistema clinico 10-20 durante registrazioni di routine. Dopo uno screening automatico iniziale delle features estratte, sono stati mantenuti i canali o gruppi di canali che trasportavano la maggior parte dell'informazione. Infine, diversi modelli di ML, specifici per ogni descrittore della terminologia ACNS, sono stati addestrati, cross-validati e testati tramite un robusto approccio di cross-validazione annidata [23]. Da qui in poi confideremo i descrittori dell'ACNS come obiettivi dello studio.

Metodi

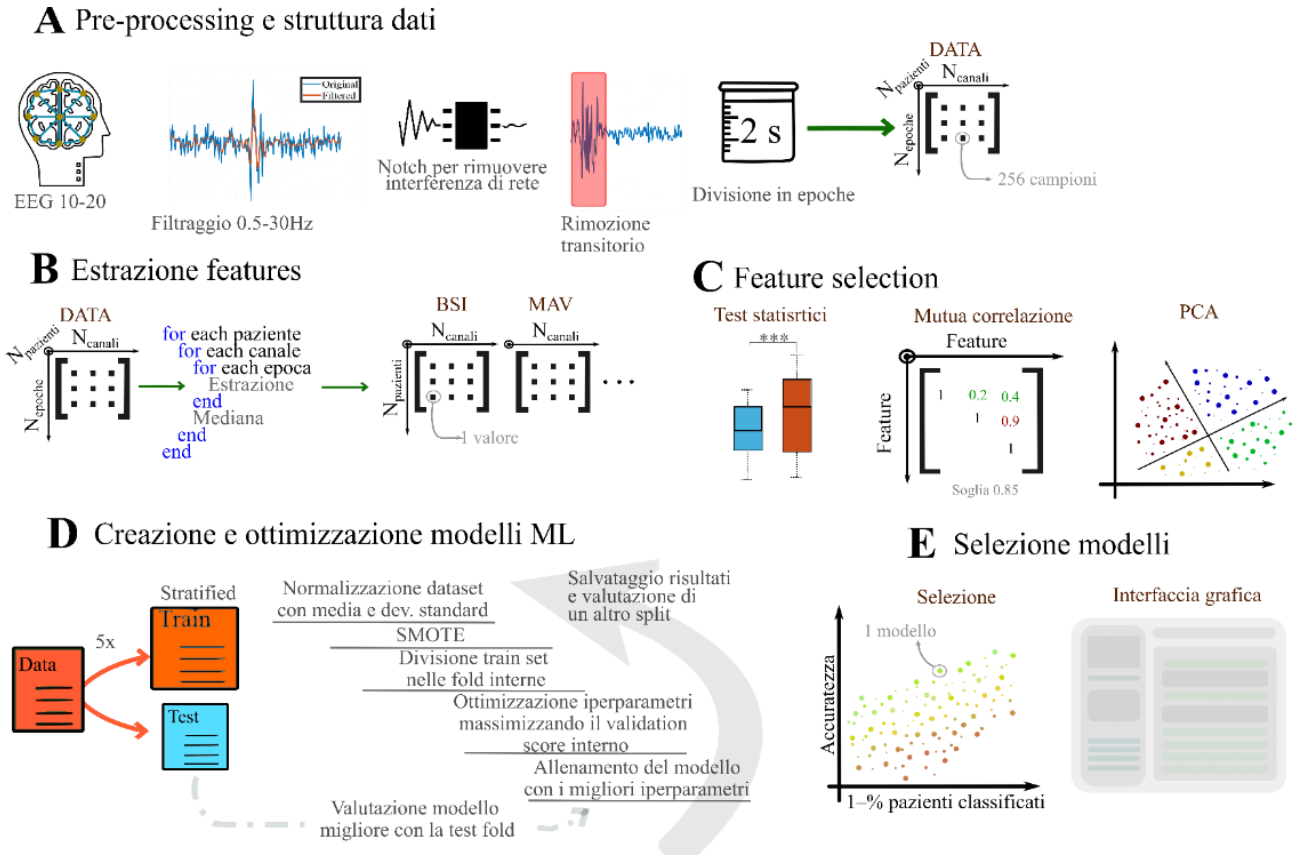


Figura 1 Schema riassuntivo del processo seguito. Questi passaggi sono stati seguiti per ogni descrittore

Introduzione allo studio

Questo studio osservazionale prospettico è stato approvato dal comitato etico locale (Comitato Area Vasta: regione Toscana: N.16606$_{OSS}$). Il presente studio ha seguito i principi della Dichiarazione di Helsinki (1964) e i successivi emendamenti. Il consenso informato è stato ottenuto dai tutori legali di tutti i pazienti. Nello studio sono stati arruolati i pazienti ricoverati presso l'IRU dell'IRCCS Fondazione Don Carlo Gnocchi tra il 1/1/2020 e il 1/4/2022. Sono stati raccolti un totale di 314 pazienti, alcuni dei quali hanno ricevuto valutazioni ripetute durante la permanenza nell'IRU ottenendo un totale di 621 registrazioni. La diagnosi di coscienza è stata effettuata dopo una valutazione clinica utilizzando il valore massimo (di almeno 3 ripetizioni

consecutive) dei sub-items della CRS-R, eseguita entro una settimana dall'ammissione all'unità di riabilitazione. La seguente è stata amministrata da neurologi, logopedisti e neuropsicologi esperti, seguendo criteri diagnostici standard [6] .

Le registrazioni EEG di 20 minuti sono state eseguite utilizzando una macchina digitale (EBNeuro GalNT, Firenze, Italia) e una cuffia EEG con frequenza di campionamento impostata a 128 Hz. Gli elettrodi sono stati posizionati seguendo il sistema standard di prelievo internazionale 10-20 [24] e i dati sono stati archiviati in European Data Format (.edf). Dettagli specifici sui parametri e protocolli di registrazione sono spiegati da Scarpino e i colleghi [11]. Gli EEG sono stati classificati secondo la terminologia ACNS tramite agreement di due diversi neurofisiologi esperti, in cieco rispetto allo stato di coscienza del paziente [7].

Descrittori della terminologia ACNS

Con voltaggio si intende una valutazione qualitativa dell'ampiezza della registrazione che comunque deve essere compreso tra 1-120 μV per essere fisiologicamente accettato. Esso può assumere una categorizzazione dicotomica in: i) ampiezza ridotta o ii) ampiezza normale.

La frequenza può essere etichettata come delta (0.5 – 3 Hz), theta (3 - 7 Hz) o alfa (7 - 14 Hz). Rispecchia in quale banda si trova prevalentemente l'attività di background.

La simmetria è un descrittore binario (presente, assente) e si manifesta come differenza tra gli emisferi. La differenza può essere in ampiezza o frequenza.

Un gradiente antero-posteriore è presente se, in qualsiasi punto dell'epoca, c'è un chiaro e persistente (almeno 1 minuto continuo) gradiente dalla zona frontale alla occipitale in tensione e frequenza. In particolare, nelle derivazioni anteriori si vedono una tensione più bassa e frequenze più veloci, nelle derivazioni posteriori si vedono tensioni maggiori e frequenze più lente. Il gradiente inverso è definito in modo identico ma con un gradiente di tensioni e frequenze da posteriore ad anteriore. I valori possono dunque essere presente, assente, presente ma invertito.

Le anomalie lente delta indicano una disfunzione cerebrale globale nella quale l'attività celebrale rallenta. Sono anomalie nel dominio della frequenza in banda delta e nel dominio del tempo in ampiezza. Il valore di questo descrittore è 1 o 0 a seconda della presenza di tali anomalie. Non sono anomalie periodiche e bastano poche manifestazioni per classificare la registrazione come 1.

I grafo-elementi epilettici possono essere di varie tipologie e forme, chiamati grafotipi, ognuno con il suo specifico significato fisiologico. Questo descrittore assume il valore 1 o 0 a seconda se all'interno della registrazione vi è almeno uno di questi grafotipi oppure se non ce n'è nessuno.

Considerazioni iniziali

Si è utilizzato usato il software Spyder e come linguaggio di programmazione Python. In particolare, la libreria Numpy è stata usata per il calcolo numerico [25], SciPy per il calcolo scientifico [26], MNE per analisi dei segnali EEG [27], *Optuna* per l'ottimizzazione degli iperparametri [28], Sklearn per lo sviluppo di modelli di ML [29] e SHAP per l'intretabilità dei risultati [30].

Sono stati caricati i dati dei pazienti e associati ognuno al proprio referto scaricato da una piattaforma di database online RedCap. I canali registrati per ogni paziente sono: Fp1, Fp2, F7, F3, Fz, F4, F8, C3, Cz, C4, T4, T5, P3, Pz, P4, T6, O1, O2, T3. Viene utilizzato il sistema di prelievo internazionale 10-20 per il prelievo EEG che consiste nel descrivere la posizione degli elettrodi del cuoio capelluto (Figura 2). Questo metodo è stato sviluppato per garantire che i risultati dello studio (clinico o di ricerca)

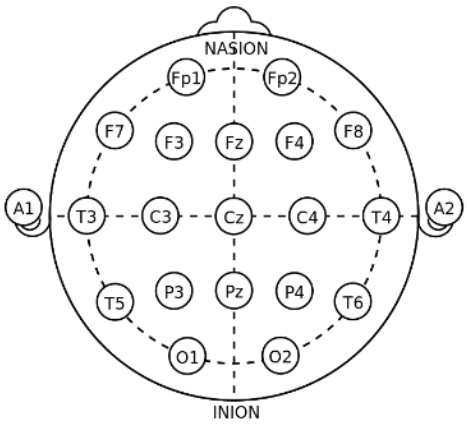


Figura 2 Sistema di prelievo internazionale 10-20 con relativa posizione dei canali sul cuoio capelluto

possano essere compilati, riprodotti, analizzati e confrontati efficacemente. Il sistema si basa sulla relazione tra la posizione di un elettrodo e l'area sottostante del cervello, in particolare la corteccia cerebrale. Il "10" e il "20" si riferiscono al fatto che le distanze effettive tra gli elettrodi adiacenti sono il 10% o il 20% della distanza totale fronte-retro o destra-sinistra del cranio. Le misurazioni vengono effettuate attraverso la parte superficiale della testa, dal *nasion* all'*inion* per la misura fronte-retro, da un padiglione auricolare al suo controlaterale per la misura destra-sinistra.

Considerando che i pazienti con GCA (con un DoC) spesso non sono in grado di seguire indicazioni funzionali, si è deciso di rimuovere dall'analisi i canali Fp1 e Fp2 poiché potrebbero essere artefattati dal battito oculare e i canali O1 e O2 poiché in molti casi artefattati dal movimento del paziente sul poggiatesta. Facendo queste considerazioni per ogni paziente 15 canali EEG sono stati mantenuti per le successive analisi.

Poiché l'attività elettrica cerebrale registrata sull'elettrodo di riferimento può annullare o influenzare in modo significativo l'attività cerebrale registrata dagli elettrodi di registrazione, è spesso necessario un metodo di ri-referenziazione . Per questo motivo si è scelto il *grand average* (media tra tutti i canali) come canale di riferimento per una misura differenziale.

Si effettua subito una prima scrematura sei segnali per individuare errori di registrazione o scollamento di elettrodi; infatti, vengono segnati tutti i punti in cui il potenziale è minore di 1 μV per un tempo minimo di 50 ms. In

seguito, non sarà tenuto conto delle porzioni di segnali contenenti questa annotazione.

Filtraggio

Si è provveduto a filtrare in modo saggio i segnali. La frequenza di campionamento è di 128Hz. Le bande del segnale EEG sono:

δ : 0.5 – 3 Hz;
θ : 3 – 7 Hz;
α : 7 – 14 Hz
β_1 : 14 – 21 Hz;
β_2 : 21 – 30 Hz;

Dato la condizione neurologica dei soggetti, le onde in banda β, (relative a uno stato di concentrazione/ sforzo cognitivo) sono state escluse dall'analisi. Il segnale è stato filtrato tramite un filtro passa banda mostrato nella figura 3 tra 0.5Hz e 30 Hz utilizzando i seguenti parametri:

- Filtro con risposta all'impulso finita (FIR) a fase zero
- Frequenza di taglio inferiore (l_{freq}): 0.50
- Frequenza di taglio superiore (h_{freq}): 30.00 Hz
- Banda di transizione inferiore (l_{bw}): è stata impostata in modo automatico secondo la formula:

$$l_{bw} = \min\left(\max(l_{freq} * 0.25\ , 2), l_{freq}\right)$$

  ottenendo un valore di 0.50 Hz (attenuazione di -6 dB alla frequenza di 0.25 Hz)

- Banda di transizione superiore (h_{bw}): è stata impostata in modo automatico secondo la formula:

$$h_{bw} = min\left(\max(h_{freq} - 0.25\ , 2)\ , \frac{s_{freq}}{2} - h_{freq}\right)$$

  dove s_{freq} è la frequenza di campionamento, ottenendo un valore di 7.5 Hz (attenuazione di -6 dB alla frequenza di 33.75 Hz)

- L'ordine del filtro viene scelto in base alle dimensioni delle regioni di transizione, cioè 6.6 volte il reciproco della banda di transizione più corta. la lunghezza diventa quindi di 845 campioni, 6.6 secondi di transitorio.

A seguito del filtraggio sono stati tagliati i primi e gli ultimi 6.6 secondi del segnale poiché soggetti al transitorio del filtro. Viene tagliato quindi lo 0.55% del segnale originale di 20 minuti. In aggiunta si è usato un filtro Notch per rimuovere la frequenza di rete 50 Hz.

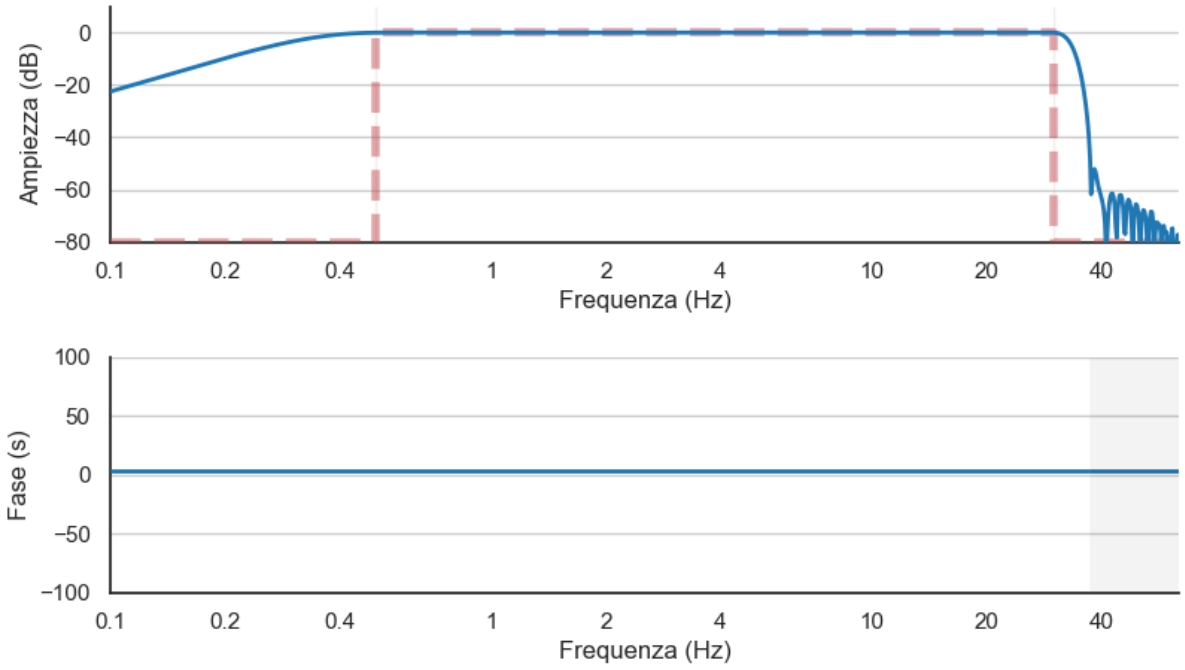


Figura 3 Funzione di trasferimento del filtro utilizzato, in alto la risposta in ampiezza dove in rosso vi è la rappresentazione di un filtro ideale progettato con gli stessi parametri, mentre in blu il filtro reale. Si può notare che avendo usato un filtro FIR non vi è ritardo in fase per nessuna frequenza (in basso).

Segmentazione (epoche)

Prima di procedere all'estrazione delle feature si dividono i segnali in epoche in modo da poter mediare la feature estratta in tutte le epoche e considerare la media come valore rappresentativo del segnale. Si è scelto di fare la lunghezza dell'epoca di 2 s per massimizzare la quantità di epoche e allo

stesso tempo mantenere le frequenze superiori al 0.5Hz poiché facenti parte della banda del segnale. A ogni epoca viene applicato un detrend lineare e viene rimosso il valor medio in modo da non considerare derive temporali degli elettrodi.

Eseguire una segmentazione per taglio netto equivale a applicare una finestra rettangolare sul segnale. Quando si esegue un'analisi delle componenti frequenziali il segnale viene convoluto con la trasformata di Fourier della finestra, questo introduce distorsioni. I lobi laterali di quest'ultima introducono una dispersione di potenza (leakage) che causa polarizzazione delle stime delle PSD di frequenze adiacenti. Così ogni volta che verrà calcolata una componente frequenziale si è deciso di applicare al segnale la finestratura di Hamming perché ha il miglior compromesso tra pendenza laterale e larghezza del lobo centrale. Per le componenti nel dominio temporale invece viene mantenuta la finestratura rettangolare per non distorcere il segnale.

Delle epoche registrate vengono rimosse quelle che escono da un certo valore di ampiezza, come range ragionevole per il segnale EEG è stato considerato 1μV – 120μV. È sufficiente che 1 solo canale sia fuori da questo intervallo per rigettare tutti i canali di quell'epoca. Siccome dovremo successivamente effettuare delle mediane tra le epoche, si elimina a priori tutti i pazienti che presentano meno di dieci epoche "buone" poiché la mediana non sarebbe rappresentativa del paziente. Si riporta in figura 4 i boxplot della percentuale di epoche scartate a causa di ogni canale.

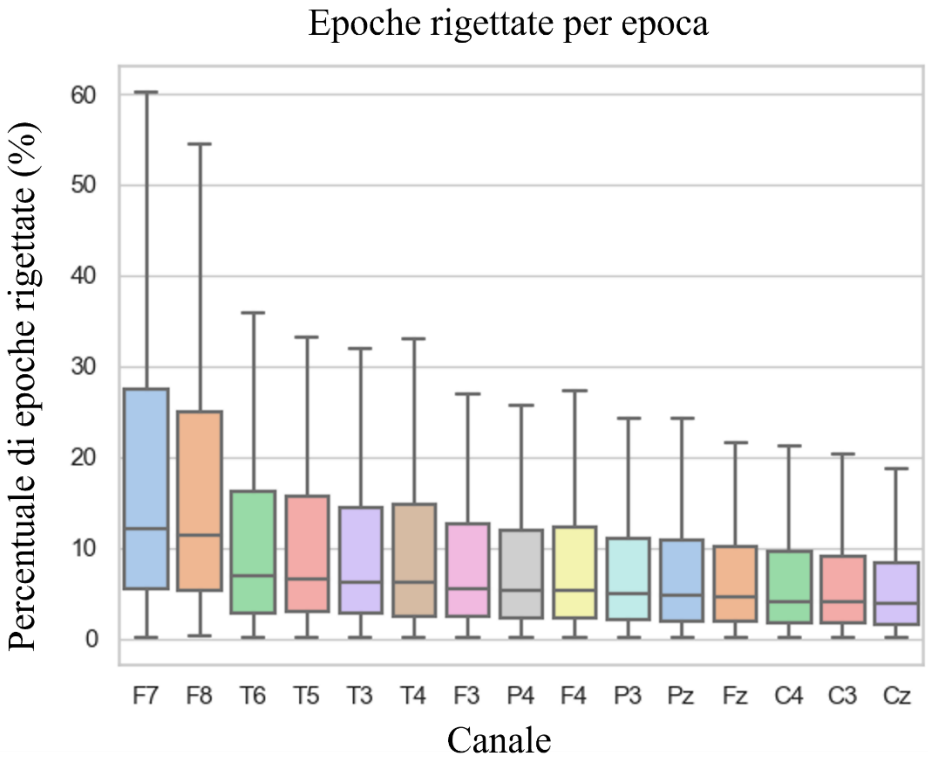


Figura 4 Percentuale di epoche rigettate per ogni canale

Estrazione feature

Background

Per quanto riguarda l'estrazione delle feature viene estratto il valore per ogni epoca di ogni canale di ogni paziente, per poi fare la mediana tra i valori nelle epoche e avere 1 valore di riferimento per ogni canale di ogni paziente. La mediana è stata scelta perché non risente di eventuali outlier e per distribuzioni normali essa è simile alla media. Sono state estratte diverse features nei domini del tempo e della frequenza. Si possono distinguere due gruppi di features, la prima valuta molte caratteristiche su tutto il segnale ("forza bruta"), la seconda ne analizza solo alcune, ma le cerca nelle diverse bande di interesse del segnale EEG.

Sono stati considerati diversi approcci tra i diversi descrittori. Per frequenza - voltaggio - grafoelementi epilettici, sono state estratte le feature come descritto sopra. Per simmetria – gradiente antero-posteriore vengono estratte

le feature per ogni epoca di ogni canale e poi valutata la differenza tra canali o gruppi di canali (zone celebrali) all'interno della stessa epoca, poi è stata fatta la mediana tra i risultati e ottenere il valore rappresentativo. Per anomalie lente delta si è considerato il fatto che esse si presentano solo nel periodo di calibrazione all'inizio del segnale e quindi vengono considerate solo le epoche entro i primi 5 min. Le anomalie lente delta si presentano con una banda che va dai 0.5 a 2 Hz, quindi, sono state scelte epoche di 2 secondi sovrapposte del 50%, in modo da cercare di inglobare un'eventuale anomalia all'interno di un'epoca.

Per quanto riguarda l'estrazione delle feature nelle varie bande si è scelto di considerare le seguenti bande di interesse: (0.5, 3), (0.5, 7), (0.5, 13), (0.5, 20), (0.5, 28), (3, 7), (3, 13), (3, 20), (3, 28), (7, 13), (7, 20), (7, 28), (13, 20), (13, 28), (20, 28); i valori sono espressi in Hz. All'interno di queste bande si è estratto il valore medio del segnale rettificato e la potenza in quella banda. Per effettuare questo il segnale viene filtrato con un filtro 'butter' passa banda di ordine 4. Per il calcolo della PSD viene usato il metodo di Welch, questo calcola una stima della densità spettrale di potenza dividendo i dati in segmenti sovrapposti, calcolando un periodogramma modificato per ciascun segmento e facendo la media dei periodogrammi [31]. Successivamente vengono estratte le potenze relative tra diverse bande e le bande considerate sono: $(\delta/\alpha), (\theta/\alpha), (\delta/\beta), (\delta/\theta), (\alpha/\beta), (\delta/\alpha+\beta), (\theta/\alpha+\beta), (\delta/\theta+\alpha+\beta)$. Per questa feature viene utilizzata densità spettrale di potenza utilizzando i multitaper, ovvero calcola la densità spettrale per i taper [32], quindi li media insieme per ogni canale e epoca [33]. Infine, viene calcolata la media della distanza dei picchi di autocorrelazione trovati con la funzione find_peaks e usando diversi valori di *prominence*: 0.01, 0.02, 0.035, 0.05, 0.075, 0.1. La *prominence* è un valore che indica la prevalenza di un picco rispetto a quelli adiacenti, essa è basata sull'area sottesa dal picco.

Ogni feature estratta Φ_i richiede il calcolo della necessaria trasformazione matematica $f(\cdot)$ sulla registrazione di uno $f(\mathrm{ch})$ o più canali $f(\mathrm{ch}_1, \mathrm{ch}_2, \ldots, \mathrm{ch}_{N_{ch}})$. Per consentire l'aggregazione di tali trasformazioni attraverso gruppi di canali localizzati nello spazio (ChG) si definisce la funzione aggregatore di canali f^{ChG} definita come segue:

$$f^{\mathrm{ChG}}(\mathrm{G}) = \mathrm{ChAgg}\big(f(\mathrm{ch})\big) \doteq \frac{1}{\mathrm{N_{ch}}} \sum_{\mathrm{ch}\in\mathrm{ChG}} f(\mathrm{ch})$$

con N_{ch} il numero di canali nel gruppo di canali ChG. Pertanto, l'aggregazione dei canali comporta l'esecuzione della media tra i canali appartenenti allo stesso gruppo. Le trasformazioni matematiche utilizzate includevano l'indice della potenza nelle varie bande (BSI), l'indice del rapporto di potenza (PRSI), l'indice del valore assoluto medio (MAVSI), l'indice di autocorrelazione (ACSI). Le definizioni fatte per qualsiasi indice quantitativo *R* valgono anche per la controparte sinistra *L*.

Per i descrittori di gradiente antero-posteriore e simmetria è stato necessario estrarre le feature per ogni canale in ogni epoca, per poi calcolarne il rapporto o la differenza tra i diversi canali o diverse aree raggruppando più canali vicini e facendone una media.

Per la simmetria si sono quindi scelti questi gruppi, i canali sotto il Macron (‾) rappresentano un gruppo i cui valori vengono mediati per ottenere un valore rappresentativo dell'area celebrale (figura 5), le parentesi tonde invece rappresentano un'operazione tra due gruppi di canali:

$pairs$: $(F_3 - F_4)\,, (F_7 - F_8), (C_3 - C_4),\ (T_3 - T_4),\ (T_5 - T_6), (P_3 - P_4)$

$lines$: $\left(\overline{F_3F_7} - \overline{F_4F_8}\right), \left(\overline{C_3T_3} - \overline{C_4T_4}\right), \left(\overline{P_3T_5} - \overline{P_4T_6}\right)$

$quart$: $\left(\overline{F_3F_7C_3T_3} - \overline{F_4F_8C_4T_4}\right),\ \left(\overline{P_3T_5C_3T_3} - \overline{P_4T_6C_4T_4}\right)$

$hemis$: $\left(\overline{F_3F_7C_3T_3P_3T_5} - \overline{F_4F_8C_4T_4P_4T_6}\right)$

Allo stesso modo sono stati creati i gruppi per il gradiente antero-posteriore:

$pairs$: $(F_7 - T_5)\,, (F_3 - P_3), (F_z - P_z), (F_4 - P_4), (F_8 - T_6)$

$lines$: $\left(\overline{F_7F_3F_zF_4F_8} - \overline{T_5P_3P_zP_4T_6}\right)$

$quart$: $\left(\overline{F_7F_3} - \overline{T_5P_3}\right), \left(\overline{F_4F_8} - \overline{P_4T_6}\right), \left(\overline{F_7F_3T_3C_3} - \overline{T_5P_3}\right),$
$\left(\overline{F_4F_8C_4T_4} - \overline{P_4T_6}\right), \left(\overline{F_7F_3} - \overline{T_3C_3T_5P_3}\right), \left(\overline{F_4F_8} - \overline{C_4T_4P_4T_6}\right)$

$hemis$: $\left(\overline{F_7F_3F_zF_4F_8} - \overline{T_3C_3C_zC_4T_4\,T_5P_3P_zP_4T_6}\right),$
$\left(\overline{F_7F_3F_zF_4F_8\,T_3C_3C_zC_4T_4} - \overline{T_5P_3P_zP_4T_6}\right)$

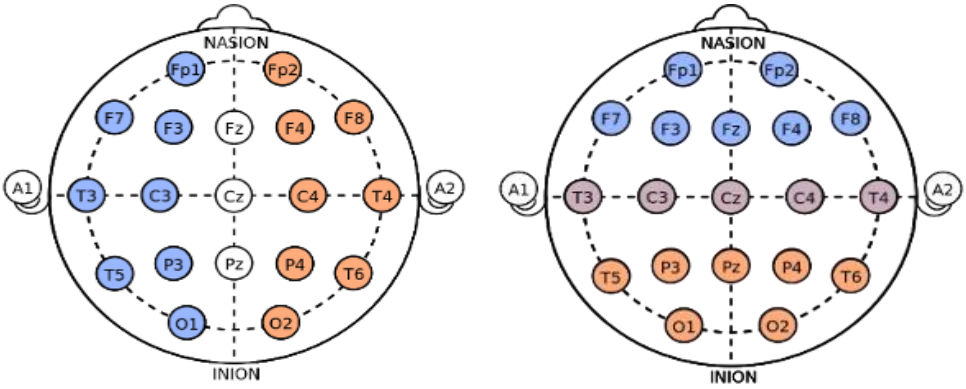


Figura 5 Figure 2 Differenze tra aree celebrali per i descrittori di simmetria (sinistra) e gradiente antero-posteriore (destra). Sono state colorate in blu e in arancione le principali aree considerate. Per il gradiente i canali centrali sono stati considerati a volte nella parte anteriore e a volte in quella posteriore, per questo motivo il colore è misto.

Per facilitarne la comprensione sono stati tutti raggruppati a seconda della tipologia di gruppi unificati: *pairs* sono le coppie, *quarts* sono i quarti, *lines* sono linee perpendicolari a quella che dovrebbe essere l'ipotetico fronte del gradiente del voltaggio, in *hemis* tutti i canali contribuiscono in un gruppo o nell'altro. A tutti questi valori sono stati assegnati dei nomi per identificare l'operazione corrispondente seguiti dalla tipologia di raggruppamenti fatti. Per le feature estratte viene utilizzato l'operatore differenza tra i gruppi di canali in modo da rispettare il range di valori fisionomico con le relative unità di misura. Per gli altri descrittori (frequenza, voltaggio, anomalie lente delta, grafoelementi epilettici) sono state estratte le stesse feature ma senza effettuare delle differenze tra canali, anche il nome è stato mantenuto legato al canale di prelievo.

Entrando nel dettaglio delle feature estratte si riporta l'elenco con il nome della feature, una piccola descrizione e la formula utilizzata:

Valore assoluto medio (MAV)

Si tratta del valore medio del segnale rettificato:

$$MAV = \frac{1}{b_2 - b_1} \sum_{k=b_1}^{b_2} |s_k|$$

Valore assoluto mediano

Rappresenta il baricentro dell'istogramma del segnale rettificato. Esso è uguale alla media se e solo se la distribuzione è normale. La mediana si ottiene prendendo il valore centrale di una serie di valori disposti in ordine crescente.

Deviazione standard

La deviazione standard è un indice di dispersione statistico, vale a dire una stima della variabilità di una serie di dati. Si calcola come:

$$\sigma_X = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu_X)^2}$$

Dove μ_X è la media aritmetica del segnale rettificato.

Range interquartile

Il range interquartile (IQR, figura 6) è la differenza tra il terzo e il primo quartile, ovvero l'ampiezza della fascia di valori che contiene la mediana dei valori osservati. I quartili sono quei valori che ripartiscono la popolazione ordinata in quattro parti di uguale numerosità.

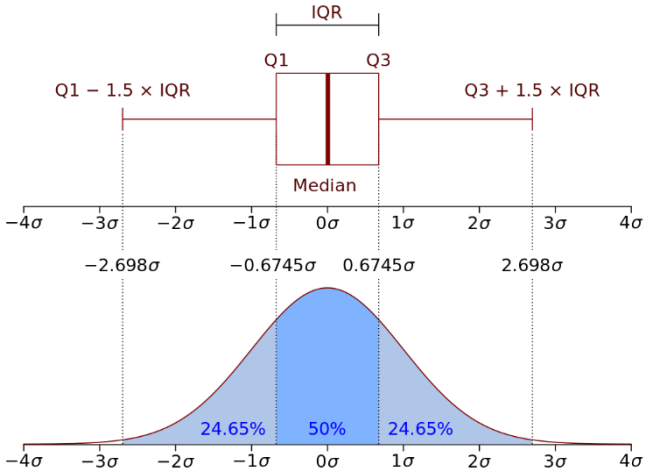


Figura 6 Figura dimostrativa del range interquartile

Skewness e skewness del segnale rettificato

La skewness è una misura dell'asimmetria della distribuzione di probabilità di una variabile casuale a valori reali sulla sua media. Per una distribuzione

unimodale, l'inclinazione negativa indica comunemente che la coda si trova sul lato sinistro della distribuzione e l'inclinazione positiva indica che la coda è a destra. Nei casi in cui una coda è lunga ma l'altra coda è grassa, l'asimmetria non obbedisce a una semplice regola. Ad esempio, un valore zero significa che le code su entrambi i lati della media si bilanciano complessivamente; questo è il caso di una distribuzione simmetrica, ma può anche essere vero per una distribuzione asimmetrica in cui una coda è lunga e sottile e l'altra è corta ma grassa. La skewness di una serie si può misurare con:

$$\tilde{\mu}_3 = E[(\frac{X - \mu}{\sigma})^3]$$

dove μ è la media, σ è la deviazione standard, E è l'operatore di valore atteso.

È stata anche implementata la stessa funzione sul segnale rettificato.

Kurtosis e kurtosis del segnale rettificato

L'indice di kurtosis è uno degli indici relativi alla forma di una distribuzione, che costituisce una misura dello "spessore" delle code di una funzione, ovvero il grado di "appiattimento" (distribuzione platicurtica) o di "allungamento" (distribuzione leptocurtica) di una distribuzione. La kurtosis si può calcolare come:

$$\gamma_2 = \frac{m_4}{m_2^2} - 3$$

dove m_4 e m_2 sono rispettivamente il momento centrale di ordine 4 e 2.

È stata implementata la stessa funziona anche sul segnale rettificato ottenendo un altro valore di kurtosis

Range

Il range è identificato come la differenza tra il massimo e il minimo del segnale. Rappresenta il codominio del segnale.

$$range = \max(s) - \min(s)$$

Numero di picchi

Con questa feature si rappresenta il numero di picchi del segnale. Ovvero il numero di volte in cui si annulla la derivata prima e la derivata seconda è negativa.

Massimo della derivata prima

Questa feature rappresenta il massimo della derivata del segnale ovvero la pendenza massima che si raggiunge in valore assoluto. Esso rappresenta la massima velocità di depolarizzazione rilevata in un canale.

$$maxDerivata1 = \max\left(\frac{\partial s(t)}{\partial t}\right)$$

Massimo della derivata seconda

Questa è il massimo della derivata seconda del segnale:

$$maxDerivata2 = \max\left(\frac{\partial^2 s(t)}{\partial t^2}\right)$$

Integrale di linea

Questa feature è indice di quanto è tortuoso il segnale, è identificato come l'integrale di linea:

$$L = \int_{\gamma} s(l)\ \mathrm{d}l$$

Numero di zero-crossing

Qui si vuole indicare il numero di volte in cui il segnale passa lo zero, ovvero da potenziale positivo diventa negativo e viceversa. È stato calcolato:

$$numZeroCrossing = \sum \frac{\partial\ sign(s)}{\partial t}$$

Rapporto segnale rumore

Indica la quantità di rumore gaussiano bianco è presente sul segnale, Dato che anche l'EEG ha un andamento non deterministico questo valore è definito come ampiezza del segnale diviso quattro volte la deviazione standard:

$$SNR = \frac{A_s}{4\sigma_s}$$

Ampiezza picco-valle

Indica la media tra le differenze in ampiezza tra un picco e la valle immediatamente successiva, è stata ottenuta mediando ogni differenza.

Potenza assoluta e relativa in bande delta, teta, alpha

Sono una serie di feature che identificano la potenza espressa nelle bande delta, teta e alpha. Queste potenze vengono espresse in modo assoluto o relazionate all'intera potenza del segnale. La potenza è stata calcolata eseguendo la PSD con il metodo di Welch del segnale e selezionate solo le bande di interesse. Lo stimatore utilizzato consiste nel calcolo del periodorgamma del segnale segmentato in epoche di 2 secondi e sovrapposte del 50% su cui è stata applicata la finestratura di Hamming. I periodogrammi vengono infine mediati.

Centroide spettrale

Il centroide spettrale misura la posizione del centro di massa dello spettro del segnale, è la media ponderata delle frequenze presenti nello spettro con grandezze come i pesi:

$$centroide = \frac{\sum f(n)x(n)}{\sum x(n)}$$

Dove *x(n)* rappresenta l'ampiezza della frequenza e *f(n)* la frequenza centrale di quel segnale.

Cresta spettrale

La cresta spettrale è definita come il rapporto tra il massimo dello spettro e la media aritmetica dello spettro:

$$cresta = \frac{\max\left(s_{k\in[b1,b2]}\right)}{\frac{1}{b_2 - b_1}\sum_{k=b_1}^{b_2} s_k}$$

dove s_k è la k-esima componente spettrale di potenza, b_1 e b_2 i bordi di banda su cui calcolare la cresta spettrale. La cresta spettrale è un'indicazione del picco dello spettro, più alta indica più chiarezza, mentre una cresta spettrale più bassa indica più rumore.

Entropia spettrale

L'entropia spettrale (SE) di un segnale è una misura della sua distribuzione di potenza spettrale. Il concetto si basa sull'entropia di Shannon, o entropia dell'informazione, nella teoria dell'informazione. L'SE tratta la distribuzione di potenza normalizzata del segnale nel dominio della frequenza come una distribuzione di probabilità e calcola la sua entropia di Shannon. L'entropia di Shannon e l'entropia spettrale sono equivalenti in questo contesto. La sua formulazione si basa sulla seguente distribuzione di probabilità:

$$P(m) = \frac{S(m)}{\sum_i S(i)}$$

P(m) è la distribuzione di probabilità calcolata dallo spettro di potenza *S(m)* del segnale (lo spettro di potenza è calcolato come modulo quadrato dello spettro del segnale).

$$SE = -\sum_m P(m) log_2\big(P(m)\big)$$

Dove *H* è l'entropia spettrale; può essere normalizzato da $log_2(N)$, dove *N* è il totale dei punti di frequenza. Rappresenta la massima entropia spettrale del rumore bianco, uniformemente distribuito nel dominio della frequenza.

Uniformità spettrale

L' uniformità spettrale o *spectral flatness* misura il rapporto tra la media geometrica dello spettro e la media aritmetica dello spettro.

$$uniformità = \frac{\left(\prod_{k=b_1}^{b_2} s_k\right)^{\frac{1}{b_2-b_1}}}{\frac{1}{b_2-b_1}\sum_{k=b_1}^{b_2} s_k}$$

Dove s_k è la *k*-esima componente dello spettro di potenza, b_1 e b_2 i bordi di banda su cui calcolare la planarità spettrale. Come la cresta spettrale, la uniformità spettrale è un'indicazione del picco dello spettro, dove una se maggiore indica rumore, mentre se inferiore indica chiarezza.

Flusso spettrale

Il flusso spettrale è una misura della velocità con cui cambia lo spettro di potenza di un segnale, calcolato confrontando lo spettro di potenza di un'epoca con lo spettro di potenza dell'epoca precedente. Più precisamente, è calcolato come la distanza L2 tra i due spettri. La sua formulazione è la seguente:

$$flusso(t) = \left(\sum_{i=k}^{B} |ep_s(i) - ep_s(i-1)|^2\right)^{\frac{1}{2}}$$

Dove s_k è il valore spettrale alla k-esima frequenza della banda, B è la larghezza di banda su cui viene calcolato il flusso.

Kurtosis spettrale

Lo spectral kurtosis è il valore di kurtosis (che è il momento del quarto ordine) dello spettro di potenza. È una misura della planarità dello spettro attorno al suo baricentro, dove valori più bassi indicano uno spettro più rumoroso.

Roll-off spettrale

Il roll-off spettrale è la frequenza al di sotto della quale è contenuta una percentuale (valore standard 95%) dell'energia totale del segnale.

$$\sum_{k=b_1}^{i} s_k = 0.95 \sum_{k=b_1}^{b_2} s_k$$

Dove s_k è il valore spettrale alla frequenza k, b_1 e b_2 i bordi della banda, su cui calcolare il roll-off spettrale. Può essere utilizzato per differenziare tra larghezza di banda armonica e rumorosa dello spettro del segnale.

Skewness spettrale

Lo spectral skewness è il valore di skewness (che è il momento del terzo ordine) dello spettro di potenza. È una misura della simmetria attorno al baricentro: indica la forza relativa delle armoniche superiori e inferiori. Ad esempio, uno skewness positivo indica una predominanza delle frequenze più basse rispetto a quelle più alte e viceversa.

Diffusione spettrale

La diffusione spettrale o *spectral spread* è la deviazione standard attorno al baricentro spettrale.

$$spectral\ spread = \sqrt{\frac{\sum_{k=b_1}^{b_2}(f_k - \mu)^2 s_k}{\sum_{k=b_1}^{b_2} s_k}}$$

Dove *fk* è la frequenza in Hz corrispondente al campione *k*, μ è il baricentro spettrale, s_k è il valore dello spettro di potenza alla frequenza *k*, b_1 e b_2 i bordi di banda su cui calcolare lo *spectral spread*.

Massimo dell'autocorrelazione

Questa feature indica il valore massimo della sequenza di autocorrelazione, essa si presenta con un ritardo pari a zero per segnali non periodici.

Energia spettrale

Qui si calcola l'energia del segnale come integrale della densità spettrale di energia.

$$Sp_{energy} = \int PSD_s(f)\ df$$

Area

Esso calcola l'integrale del segnale rettificato.

$$A = \int |\mathrm{s(t)}|\ dt$$

Area dell'autocorrelazione

Qui si calcola l'integrale della sequenza di autocorrelazione.

$$AC_{energy} = \int S(\tau)\ d\tau$$

Brain Signal Index (BSI)

Il BSI ha prodotto una misura di simmetria basata sul coefficiente della fast Fourier transform (FFT).

$$BSI = \frac{1}{K}\sum_{k=1}^{K} \left| \frac{R(k) - L(k)}{R(k) + L(k)} \right|$$

dove $\mathrm{R(k)} = \frac{1}{\mathrm{N_{ch}}}\sum_{\mathrm{ch}=1}^{\mathrm{N_{ch}}} \alpha_{\mathrm{k}}(\mathrm{ch})$, $\alpha_{\mathrm{k}}(\mathrm{ch})$ è il coefficiente di Fourier alla frequenza $k{\cdot}f_{sampling}$ e K il numero di campioni nella FFT. In questa implementazione, viene seguita l'implementazione basata su Welch di Sheoraojpanday e i colleghi [34] e calcolato il BSI per tutti i tipi e le posizioni dei ChG. Quando sono state considerate coppie di elettrodi simmetrici, l'equazione precedente diventa:

$$pdBSI = \frac{1}{N_{ch}\ K}\sum_{ch\,=\,1}^{N_{ch}}\sum_{k\,=\,1}^{K} \left| \frac{R_{k,ch} - L_{k,ch}}{R_{k,ch} + L_{k,ch}} \right|$$

mostrando come il concetto ChG consenta che la coppia derivata sia un caso limite di una misura generale di simmetria del coefficiente di Fourier. In generale, il ChG-BSI è stato calcolato come segue:

$$ChG - BSI^{G} = \frac{1}{K}\sum_{k=1}^{K} \left| \frac{R_k^G - L_k^G}{R_k^G + L_k^G} \right|$$

dove $\mathrm{R_k^G} = \mathrm{ChAgg}\left(\widehat{\alpha_{\mathrm{k}}}(\mathrm{ch})\right)$, $\widehat{\alpha_{\mathrm{n}}}(\mathrm{ch},\mathrm{t})$ è la stima Welch PSD e $\mathrm{R^G}$, $\mathrm{L^G}$ il sottoinsieme destro e sinistro del gruppo di canali G.

Power Ratio Signal Index

Il background di frequenza e i rapporti tra potenza in bande specifiche hanno già dimostrato di differenziare molte condizioni e patologie [35]. In particolare, è noto che rapporti noti come il rapporto $\frac{\delta}{\alpha}$ (DAR) o il rapporto $\frac{\delta+\theta}{\alpha}$ (DTAR) sono correlati a condizioni cerebrali specifiche e al livello di coscienza dei pazienti. Nello specifico, ciascun indice di simmetria del rapporto di potenza (PRSI) può essere calcolato come il rapporto tra la potenza di frequenza in diverse bande (es. DAR= $\frac{\mathrm{P}_\delta}{\mathrm{P}_\alpha}$). Tra tutte le possibili

combinazioni di bande al numeratore $\mathcal{N}$ e al denominatore $\mathcal{D}$ è stato scelto un insieme limitato di combinazioni sulla base di quanto segue:

- $\mathcal{N} \cap \mathcal{D} = \emptyset$
- Tutte le bande in *N* hanno una frequenza inferiore rispetto alle bande in *D*
- Nessuna somma è consentita nel numeratore *N* per evitare combinazioni già calcolate.

Attraverso tali regole, un insieme $\mathcal{R}$ delle possibili coppie $(\mathcal{N}, \mathcal{D})$ è stato utilizzato come parametro durante l'estrazione del PRSI. In particolare, il PRSI, definito come:

$$\mathrm{PRSI} = \left|\mathrm{L}^{\mathrm{G}} - \mathrm{R}^{\mathrm{G}}\right|$$

è stato estratto per tutte le combinazioni in $\mathcal{R}$ con $\mathrm{R}^{\mathrm{G}} = \frac{\sum_{\mathrm{band_N} \in \mathcal{N}} \mathrm{P}^{\mathrm{G}}_{\mathrm{band_N}}}{\sum_{\mathrm{band_D} \in \mathcal{D}} \mathrm{P}^{\mathrm{G}}_{\mathrm{band_D}}}$, $\mathrm{P}^{\mathrm{G}}_{\mathrm{band}} = \mathrm{ChAgg}\left(\mathrm{P}^{\mathrm{ch}}_{\mathrm{band}}\right)$ e

$$\mathcal{R} = \{\frac{\delta}{\alpha}, \frac{\theta}{\alpha}, \frac{\delta}{\beta}, \frac{\theta}{\beta}, \frac{\alpha}{\beta}, \frac{\delta}{\alpha+\beta}, \frac{\theta}{\alpha+\beta}, \frac{\delta}{\theta+\alpha+\beta}\}$$

Mean Absolute Value Signal Index

Le misure basate sull'ampiezza non sono influenzate dai valori assoluti dei gruppi di canali R e L ma solo dal valore assoluto della differenza. L'ampiezza valutata tramite il valore medio assoluto (MAV) è stata estratta dopo che i segnali sono stati ulteriormente filtrati con un filtro IIR del quarto ordine Butterworth per ridurre l'effetto delle variazioni istantanee dell'ampiezza dovute a variazioni di alta frequenza, che in caso di brevi epoche (2 secondi) può influenzare il calcolo di tale indice. Inoltre, l'utilizzo di una doppia passata anche di un filtraggio anti-causale garantisce uno sfasamento nullo. Anche per il MAVSI sono state impostate come parametri di estrazione le bande di frequenza δ, θ, α, β.

Il MAVSI è definito come la differenza relativa tra MAV calcolata su ChG sinistro e destro, rispetto al MAV di riferimento (REF), in questo caso il grand average:

Nello specifico, il MAVSI è stato calcolato come

$$\mathrm{MAVSI} = \left|\frac{L^G - R^G}{REF}\right|$$

dove $R^G = \frac{1}{N}\sum_{n=1}^{N}\left|r^G(n)\right|$, N è il numero di campioni in un'epoca e $r^G(n) = \mathrm{ChAgg}\left(r^{ch}(n)\right)$.

AutoCorrelation Signal Index

La terminologia dell'EEG di terapia intensiva ha definito l'asimmetria di frequenza come una differenza di oltre 0,5 Hz tra i gruppi sinistro e destro. Nella pratica clinica per frequenza di fondo si intende l'inverso del periodo della componente oscillatoria principale del segnale. Poiché i segnali EEG contengono diverse forme d'onda e componenti più o meno rumorosi, non è possibile estrarre una tale quantità dal segnale originale. La funzione di autocorrelazione di un segnale periodico mantiene le oscillazioni con lo stesso periodo: tuttavia, qualsiasi segnale non correlato come rumore e artefatti sporadici è meno evidente nella funzione di autocorrelazione (ACF). Quest ultimo può essere definito come:

$$\mathrm{ACF}_R(\tau) = r^G(n) \otimes r^G(-n) = \sum_{n}^{N} r^G(n) r^G(n-\tau)$$

dove N è il numero di campioni in un'epoca e $r^G(n) = \mathrm{ChAgg}\left(r^{ch}(n)\right)$

Sull'ACF, una ricerca iterativa dei picchi ha consentito la stima dell'inverso del numero di picchi. *L'Autocorrelation Signal Index* (ACSI) può essere quindi definito come ;

$$\mathrm{ACSI} = |L_G - R_G|$$

dove $R_G = \frac{1}{T_R^G}$ e T_R^G il periodo stimato dell'autocorrelazione ChG. In questo caso, T_R^G è stato approssimato tramite

$$T_R^G \approx \frac{1}{N_P - 1}\sum_{j=i+1}^{p\in P}\left(p_i - p_j\right)$$

con p_i il tempo di picco nell'insieme ordinato di picchi *P* trovato nell'ACF, con $p_0=0$ e N_p il numero totale di picchi trovati.

Feature selection

Test statistici

Si va ora a valutare quali di queste feature sono correlate per i descrittori tramite i test statistici appropriati. Per ogni descrittore da valutare si va a dividere i pazienti in due o più gruppi a seconda di quanti possibili valori può assumere, per esempio dal referto del medico i valori di simmetria possono essere 1=simmetrico, 0=asimmetrico, in totale 2 gruppi; mentre i valori di frequenza possono essere 0=delta, 1=teta, o 2=alpha, in totale 3 gruppi.

Per i descrittori binari si esegue il test statistico di Mann Whitney, condizionato al test di normalità di Shapiro-Wilk, per calcolare il p-value e dire se i due gruppi sono statisticamente differenti, se lo sono si considera quella feature come variabile rilevante per quel descrittore. Il test Mann-Whitney è una versione non parametrica del t-test per campioni indipendenti.

Per descrittori non binari viene eseguito il test statistico di Kruskal-Wallis [36] , condizionato sempre a test di normalità, per valutare se all'interno dei gruppi vi sono delle differenze. Il test di Kruskal-Wallis verifica l'ipotesi nulla che la mediana della popolazione di tutti i gruppi sia uguale, è una versione non parametrica di ANOVA. Il test funziona su due o più campioni indipendenti, che possono avere dimensioni diverse. Si noti che rifiutare l'ipotesi nulla non indica quale dei gruppi differisce. Sono necessari confronti post hoc di Dunn con la restrizione di Bonferroni[37] tra gruppi per determinare quali gruppi sono diversi.

Per poter vedere in modo più qualitativi i dati estratti vengono proposti dei boxplot come in figura 7 per ogni feature di ogni descrittore. Vengono messi gli asterischi in base al p-value ricavato al passo precedente tra le diverse classi: *p<0.05, **p<0.01, ***p<0.005, ****p<0.0001, ns sta per gruppi non significativamente differenti. Dal boxplot si possono avere informazioni qualitative sulle mediane, i range interquartili e gli outlier. I punti invece rappresentano tutti i pazienti che ricadono in quella statistica, in modo da avere l'informazione relativa alla numerosità delle classi. Si riporta un esempio che rappresenta quanto la feature del valore mediano sul segnale rettificato è correlata con la frequenza per ogni canale:

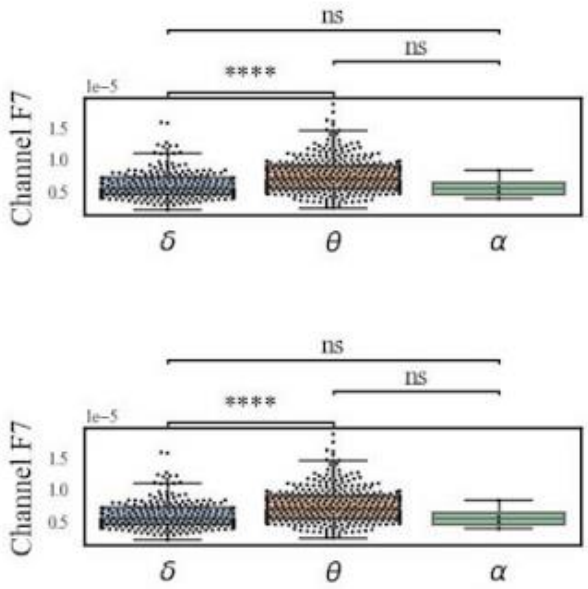


Figura 7 Boxplot che rappresenta i test statistici per quanto riguarda il valore mediano del segnale rettificato in relazione alla frequenza. Vengono messi gli asterischi in base al valore di p ottenuto con il test statistico

Grazie a questi boxplot si riesce a valutare la qualità di una feature e la sua espressività sul relativo descrittore. Si è deciso quindi di eliminare a priori alcune features e i motivi possono essere: un elevato numero di outlier, mediane e interquartili coincidenti tra i gruppi.

Si riporta quali feature sono state rimosse per uno dei motivi sopra citati per ogni descrittore:

Simmetria: Twisted, SpectralSkewness, Sweekness, AbsPowerDelta, StandardDeviation, AbsPowerTeta, NumPeaks, SpectralKurtosis, MedianAbsoluteValue, InterquartileRange, MaxDerivate2, SpectralCentroid, SpectralFlatness, MaxAutocorrelation, MaxDerivate1, MeanAbsoluteValue, SpectralRollOff, Kurtosis, SpectralCrest, AbsPowerAlfa, Peak2Valley, SpectralSpread, RelPowerTeta, Range, RelPowerAlfa, AbsKurtosis, NumZeroCrossing, AbsSweekness, SpectralEntropy, RelPowerDelta, SpectralVariance.

Gradiente antero-posteriore: Twisted, SpectralSkewness, Sweekness, AbsPowerDelta, StandardDeviation, AbsPowerTeta, NumPeaks, SpectralKurtosis, MedianAbsoluteValue, InterquartileRange, MaxDerivate2, SpectralCentroid, SpectralFlatness, MaxAutocorrelation, MaxDerivate1, MeanAbsoluteValue, SpectralRollOff, Kurtosis, SpectralCrest, AbsPowerAlfa, Peak2Valley, SpectralSpread, RelPowerTeta, Range, RelPowerAlfa, AbsKurtosis, NumZeroCrossing, AbsSweekness, SpectralEntropy, RelPowerDelta, SpectralVariance.

Voltaggio: Sweekness, AbsSweekness, Kurtosis, AbsKurtosis, MaxDerivate2, AbsPowerDelta, AbsPowerTeta, AbsPowerAlfa, SpectralFlatness, SpectralEntropy, SpectralRollOff, SpectralSpread, MaxAutocorrelation, SpectralVariance.

Anomalie lente delta: AbsKurtosis, SpectralKurtosis, SpectralCrest, AbsPowerAlfa, SpectralVariance, Kurtosis, Sweekness, AbsPowerDelta, MaxAutocorrelation, SpectralSkewness, SpectralFlatness, AbsPowerTeta, AbsSweekness, MaxDerivate2.

Frequenza: AbsSweekness, AbsKurtosis, NumPeaks, Peak2Valley, AbsPowerDelta, AbsPowerTetaAbsPowerAlfa, SpectralFlatness, SpectralEntropy, SpectralVariance, AbsPowerTeta, AbsPowerAlfa.

Grafoelementi epilettici: SpectralKurtosis, SpectralCrest, AbsPowerAlfa, SpectralVariance, Kurtosis, Sweekness, AbsPowerDelta, Twisted, SpectralSkewness, SpectralFlatness, MaxDerivate1, AbsPowerTeta, MaxDerivate2.

Strategia di feature selection

Fino ad ora il numero di feature ottenuto è troppo elevato, molte delle quali sono correlate tra loro o con informazioni ridondanti. Per questo motivo è stato adottato una strategia di feature selection in modo da ridurre il numero di feature su cui allenare un modello. Sono stati adottati tre metodi applicati in ordine e in cascata l'uno all'altro, tutti selezionano un gruppo di feature secondo diversi aspetti.

Il primo metodo è quello della mutua correlazione. Viene creata una matrice triangolare (figura 8) in cui sulle righe e sulle colonne vi coso le feature e nelle celle la rispettiva mutua correlazione. Vengono eliminate le feature il cui valore è maggiore di una soglia preimpostata (0.99 e 0.95).

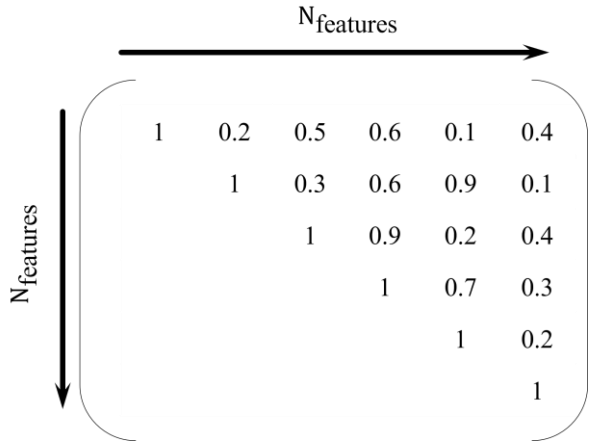


Figura 8 Esempio matrice triangolare della mutua correlazione tra features

Il secondo metodo è basato sull'analisi statistica e il p-value. Grazie ai test statistici precedentemente descritti si valuta la correlazione di ogni feature con il descrittore e si escludono tutte quelle con un p-value minore di una soglia preimpostata (0.05 e 0.01).

Il terzo metodo è l'analisi dei componenti principali (PCA, figura 9). Questo metodo prevede la riduzione della dimensionalità lineare utilizzando la scomposizione del valore singolare per proiettarli in uno spazio dimensionale inferiore. L'obiettivo è ottenere nuove feature ordinate per varianza spiegata sul descrittore e il più possibile ortogonali tra loro. Successivamente viene

messa una soglia arbitraria (0.9999, 0.999 e 0.99) per considerare solo le feature che spighino tale percentuale di varianza.

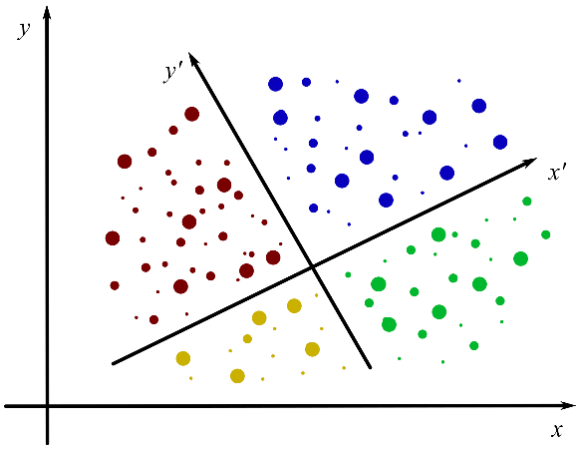


Figura 9 Immagine di ortogonalizzazione della PCA

È stato implementato un quarto metodo chiamato Sequential Feature Selection, il quale però non ha portato un'utilità pratica e per cui non è stato usato in seguito. Questo metodo prevede l'utilizzo di modello di classificatore che viene inizialmente allenato, in questo caso un classificatore KNN e vengono estratte le feature a cui sono state associati dei pesi maggiori. Essi rappresentano secondo il classificatore le feature più significative. A seconda dei valori imposti si limitano il numero di feature a discapito della varianza spiegata dal sistema.

Costruzione modelli ML

Panoramica sui metodi di cross-validazione

In questa sezione si analizzano diverse strategie di cross-validazione di un modello predittivo. La cross-validazione è un processo che garantisce la possibilità di scegliere il modello migliore tra quelli creati. Solitamente si divide il dataset in diversi "*set*" in modo da poter allenare il modello su un dataset di allenamento (*train set*), validarlo su un dataset di validazione (*validation set*) e poter scegliere il modello migliore o fare scelte arbitrarie e infine il modello migliore ottenuto si valuta su un nuovo dataset di test (*test set*). Queste suddivisioni sono necessarie per capire quanto un modello sia in grado di generalizzare il problema posto; infatti, dovrebbe essere in grado di

valutare elementi mai visti in precedenza nel migliore dei modi possibili. Per fare ciò si riportano alcune procedure per la creazione dei diversi dataset.

Esistono diversi modi per dividere il dataset originale di grandezza *N* nei diversi gruppi, il più semplice è il leave one out, ovvero considera un solo elemento nel validation set e utilizza tutti gli altri per la costruzione del modello. Vengono fatte *N* ripetizioni per ogni modello in modo che ogni elemento sia stato usato una volta come elemento di test. Tuttavia, effettuare le *N* ripetizioni può comunque richiedere un tempo di calcolo piuttosto elevato, nel qual caso altri approcci come la k-fold possono essere più appropriati.

Il metodo k-fold invece divide il dataset con un'estrazione casuale di un certo numero di campioni come appartenenti al validation set, i restanti formano il train set. Questo metodo però non tiene conto della tipologia di campioni considerati. Per evitare il problema si preferisce usare il metodo stratified *k*-fold il quale divide il dataset in due set in modo casuale, ma mantenendo il più possibile il rapporto di numerosità tra le classi. Questo metodo necessita di $^{N}/_{k}$ iterazioni in modo che ogni elemento sia stato considerato una volta come test

Nel metodo holdout, si assegnano casualmente i campioni a due set, train set e validation set. La dimensione di ciascuno dei set è arbitraria, anche se in genere il set di test è inferiore al set di allenamento. Quindi si esegue il training e si valuta. Nella tipica cross-validazione, i risultati di più esecuzioni di test del modello vengono mediati insieme; al contrario, il metodo holdout comporta una singola esecuzione. Dovrebbe essere usato con cautela perché senza tale media di più corse, si possono ottenere risultati altamente fuorvianti. Il proprio indicatore di accuratezza predittiva tenderà ad essere instabile poiché non sarà appianato da più iterazioni. Il metodo di holdout può essere inquadrato come "il più semplice tipo di cross-validazione ", molte fonti classificano invece come un tipo di "validazione semplice", piuttosto che una forma semplice o degenerata di cross- validazione.

Il metodo bootstrap ricampiona il dataset con sostituzione producendo nuovi dataset "surrogati" con lo stesso numero di casi del set di dati originale. A

causa dell'algoritmo di sostituzione, un dataset bootstrap può contenere un numero arbitrario di istanze degli stessi campioni originali.

In questo studio è stato usato un altro metodo più comlesso: cross-validazione incrociata (*nested cross-validation*) con il metodo stratifid *k*-fold per la selezione degli elementi nei dataset.

Nested

Un nuovo metodo per validare i modelli di machine learning su un set di dati è utilizzare la cross-convalida nested [38]. La procedura divide un set di dati limitato in *k* gruppi chiamati fold non sovrapposti e la dimensione di ogni fold è $1/k$ per il numero totale dei campioni. Questo procedimento viene effettuato *k* volte, chiamati "split", dove ciascuna delle *k*-fold viene utilizzata come test set in split diversi, mentre tutte le altre fold collettivamente vengono utilizzate come dati di addestramento. Un totale di *k* modelli viene allenato e testato, infine viene riportata la prestazione media. Questo viene chiamato loop esterno. Ogni algoritmo di machine learning include uno o più iperparametri che consentono di personalizzare il comportamento dell'algoritmo in base a un set di dati specifico. Il problema è che raramente, se non mai, c'è una buona euristica su come configurare gli iperparametri del modello per un dataset. Al contrario, viene utilizzata una procedura di ottimizzazione per individuare un set di iperparametri che funzionano bene o meglio nel train set. In questo modo è stato usato un'altra suddivisione in *k*-fold per andare a valutare il set di iperparametri, questo si chiama loop interno. A partire dal train set estratto dal loop esterno si va a creare altre *k*-fold andando suddividere il train set in un nuovo train set e il validation set. È proprio in questo loop interno si può andare a creare diversi modelli, valutarli e si può inoltre effettuare delle scelte andando a selezionare il modello migliore. Si nota che il *k* del loop esterno può essere diverso da quello

del loop interno. (Figura 10). In questo studio sono stati utilizzati 5 split per il loop esterno e 10 split per il loop interno.

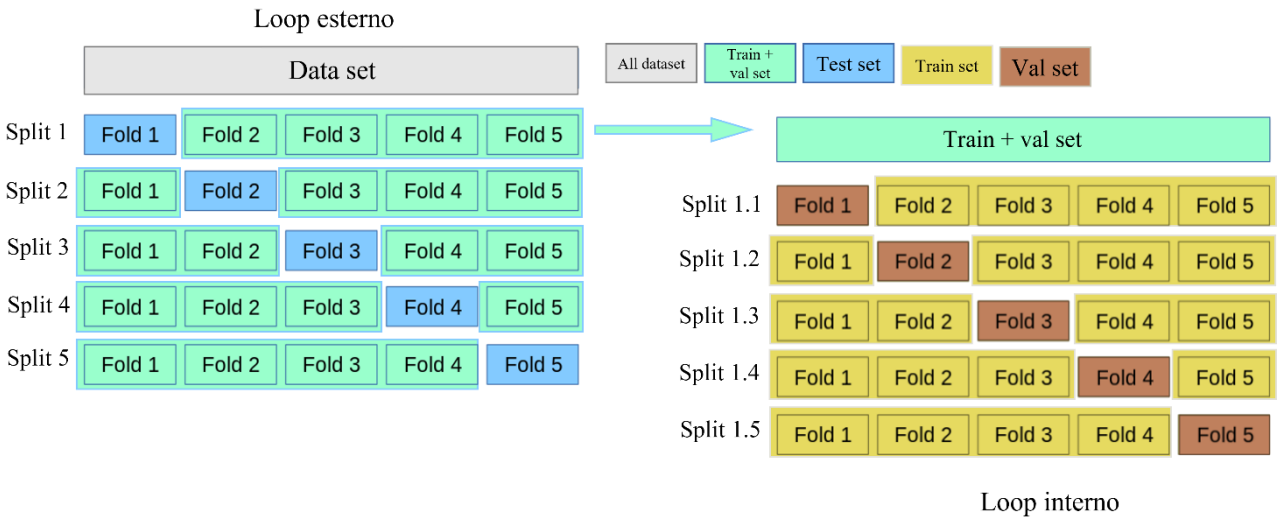


Figura 10 Schema della nested cross-validation

Bias and variance

Quando si valuta la qualità di un modello predittivo, per il suo miglioramento, è importante comprendere i concetti di bias e varianza associati al modello. Sono le due principali fonti di errore in ML; stimare la quantità di bias e varianza di un modello può aiutare a comprenderne i problemi principali e quali strategie adottare per migliorarne le prestazioni.

Il bias può essere visto come la differenza tra i valori effettivi e previsti. In altre parole, il bias è associato alle ipotesi del modello predittivo sui dati, quindi un bias elevato significa che il modello è troppo semplice e non può apprendere complessità rilevanti nei dati. Questa condizione è chiamata underfitting; l'errore sul training set può essere preso informalmente come un pregiudizio.

La varianza può essere vista come la sensibilità del modello alle fluttuazioni dei dati. È la condizione in cui il modello "memorizza" i dati di training, compreso il rumore, che porta ad una bassa capacità di generalizzazione. Questa condizione è anche chiamata overfitting; è caratterizzato da un divario

relativamente ampio tra l'errore di addestramento e l'errore di convalida; questa differenza può essere considerata come la varianza del modello.

Vale la pena considerare che la riduzione di uno di questi due errori potrebbe portare ad un aumento dell'altro; quindi, si dovrebbe trovare un equilibrio che mantenga l'errore del modello il più basso possibile. In altre parole, il modello dovrebbe essere sufficientemente complesso da catturare i modelli rilevanti nei dati, ma ignorare comunque il rumore ed essere in grado di generalizzare. (Figura 11)

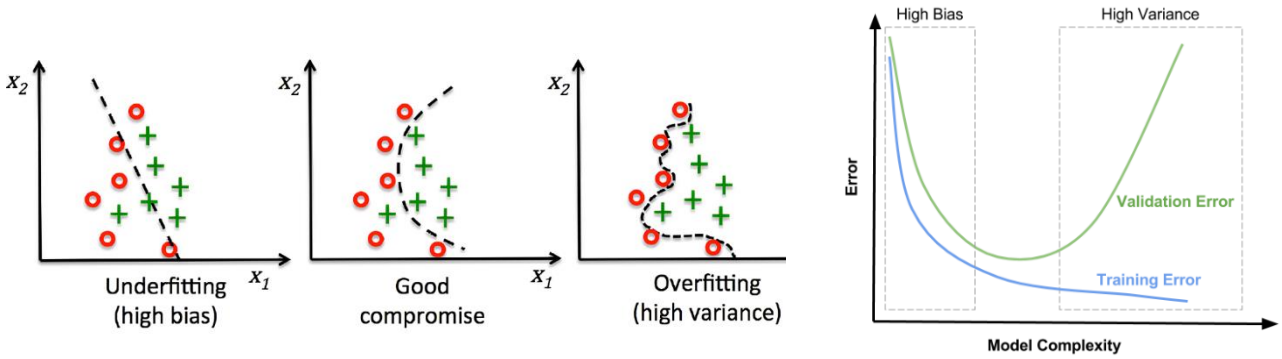


Figura 11 effetti del bias e della varianza sul confine decisionale del modello (a sinistra); effetto della complessità del modello sull'errore di test e train (a destra).

Diverse strategie possono essere adottate per ridurre il bias e la varianza del modello, per migliorare il bias gli approcci comuni includono:

- Aumentare la complessità del modello; questo viene fatto aumentando la dimensionalità dei dati, ovvero aggiungendo funzionalità, o aggiungendo termini non lineari al modello.
- Ridurre o addirittura eliminare la regolarizzazione del modello, che funziona per mantenerlo semplice.
- Analizzare i modelli di errore nei dati che portano a un errore sistematico; una volta identificato, la preelaborazione dei dati o la definizione di nuove funzionalità può aiutare a risolverlo.

D'altra parte, per migliorare la varianza del modello:

- L'aggiunta di dati di addestramento è l'approccio più semplice e affidabile, quando disponibile e se non richiede una potenza di calcolo eccessiva.
- Aggiunta o aumento della regolarizzazione, che riduce i parametri del modello per scartare alcuni termini per semplificarlo.
- Selezione delle caratteristiche: riducendo la dimensionalità dei dati si riducono anche i parametri del modello, favorendone la messa a punto anche con dataset di piccole dimensioni.

È importante ricordare che i dati sono generalmente rumorosi e anche il modello migliore non fornisce una classificazione perfetta. Questo tipo di errore irriducibile viene anche chiamato errore ottimo di Bayes ed è definito come l'errore della funzione migliore che mappa i campioni di input alle loro classi di appartenenza. Tale errore può essere considerato come l'obiettivo di prestazione per un determinato modello. Anche se esistono diversi approcci per stimare l'errore ottimale di Bayes, la migliore prestazione umana può essere impostata come obiettivo di prestazione.

Metriche di validazione

Una volta ottenuti diversi modelli dal train set, essi vengono valutati sul validation set. Nelle attività di classificazione, le metriche di valutazione dell'errore vengono utilizzate per quantificare la qualità del modello. Attraverso queste metriche è possibile confrontare diversi modelli, e selezionare quello più appropriato per risolvere il problema.

La matrice di confusione è la misurazione delle prestazioni più completa di un classificatore. È una tabella riepilogativa del numero di classificazioni corrette e errate prodotte da un modello in un'attività di classificazione. Le sue righe rappresentano le classi effettive, mentre le sue colonne le classi predette dal classificatore. Si tratta quindi di una matrice quadrata con numero di righe e colonne pari al numero di classi. Gli elementi diagonali rappresentano le classificazioni corrette, mentre gli elementi al di fuori della diagonale le classificazioni errate.

Consideriamo ad esempio la matrice di confusione 2×2 (figura 12) di una generica classificazione binaria, con una classe positiva (1) e una classe negativa (0):

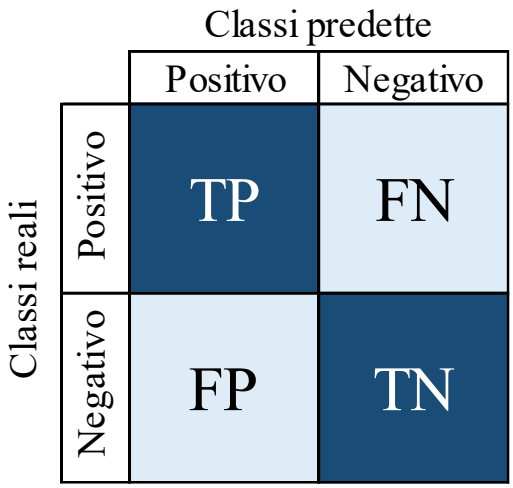


Figura 12 matrice di confusione

Possiamo definire:

True Positive (TP): numero di campioni positivi correttamente classificati.
True Negative (TN): numero di campioni negativi correttamente classificati.
False Positive (FP): numero di campioni negativi erroneamente classificati come positivi.
False Negative (FN): numero di campioni positivi classificati erroneamente come negativi.

Anche se la matrice di confusione fornisce un quadro completo delle prestazioni del modello, il confronto di diversi modelli attraverso le matrici non è pratico. L'approccio consolidato consiste nel calcolare metriche scalari dalla matrice di confusione per effettuare confronti tra modelli; i più comuni includono:

Accuratezza: definita come il rapporto tra il numero di classificazioni corrette e il numero totale di classificazioni, fornisce informazioni sulla performance globale del modello. Nonostante sia la metrica più utilizzata e molto facile da capire, non è adatta quando si tratta di compiti di classi sbilanciate, poiché potrebbe assumere grandi valori anche quando la classe minore è sempre classificata erroneamente. È stato quindi introdotta l'accuratezza bilanciata, la quale pesa il valore in base alla numerosità di ogni classe.

$$accuratezza = \frac{TP + TN}{TP + TN + FP + FN}$$

$$accuratezza\ bilanciata = \frac{(N - n_1) * TP + (N - n_2) * TN}{N * (TP + TN + FP + FN)}$$

Dove N è il numero totale di campioni, n_i è il numero di campioni appartenente alla classe *i.*

Sensitività o "recall": è definita come il rapporto tra il numero di campioni positivi correttamente classificati e il numero totale di campioni positivi. Rappresenta il vero tasso positivo. Si noti che fornisce informazioni solo sulla performance del modello sulla classe positiva.

$$sensitività\ = \frac{TP}{TP + FN}$$

Specificità: è la controparte della sensitività per la classe negativa, definita come il rapporto tra il numero di campioni negativi correttamente classificati e il numero totale di campioni negativi. Rappresenta il numero tasso di true negative.

$$specificità = \frac{TN}{TN + FP}$$

Precisione: definita come il rapporto tra il numero di campioni positivi correttamente classificati e il numero totale di classificazioni positive, indica la qualità delle previsioni positive.

$$precisione = \frac{TP}{TP + FP}$$

La sensibilità, la specificità e la precisione contengono solo informazioni su una singola classe; pertanto, sono insufficienti se usati da soli. Invece di utilizzare due o più metriche scalari per confrontare i modelli, è possibile combinarle per ottenere nuove metriche che contengono informazioni su entrambe le classi e affidabili quando si affrontano problemi di classi sbilanciate.

F1-score: è definito come la media armonica di precisione e sensitività; pertanto, penalizza qualsiasi classificazione in cui una di queste due metriche

è scadente. Il valore F1-score è preferito per confrontare diversi modelli poiché è un valore scalare.

$$F1_{score} = 2\frac{precision \times recall}{precision + recall} = \frac{2TP}{2TP + FP + FN}$$

Area sotto la curva ROC (Receiver Operating Characteristic) (AUC) (Figura 13): lo spazio ROC è bidimensionale, graficamente con il tasso di falsi positivi (1-specificità) sull'asse orizzontale e il tasso di vero positivo, o sensibilità, sull'asse verticale. Qualsiasi classificatore è mappato come un punto nello spazio ROC e la sua posizione fornisce informazioni sulla qualità della classificazione, dove il punto (0,1) identifica la classificazione perfetta e la linea diagonale qualsiasi prestazione di ipotesi casuale. Quando la classificazione è basata su una probabilità a posteriori di appartenere a una classe in un intervallo [0,1] continuo, è possibile variare la soglia di discriminazione per ottenere un numero di punti nello spazio ROC associati allo stesso classificatore in diverse condizioni operative, che generalmente mostrano prestazioni diverse. Tali punti, una volta interpolati, forniscono la cosiddetta curva ROC di un classificatore. L'area sotto la curva ROC è una metrica scalare comunemente utilizzata per confrontare i modelli, che fornisce informazioni su una performance globale del modello in diverse condizioni di discriminazione, in base sia alla sensibilità che alla specificità, favorendo quindi quei modelli che funzionano bene su entrambe le classi.

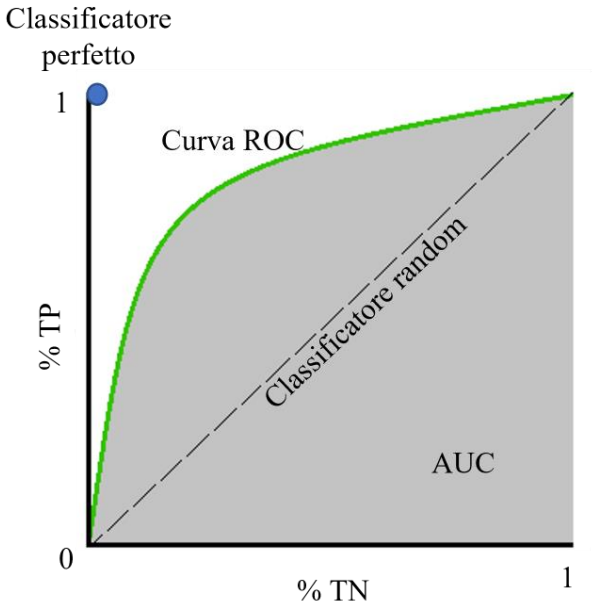


Figura 13 Curva ROC

Nel momento in cui si possiede un numero elevato di campioni si può pensare di utilizzare una doppia soglia, le probabilità a posteriori che ne ricadono all'interno di queste non vengono considerate per le metriche. In questo modo si riduce il numero di campioni classificati, ma l'accuratezza su quelli classificati sale. Al variare delle soglie si possono ottenere diversi punti come mostrato in figura 14.

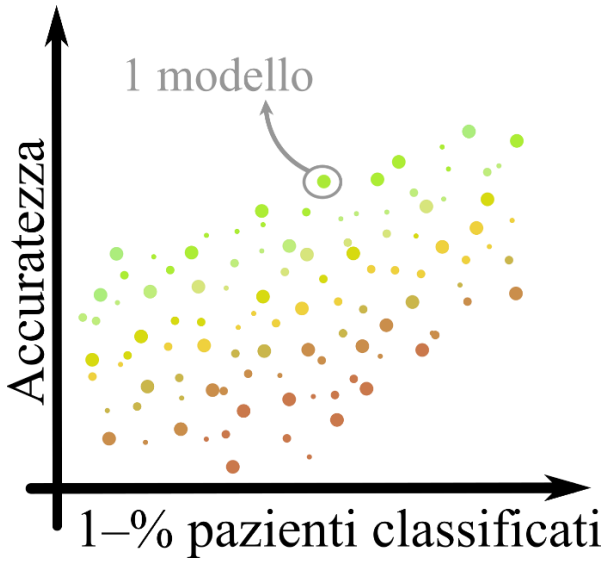


Figura 14 grafico a dispersione di tutti i modelli creati, la linea identifica i modelli migliori selezionati per l'utilizzo.

Sulle ascisse si ha la percentuale di pazienti classificati, mentre sulle ordinate l'accuratezza bilanciata corrispondente. I punti di coordinate [0,1] rappresentano un modello con una classificazione perfetta in cui si ha il 100% di accuratezza su tutti i pazienti, a scalare sulla diagonale i modelli sempre peggiori. La qualità dei modelli viene rappresentata dalla scala cromatica. Allo scalare della percentuale di pazienti classificati sono stati considerati i modelli con accuratezza migliore, essi sono stati uniti con una linea per identificarli. Per quanto riguarda il "problema dei non classificati", in questo studio essi sono stati sottoposti a un modello immediatamente successivo avente la percentuale di classificazione superiore e un'accuratezza inferiore. Il loop procede fino a quando non si trova un modello, con i relativi parametri, che riesca a classificare il campione.

Creazione modelli

Smote

Quando si tratta di classi sbilanciate, la maggior parte dei classificatori ML tende a classificare solamente le classi maggioritarie, avendo prestazioni significativamente peggiori sulle classi minori. L'approccio comune è quello di introdurre alcuni metodi di bilanciamento che possono migliorare la qualità del modello. Le soluzioni principali prevedono la modifica del modello stesso o del dataset. In questo lavoro vengono adottati due metodi distinti:

- Viene utilizzata l'accuratezza bilanciata sulla numerosità delle classi come metrica per valutare la scelta degli iperparametri.
- L'uso di SMOTE (Synthetic Minority Oversampling Technique), ovvero un metodo per il ricampionamento dei dati.

Il primo approccio consiste nel modificare la funzione di costo del classificatore. In pratica, utilizzando l'accuratezza bilanciata come metodo per valutare gli iperparametri del modello attraverso la sua massimizzazione, si riesce a fare in modo che il costo di errata classificazione di ciascuna classe sia inversamente correlato alla numerosità della stessa. Si prevede pertanto

che il modello migliori la sua capacità di classificare le classi meno rappresentate.

Questo primo metodo di bilanciamento agisce direttamente sul modello modificandone la funzione di costo. Invece, la seconda tecnica agisce solo sul set di dati.

L'algoritmo SMOTE esegue l'aumento dei dati creando campioni sintetici basati su campioni originali. Di solito è preferito rispetto a tecniche più tradizionali come il sotto campionamento, che trascura una percentuale di dati preziosi, e il sovra campionamento, che genera semplicemente duplicati. SMOTE può essere visto come una versione avanzata del sovra campionamento in cui i campioni generati sono leggermente diversi da quelli originali. Questo algoritmo funziona come segue:

Un campione di una classe di minoranza viene scelto casualmente. Vengono identificati i *k*-elementi più vicini al campione appartenenti alla stessa classe. Uno viene scelto casualmente e viene calcolato il vettore tra il campione originale e il vicino selezionato. Tale vettore viene moltiplicato per un valore casuale compreso tra 0 e 1, e aggiunto al campione originale, per ottenere il campione sintetico. Questa procedura viene ripetuta per tutte le classi minoritarie fino a quando tutte le classi hanno la stessa numerosità.

Questa procedura consiste essenzialmente nello spostare il punto dato nella direzione del suo vicino; questo fa sì che i campioni sintetici non siano copie esatte degli originali, ma anche non troppo differenti. In questo lavoro, l'algoritmo SMOTE è stato utilizzato sulle tre classi minori in modo che tutte le classi abbiano raggiunto la stessa numerosità. Per una valutazione più veritiera del classificatore, i campioni sintetici sono stati utilizzati solo durante la fase di addestramento del modello. L'algoritmo non viene applicato per il set di test, nel quale vengono utilizzati i campioni originali.

Si presenta ora una breve descrizione dei modelli di ML utilizzati.

Classificatore di regressione logistica (LOG)

La regressione logistica, nonostante il suo nome, è un modello lineare per la classificazione piuttosto che la regressione. In questo modello, le probabilità

che descrivono i possibili risultati di un singolo studio sono modellate utilizzando una funzione logistica. Come problema di ottimizzazione, la regressione logistica penalizzata della classe binaria riduce al minimo la seguente funzione di costo:

$$f(w,c) = \min \left[\frac{1}{2}w^T\mathrm{w} + \mathrm{C}\sum_{i=1}^{n}\log\left(e^{-y_i(X_i^T w+c)}+1\right)\right]$$

La regolarizzazione elastic-net usata in questo studio è una combinazione di l1 e l2 essi sono controllati dal parametro ρ, e riduce al minimo la seguente funzione di costo:

$$f(w,c) = \min \left[\frac{1-\rho}{2}w^T\mathrm{w} + \rho||w||_1 + \mathrm{C}\sum_{i=1}^{n}\log\left(e^{-y_i(X_i^T w+c)}+1\right)\right]$$

dove ρ controlla la forza della regolarizzazione. Si noti che, in questa notazione, si presume che il target assuma valori -1, 1 allo step *i*. Esistono diversi risolutori che permettono queste ottimizzazioni i principali presi in considerazione sono:

Linear Discriminant Analysis (LDA)

Questi classificatori sono interessanti perché hanno soluzioni a forma chiusa che possono essere facilmente calcolate. Il modello LDA [39], [40] presume che gli input per ogni classe condividano la stessa matrice di covarianza [41] : $\Sigma_k = \Sigma$ per tutti k, ovvero siano condizionatamente indipendenti in ciascuna classe. Ciò riduce il logaritmo delle probabilità a posteriori a:

$$logP(y=k|x) \;=\; -\frac{1}{2}(x-\mu_k)^t\sum^{-1}(x-\mu_k) + logP(y=k) + Cst$$

Il termine $(x-\mu_k)^t\sum^{-1}(x-\mu_k)$ corrisponde alla distanza di Mahalanobis tra il campione x e la media μ_k. La distanza di Mahalanobis indica quanto x è vicino a μ_k, tenendo anche conto della varianza di ciascuna caratteristica. Possiamo quindi interpretare LDA come l'assegnazione di x alla classe la cui media è la più vicina in termini di distanza di Mahalanobis, tenendo conto anche delle probabilità a priori della classe. Il logaritmo della probabilità a posteriore può anche essere scritto come:

$$logP(y = k|x) = w_k^t x + w_{k0} + Cst$$

Dove $w_k = \sum^{-1} \mu_k$ e $w_{k0} = -\frac{1}{2}\mu_k^t \sum^{-1} \mu_k + logP(y = k)$. Queste quantità corrispondono rispettivamente agli attributi e ai coefficienti intercettanti. Dalla formula di cui sopra, è chiaro che LDA ha una superficie decisionale lineare.

Esistono diversi tipi di risolutori:

Il risolutore 'svd' per evitare di calcolare la sommatoria calcola due SVD (singular value decomposition): l'SVD della matrice di input centrata X e l'SVD dei vettori medi di classe.

Il risolutore 'lsqr' è un algoritmo efficiente che funziona solo per la classificazione. Deve calcolare esplicitamente la matrice di covarianza e supporta il restringimento e gli stimatori di covarianza personalizzati. Questo risolutore calcola i coefficienti $w_k = \sum^{-1} \mu_k$ risolvendo per $\Sigma\omega = \mu_k$, evitando così il calcolo esplicito dell'inverso $\sum^{-1}$.

Il risolutore "eigen" si basa sull'ottimizzazione del rapporto di dispersione tra le classi all'interno del rapporto di dispersione delle classi. Tuttavia, il risolutore deve calcolare la matrice di covarianza, quindi potrebbe non essere adatto a situazioni con un numero elevato di funzionalità.

Support vector machine (SVM)

La support vector machine (SVM) costruisce un iperpiano o un insieme di iperpiani in uno spazio dimensionale alto o infinito, che può essere utilizzato per la classificazione, la regressione o altre attività. Intuitivamente, una buona separazione è ottenuta dall'iperpiano che ha la maggiore distanza dai punti dati di addestramento più vicini di qualsiasi classe, il cosiddetto margine funzionale (figura 15), poiché in generale maggiore è il margine minore è l'errore di generalizzazione del classificatore. La figura seguente mostra la funzione decisionale per un problema separabile linearmente, con tre campioni sui limiti del margine, chiamati "vettori di supporto":

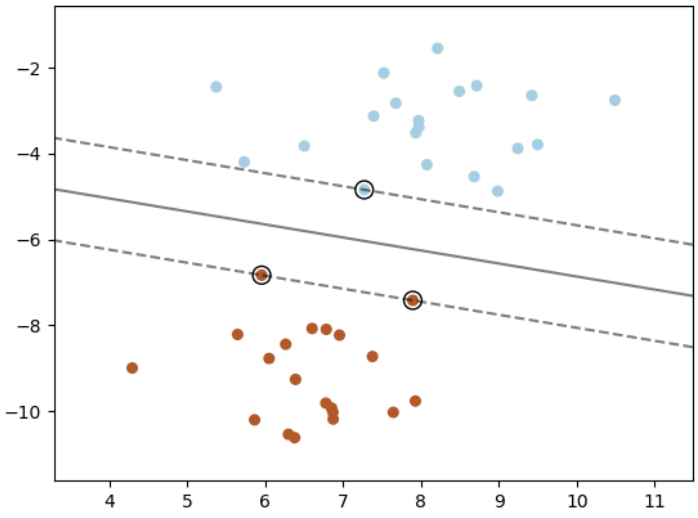


Figura 15 Schema SVM di un problema bidimensionale

Intuitivamente, si sta cercando di massimizzare il margine, incorrendo in una penalità quando un campione viene classificato in modo errato o all'interno del limite del margine. I problemi di solito non sono sempre perfettamente separabili con un iperpiano; quindi, si permette ad alcuni campioni di essere a una distanza dal loro corretto limite di margine. Il termine di penalità C controlla la forza di questa penalità e, di conseguenza, funge da parametro di regolarizzazione inversa. Una volta risolto il problema di ottimizzazione, l'output della predizione per un determinato campione diventa:

$$\sum_{i \in SV} y_i \alpha_i K(x_i, x) + b$$

e la classe prevista corrisponde al suo segno. Rimane solo da sommare i vettori di supporto (cioè i campioni che si trovano all'interno del margine) perché i doppi coefficienti α_i sono zero per gli altri campioni.

K-nearest neighbors (KNN)

La classificazione basata su neighbors (vicinati) è un tipo di apprendimento basato su istanza o apprendimento non generalizzante, non tenta di costruire un modello interno generale, ma memorizza semplicemente le istanze dei dati di training. La classificazione viene calcolata da un voto a maggioranza semplice dei valori di train più vicini di ciascun punto, a un punto viene

assegnata la classe che ha il maggior numero di rappresentanti all'interno dei neighbors più vicini al punto. In alcune circostanze, è meglio pesare i voti in modo tale che i neighbors più vicini contribuiscano maggiormente all'adattamento. Le distanze possono essere calcolate con diversi algoritmi:

L'implementazione di ricerca vicina più ingenua coinvolge il calcolo a forza bruta delle distanze tra tutte le coppie di punti nel set di dati: per *N* campioni in *D* dimensioni, questo approccio scala come $O[DN^2]$. Le ricerche efficienti dei vicini di forza bruta possono essere molto competitive per i piccoli campioni di dati. Tuttavia, man mano che il numero di campioni cresce, l'approccio a forza bruta diventa rapidamente irrealizzabile.

Per affrontare le inefficienze computazionali dell'approccio della forza bruta, sono state inventate una varietà di strutture di dati basate su alberi. In generale, queste strutture tentano di ridurre il numero richiesto di calcoli di distanza codificando in modo efficiente le informazioni aggregate sulla distanza per il campione. L'idea di base è che se il punto A è molto distante dal punto B e il punto C è molto vicino al punto B, allora i punti A e C sono molto distanti, senza dover calcolare esplicitamente la loro distanza. In questo modo, il costo computazionale di una ricerca dei neighbors più vicini può essere ridotto a $O[\log(N)]$.

L'albero KD è una struttura ad albero binaria che partiziona ricorsivamente lo spazio dei parametri lungo gli assi, dividendolo in regioni ortotropiche nidificate in cui sono archiviati i punti. La costruzione di un albero KD è molto veloce: poiché il partizionamento viene eseguito solo lungo gli assi, non è necessario calcolare distanze D-dimensionali.

Per affrontare le inefficienze di KD Trees in dimensioni superiori, è stata sviluppata la struttura dei dati dell'albero della palla. Dove gli alberi KD partizionano i dati lungo gli assi cartesiani, gli alberi a sfera partizionano i dati in una serie di ipersfere nidificanti. Ciò rende la costruzione dell'albero più costosa di quella dell'albero KD, ma si traduce in una struttura dati che può essere molto efficiente su dati altamente strutturati, anche in dimensioni molto elevate. Un albero a sfera divide ricorsivamente i dati in nodi definiti da un centroide *C* e da un raggio *r*, in modo tale che ogni punto del nodo si

trovi all'interno dell'ipersfera definita da *C* e *r*. Il numero di punti candidati per una ricerca vicina è ridotto attraverso l'uso della disuguaglianza del triangolo:

$$|x + y| \leq |x| + |y|$$

Con questa configurazione, un singolo calcolo della distanza tra un punto di prova e il centroide è sufficiente per determinare un limite inferiore e superiore sulla distanza da tutti i punti all'interno del nodo. A causa della geometria sferica dei nodi dell'albero a sfera, può superare le prestazioni di un albero KD in dimensioni elevate, sebbene le prestazioni effettive dipendano fortemente dalla struttura dei dati di allenamento.

Come notato in precedenza, per campioni di piccole dimensioni una ricerca a forza bruta può essere più efficiente di una query ad albero. Questo fatto è spiegato nell'albero della palla e nell'albero KD passando internamente alle ricerche di forza bruta all'interno dei nodi foglia. Con l'iperparametro leaf_size (dimensione del nodo) si possono produrre diversi effetti:

Un leaf_size grande porta a un tempo di costruzione dell'albero più veloce, perché è necessario creare meno nodi. Con l'aumentare del leaf_size, la memoria necessaria per memorizzare una struttura ad albero diminuisce.

Alberi decisionali

Gli alberi decisionali sono un metodo di apprendimento supervisionato non parametrico utilizzato per la classificazione e la regressione. L'obiettivo è creare un modello che preveda il valore di una variabile target imparando semplici regole decisionali dedotte dalle caratteristiche dei dati (figura 16). Un albero può essere visto come un'approssimazione costante. Dati i vettori di addestramento $x_i \in R^n$,e un vettore di classi $y \in R^i$, un albero decisionale partiziona ricorsivamente lo spazio delle feature in modo tale che i campioni con le stesse etichette o valori target simili siano raggruppati insieme. Lascia che i dati al nodo m siano rappresentati da Q_m con n_m esempi. Per ogni divisione candidata costituita da una feature j e una soglia t_m, partizionare i dati nei sottoinsiemi $Q_m^{left}(\theta)$ e $Q_m^{right}(\theta)$:

$$Q_m^{left}(\theta) = \{(x, y) | x_j \leq t_m\}$$

$$Q_m^{right}(\theta) = {Q_m}/{Q_m^{left}(\theta)}$$

La qualità di una divisione candidata del nodo viene quindi calcolata utilizzando una funzione di impurità o funzione di perdita $H()$, la cui scelta dipende dal compito da risolvere (classificazione o regressione)

$$G(Q_m, \theta) = \frac{n_m^{left}}{n_m} H\left(Q_m^{left}(\theta)\right) + \frac{n_m^{right}}{n_m} H\left(Q_m^{right}(\theta)\right)$$

Se un target è un risultato di una classificazione allora può assume valori 0, 1, ..., K-1, per il nodo *m*. Sia:

$$p_{mk} = \frac{1}{n_m} \sum_{y \in Q_m} I(y = k))$$

la proporzione di osservazioni di classe *k* nel nodo *m*. Se *m* è un nodo terminale, il valore della probabilità a posteriori della predizione per questa area è impostato su p_{mk}. Le misure comuni di impurità sono le seguenti:

Gini:

$$H(Q_m) = \sum_k p_{mk}(1 - p_{mk})$$

Perdita di log o entropia:

$$H(Q_m) = -\sum_k p_{mk}\, log(p_{mk})$$

Alcuni vantaggi degli alberi decisionali sono:

- Semplice da capire e da interpretare. Gli alberi possono essere visualizzati.
- Richiede poca preparazione dei dati. Altre tecniche spesso richiedono la normalizzazione dei dati, è necessario creare variabili fittizie e rimuovere valori vuoti. Si noti tuttavia che questo modulo non supporta i valori mancanti.

- In grado di gestire dati sia numerici che categorici. Tuttavia, l'implementazione di scikit-learn non supporta le variabili categoriche per ora.
- In grado di gestire problemi multi-output.
- Utilizza un modello a scatola bianca. Se una data situazione è osservabile in un modello, la spiegazione della condizione è facilmente spiegabile dalla logica dicotomica. Al contrario, in un modello di scatola nera (ad esempio, in una rete neurale artificiale), i risultati possono essere più difficili da interpretare.

Gli svantaggi degli alberi decisionali includono:

- I modelli ad albero decisionale sono molto propensi all'overfitting. Meccanismi come la potatura, l'impostazione del numero minimo di campioni richiesti in un nodo fogliare o l'impostazione della profondità massima dell'albero sono necessari per evitare questo problema.
- Gli alberi decisionali possono essere instabili perché piccole variazioni nei dati potrebbero comportare la generazione di un albero completamente diverso. Questo problema viene mitigato utilizzando alberi decisionali all'interno di un insieme.

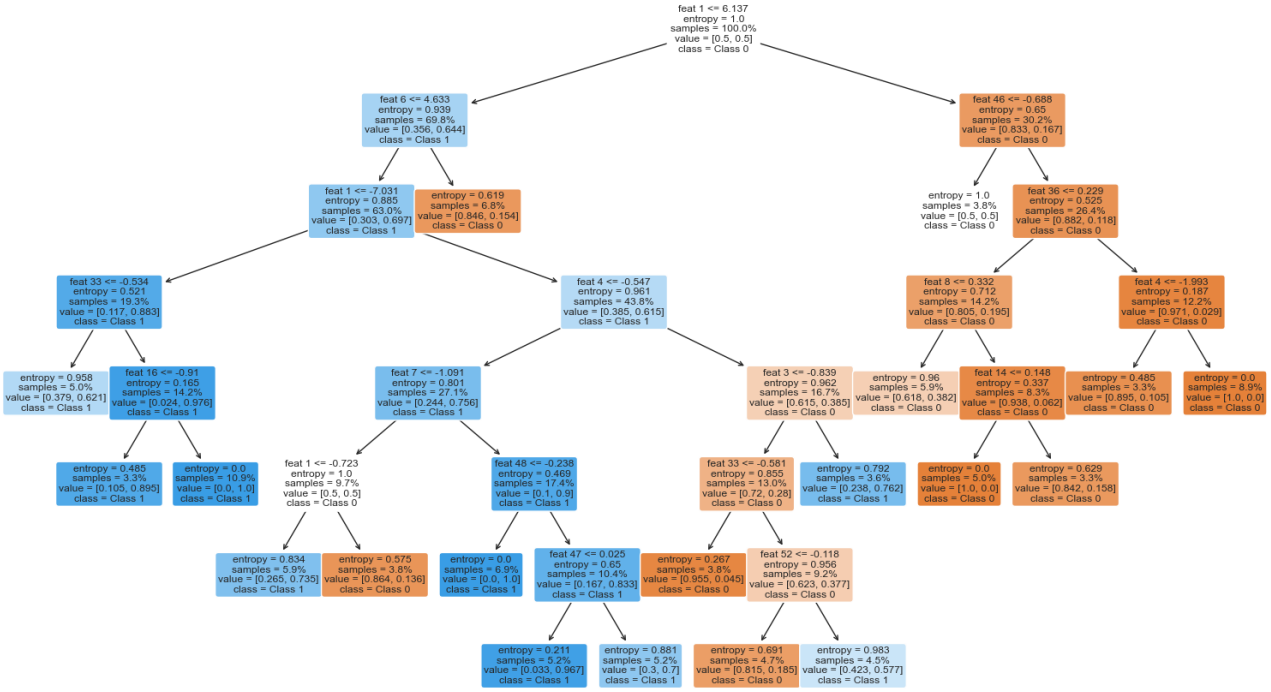


Figura 16 Schema del modello di un albero decisionale con parametri di p-val 0.01, mutua correlazione 0.95 e varianza spiegata dalla pca 0.95%. effettuata la pca le features sono diventate una combinazione lineari delle precedenti, quindi sono state solo numerate

Random Forest (RF)

Una random forest (RF) è un insieme diversificato di classificatori ad albero, introducendo la casualità nella costruzione dei singoli classificatori. Ogni albero dell'insieme viene creato da un campione estratto con la sostituzione (ad esempio, un campione di bootstrap) dal set di allenamento. Inoltre, quando si divide ogni nodo durante la costruzione di un albero, la migliore divisione si trova da tutte le funzionalità di input o da un sottoinsieme casuale di dimensioni max_features. Lo scopo di queste due fonti di casualità è quello di diminuire la varianza dello stimatore della foresta. In effetti, gli alberi decisionali individuali in genere mostrano un'elevata varianza e tendono a sovradimensionarsi. La casualità iniettata nelle foreste produce alberi decisionali con errori di previsione in qualche modo disaccoppiati. Prendendo una media di queste previsioni, alcuni errori possono annullarsi. [42], [43]

I parametri principali da regolare quando si utilizzano questi metodi sono n_estimators e max_features. Il primo è il numero di alberi nella foresta. Più grande è, meglio è, ma anche più tempo ci vorrà per calcolare. Inoltre, nota che i risultati smetteranno di migliorare in modo significativo oltre un numero critico di alberi. Il secondo è la dimensione dei sottoinsiemi casuali di funzionalità da considerare quando si divide un nodo. Più bassa è maggiore è la riduzione della varianza, ma anche maggiore è l'aumento della distorsione. Si può usare anche il parametro max_depth Per controllare la profondità degli alberi e min_samples_split per controllare il minimo numero di valori su un nodo per dividere in due rami.

AdaBoost (AB)

Il principio fondamentale di AdaBoost è quello di adattare una sequenza di modelli deboli (i.e. *weak learners*). Questi sono modelli che leggermente migliori dell'ipotesi casuale e vengono addestrati su versioni ripetutamente modificate dei dati. Le previsioni di tutti loro vengono quindi combinate attraverso un voto a maggioranza ponderata (o somma) per produrre la previsione finale. Le modifiche dei dati ad ogni cosiddetta iterazione di potenziamento consistono nell'applicazione di pesi $w_1, w_2, \ldots, w_N$ a ciascuno dei campioni di allenamento. Inizialmente, questi pesi sono tutti impostati su $1/N$, in modo che il primo passo alleni semplicemente un modello debole sui dati originali. Per ogni iterazione successiva, i pesi del campione vengono modificati individualmente e l'algoritmo di apprendimento viene riapplicato ai dati riponderati. Diversamente da un semplice raggruppamento di alberi (e.g. Random Forest), man mano che le iterazioni procedono, gli esempi difficili da prevedere ricevono un'influenza sempre crescente. Ogni successivo modello debole è quindi costretto a concentrarsi sugli esempi che mancano ai precedenti nella sequenza [44].

Multi-Layer Perceptron (MLP)

Multi-Layer Perceptron (MLP) è un algoritmo di apprendimento supervisionato che apprende una funzione addestrandosi su un train set, dove m è il numero di dimensioni per l'input ed o è il numero di dimensioni per l'output (figura 17). Dato un insieme di caratteristiche e un target, il modello riesce ad adattarsi grazie a approssimatori di funzioni non lineari per un

problema di classificazione o regressione. È diverso dalla regressione logistica, in quanto tra il livello di input e quello di output, chiamati strati nascosti.

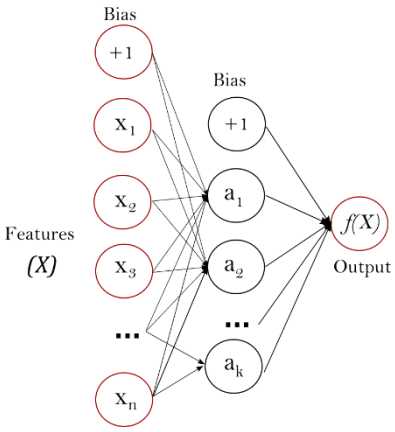


Figura 17 Schema di un multi layer percepron, a sinistra il layer di input al quale seguono i layer nascori e infine sulla destra quello di output.

Lo strato più a sinistra, noto come livello di input, è costituito da un insieme di neuroni che rappresentano le caratteristiche di input. Ogni neurone nello strato nascosto trasforma i valori dello strato precedente con una sommatoria lineare ponderata, seguita da una funzione di attivazione non lineare. Il livello di output riceve i valori dall'ultimo livello nascosto e li trasforma in valori di output. Le più comuni funzioni di attivazione dei neuroni possono essere sigmoide, tangente iperbolica e ReLU (Rectified Linear Unit) (figura 18), con le seguenti formulazioni.

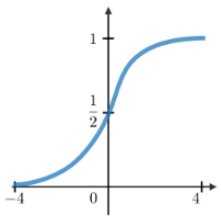
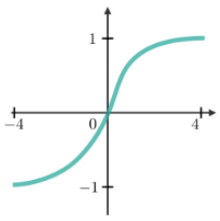
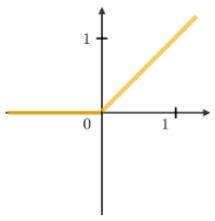
Sigmoid	Tanh	RELU
$g(z) = \frac{1}{1+e^{-z}}$	$g(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	$g(z) = \max(0, z)$
		

Figura 18 Funzioni di attivazione dei neuroni.

L'algoritmo per aggiornare i pesi si basa sul back-propagation, calcola il gradiente della funzione di Loss rispetto ai pesi della rete per un singolo esempio di input-output, aggiorna i pesi minimando l'errore in modo efficiente. MLP utilizza diverse funzioni di perdita a seconda del tipo di problema. La funzione di perdita per la classificazione è l'entropia incrociata media (Average Cross-Entropy), che in caso binario è data come:

$$Loss(\gamma, y, W) = -\frac{1}{n}\sum_{i=0}^{n}(y_i \ln(\gamma_i) + (1 - y_i)\ln(1 - \gamma_i)) + \frac{\alpha}{2n}||W||_2^2$$

Dove $\alpha||W||_2^2$è un termine di regolarizzazione L2 (aka penalità) che penalizza modelli complessi, e α è un iperparametro non negativo che controlla l'entità della penalità. Si può utilizzare il parametro α per la regolarizzazione (L2), questo aiuta a evitare l'overfitting penalizzando i pesi con valori elevati.

I vantaggi dei Multi-layer Perceptron sono:

- Capacità di apprendere modelli non lineari.

Gli svantaggi del MLP includono:

- MLP con livelli nascosti ha una funzione di perdita non convessa dove esiste più di un minimo locale. Pertanto, diverse inizializzazioni casuali del peso possono portare a una diversa accuratezza di convalida.
- Richiede l'ottimizzazione di una serie di iperparametri come il numero di neuroni, livelli e iterazioni nascosti.
- È sensibile al ridimensionamento delle features.

Votazione (media)

Sono stati aggiunti due modelli di *voting* in cui sono state valutate la moda delle classificazioni e la media delle probabilità a posteriori dei modelli sopra citati. Per entrambi i classificatori sono stati rimossi i modelli che non sono riusciti ad adattarsi bene col problema. Si è riscontrato che la media risulta spesso migliore di tutti gli altri, perché va a compensare le classificazioni dubbie (probabilità a posteriori da 0.4 a 0.6) con le opinioni degli altri classificatori.

Ottimizzazione parametri

I modelli sopra descritti contengono molti iperparametri, questi definiscono uno spazio *N* dimensionale all'interno del quale i modelli possono essere più o meno performanti. Per poter scegliere in modo opportuno gli iperparametri in questo studio si è usata la libreria *optuna,* la quale procedendo per dimezzazioni successive all'interno di un range predefinito estrae il valore ottimale di parametri dato un numero di iterazioni. Maggiore è il numero delle iterazioni più preciso sarà il risultato. Nella figura 19 si può notare con una scala cromatica il livello di accuratezza raggiunto dal tentativo. Le linee permettono di identificare un singolo set di iperparametri identificato con l'intercetta sui relativi segmenti. Per iperparametri categorici il costo computazionale sale poiché il numero di tentativi stabilito a priori viene diviso equamente tra le opzioni categoriche, le variabili di tipo float e intere invece ottengono un risultato basato per dimezzazioni successive.

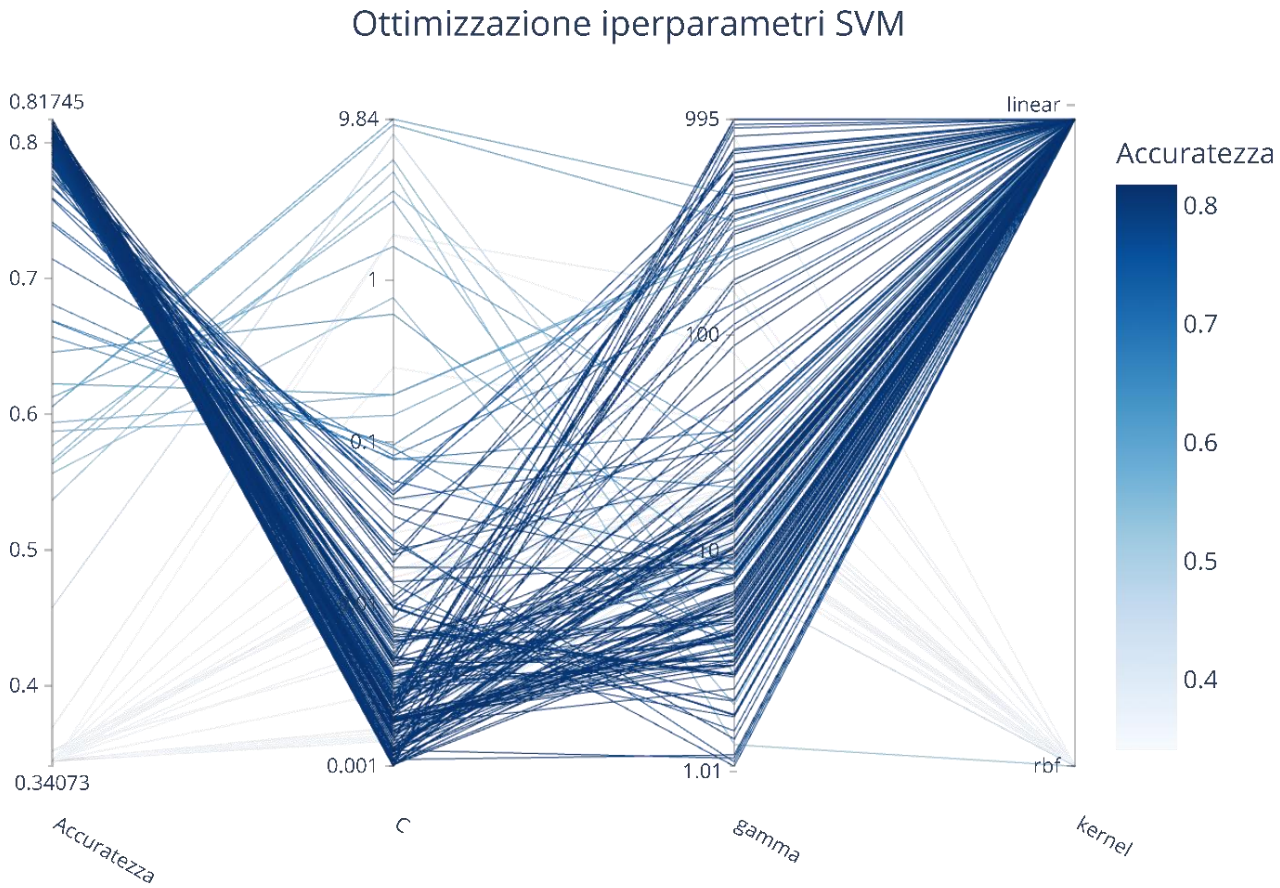


Figura 19 Visualizzazione della ricerca degli iperparametri del modello SVM tramite libreria optuna

La nested cross validazione impone la ricerca degli iperparametri attraverso il loop interno, i modelli con i valori di prova vengono allenati sul train, valutati sul val set e in base alla valutazione, *optuna* varia il set di iperparametri in modo da ottimizzare questa valutazione. Sono stati valutati i risultati utilizzando l'accuratezza, l'accuratezza bilanciata e lì'f1-score come metriche di validazione dell'errore, e il metodo di ottimizzazione procede con la massimizzazione di queste funzioni.

Ogni set di iperparametri è stato ottimizzato sul dataset di addestramento e di validazione uniti, questi vengono infine testati sul test set ottenendo così una valutazione di quanto il modello è stato in grado di generalizzare e valutare dei campioni mai visti prima. Al termine della cross-validazione si ottiene un modello per ogni fold esterna con i relativi set di iperparametri dei modelli appartenenti a quella fold.

Interpretazione risultati

Uno dei principali motivi per i quali si è preferito usare questi modelli e questo processo di validazione dei risultati è il fatto che si possa dare una motivazione della classificazione. In altri casi, per esempio quando si utilizzano CNN è difficile capire quali sono state le feature più significative per il modello e poter dare una spiegazione della classificazione, perché le feature vengono codificate nei livelli più bassi in altre feature senza possibilità di interpretazione umana. Nei modelli utilizzati, invece, si può valutare come varia la valutazione finale al variare delle feature di input. Utilizzando la libreria SHAP sui modelli migliori ottenuti si è andato a valutare quali feature influenzassero maggiormente la classificazione sbilanciandola verso una classe o nell'altra. Nella figura 20 si può valutare graficamente che al variare delle feature di input codificate con una scala cromatica si ottiene il relativo impatto sull'output del modello sulle ascisse. Un esempio di interpretazione del grafico è valutare che all'aumentare del valore della prima componente principale (PCA 0) il modello tende a diminuire la sua probabilità a posteriori e quindi a sbilanciare la classificazione verso la classe zero. Al contrario se la PCA 40 aumenta questa sbilancia positivamente la classe di output.

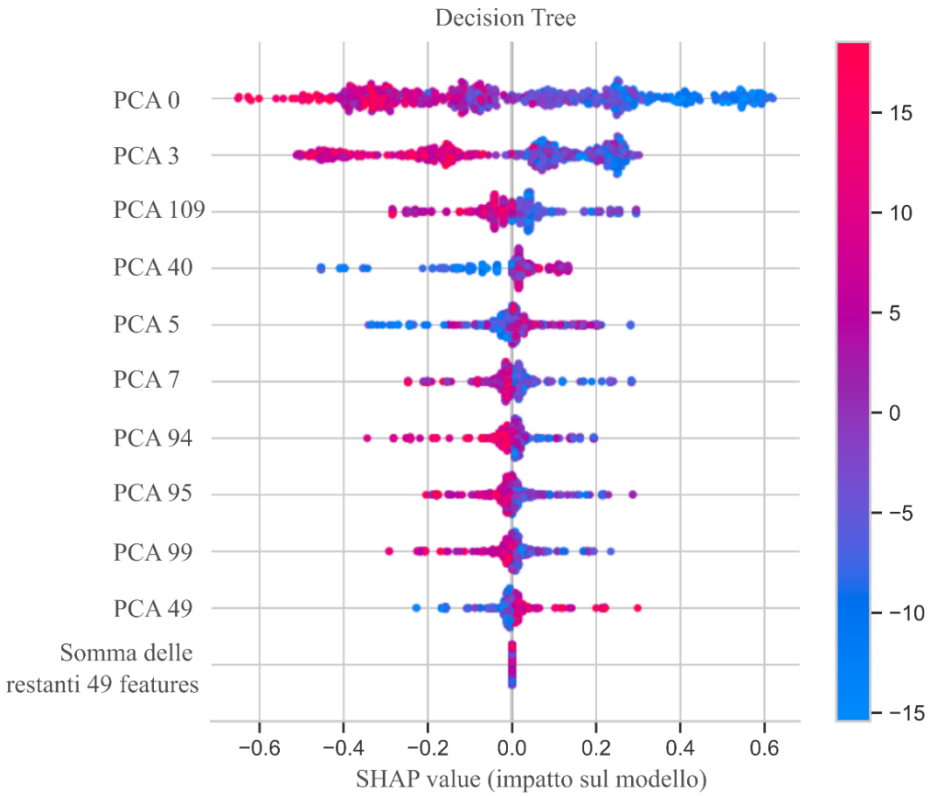


Figura 20 Impatto della variazione delle feature di input sull'output. Il modello preso in considerazione è in albero decisionale. La scala cromatica rappresenta la variazione delle feature di input, mentre la posizione sulle ascisse il relativo impatto sull'output.

Nel caso in cui il modello migliore fosse il modello media, si effettua la valutazione SHAP per tutti i modelli e si trova l'impatto medio sull'output. Si può notare come le feature analizzate non sono quelle estratte poiché nel processo di feature selection sono state create le componenti principali come combinazione lineare delle feature estratte. Quindi si è riportato nella figura 21 la composizione delle prime 5 componenti principali ripotate nella figura precedente. Sono stati rappresentati i primi 20 pesi delle feature estratte per la composizione delle componenti principali messi in ordine di importanza, ovvero con il peso maggiore in valore assoluto. La codifica colore rappresenta quali influenzano positivamente, in verde, e quali negativamente, in rosso.

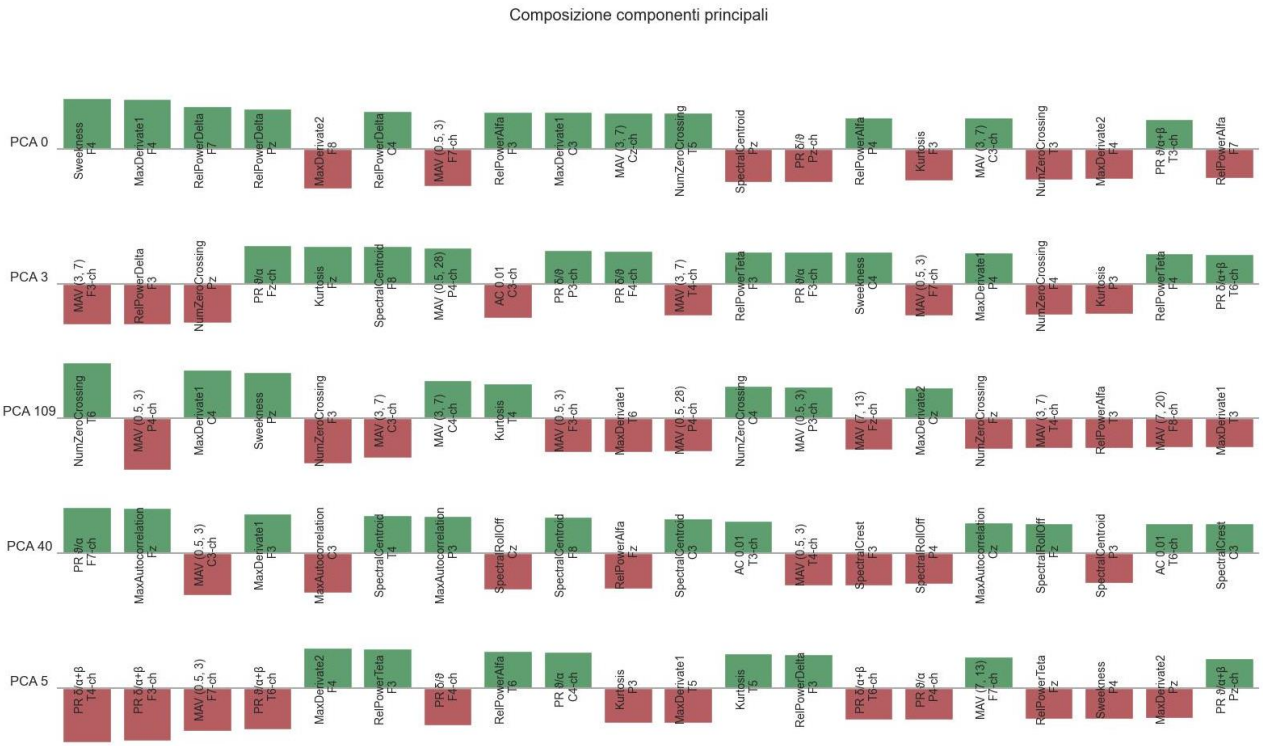


Figura 21 Composizioni delle prime 5 componenti principali, in rosso se il contributo è negativo, in verde se positivo. Sono state riportate le prime 20 feature in ordine di importanza.

A questo punto si può valutare quante feature tra quelle estratte sono state considerate in più componenti principali e si può creare un istogramma con questa informazione per ogni descrittore considerato. Si ricorda che sono state estratte le feature per ogni canale di prelievo o per gruppi di canali o con diversi parametri di analisi; quindi, nell'istogramma riportato in figura 22 si riporta la percentuale di feature rimaste dopo la procedura di feature selection. Esso rappresenta le feature più significative secondo i test statistici e che sono il più possibile scorrelate l'una dall'altra.

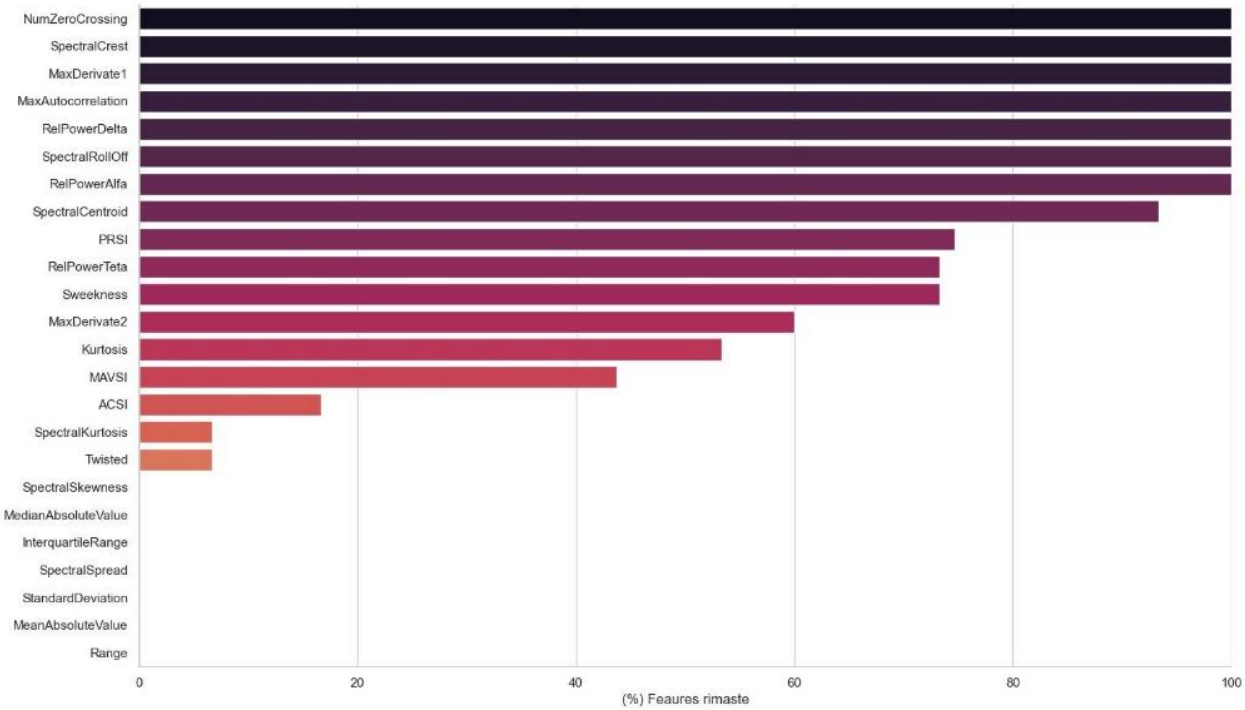


Figura 22 istogramma della percentuale di feature rimaste dopo la feature selection per il modello migliore considerato in ogni descrittore

Si riporta in appendice tutte le figure rappresentanti l'interpretazione dei modelli, la composizione delle composizioni delle componenti principali e l'istogramma delle feature rimaste dopo la procedura di feature selection.

Interfaccia grafica

Per fini pratici è stato scelto di creare un'interfaccia grafica che permetta di caricare una nuova registrazione EEG e di valutarlo secondo i modelli migliori fino ad ora trovati. Siccome il campione da testare viene fornito esternamente si può effettuare la scelta del modello migliore in base ai risultati ottenuti sul test set, in modo da utilizzare il modello che sia in grado di generalizzare maggiormente il problema posto.

I modelli di predizione restituiscono una certa probabilità a posteriori di appartenenza a una classe, essa è compresa tra 0 e 1. Applicando una soglia si possono dividere i pazienti nelle due classi e al variare di quest'ultima si

ottengono diversi valori di sensibilità e specificità che possono essere espressi tramite la curva ROC. Utilizzando due soglie, invece, si divide la scala in tre sezioni, una sopra la soglia maggiore, una in mezzo alle due soglie e una inferiore alla soglia più bassa. Se una predizione ottiene come risultato della classificazione una probabilità a posteriori che rientra nel range della prima sezione, a esso viene assegnata la classe uno. Se la probabilità a posteriori ricade nella terza sezione, quella più inferiore, allora a quella classificazione verrà assegnata la classe zero. In questa implementazione pratica si è deciso di applicare la soglia inferiore in un range da 0.3 a 0.5 con passo di 0.02 e la soglia superiore in un range da 0.5 a 0.7 con passo di 0.02. Questo ha portato a ottenere cento valutazioni per uno stesso modello.

Lo script di test parte dal modello migliore, con la massima accuratezza trovata, e tenta di classificare il nuovo paziente. Se la probabilità a posteriori risultante, per quel paziente, si troverà nell'intervallo all'interno delle due soglie allora la sua classificazione non sarà ritenuta valida. In quest'ultimo caso si procederebbe a testarlo con il modello successivo dalla lista di 30 modelli per ogni predittore fino ad ottenere un valore accettabile per la classificazione. Siccome l'ultimo modello della lista classifica utilizzando entrambe le soglie a 0.5, il paziente avrà sempre una classificazione al limite con l'ultimo modello.

L'interfaccia grafica è stata creata in locale con l'intento di poter creare una web app e poterla rendere pubblica.

Risultati

Descrizione dataset

Sono stati inclusi nello studio 314 pazienti (164 femmine, 110 maschi) con 63 anni di media (range 21.7), tra questi ci sono 129 pazienti EMCS, 77 MCS e 38 UWS. Le cause che hanno portato i pazienti nello stato di GCA sono principalmente: 27% traumatiche, 5.9% Anossiche, 17.2% ischemiche, 41.4% emorragiche, 0.8% infettive, 4.7% neoplastiche, 3.1% atre cause. Il valore mediano della CRS è di 18 (IQR = 14). I predittori dell'ACNS che sono stati refertati sono stati riportati nella seguente tabella 1:

Frequenza	Delta: 385, Teta: 258, Alpha: 4
Voltaggio	Normale: 73, Ipovoltato: 568, Soppresso: 6
Simmetria	Simmetrici: 351, Asimmetrici: 296
Gradiente ap	Diretto: 498, Assente: 99, Inverso: 1, Dubbio: 49
Anomalie lente delta	Presenti: 375, Assenti: 272
Grafoelementi epilettici	Presenti: 164, Assenti: 483

Tabella 1 Refertazione dei pazienti

A causa dei pochi casi in cui la frequenza si presenta in banda alpha queste persone verranno escluse dall'allenamento dei modelli, lo stesso ragionamento vale per chi ha il voltaggio soppresso e il gradiente antero-posteriore inverso. Anche le refertazioni non chiare e dubbie sono state rimosse dal dataset per allenare i modelli, questo si presenta nel gradiente antero-posteriore.

In totale il numero delle feature estratte per ogni descrittore sono mostrate nella tabella 2:

Frequenza	1155
Voltaggio	1155
Simmetria	924
Gradiente antero-posteriore	1232
Anomalie lente delta	1155
Grafoelementi epilettici	1155

Tabella 2 Elenco del numero totale di feature estratte

Con i test statistici quelle nel dominio della frequenza risultano essere le più significative. Di seguito, nella figura 23 si riporta l'istogramma delle features estratte, esso rappresenta quante feature risultano in uno specifico range p-value.

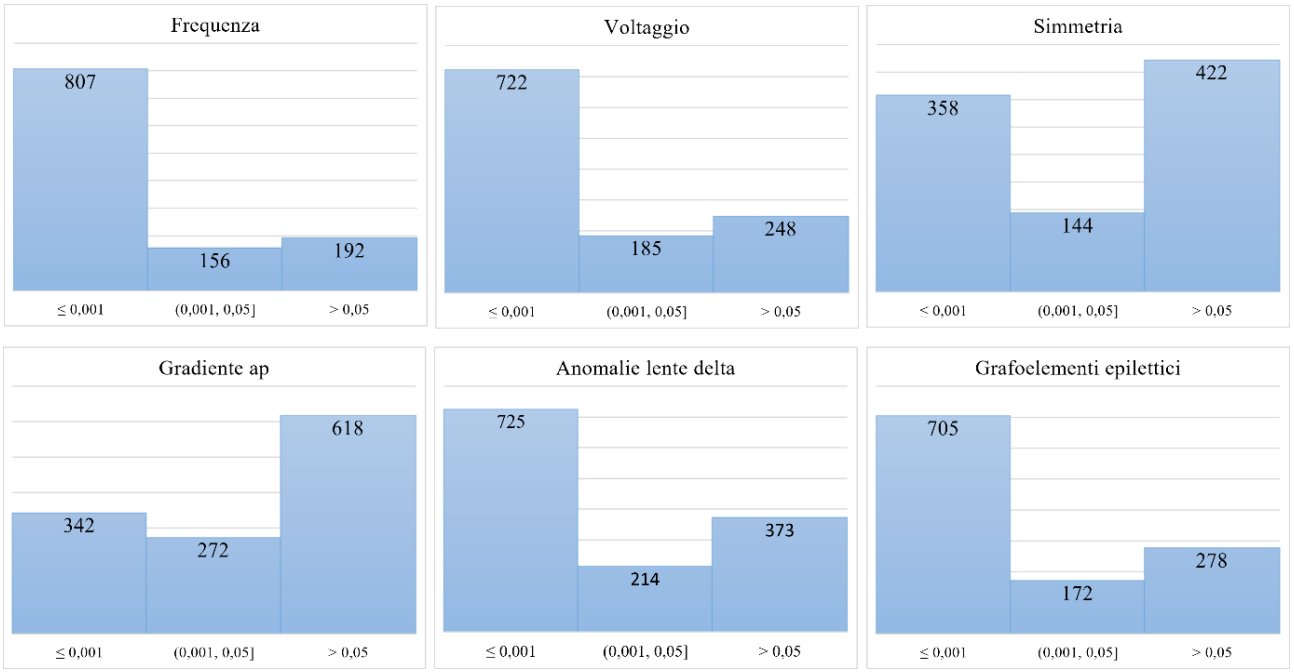


Figura 23 Numero di feature nel relativo range di p-value per ogni descrittore

Le soluzioni migliori sono quelle che hanno il maggior numero di feature aventi p-value < 0.01. Esse hanno più probabilità di aver incapsulato in quelle feature la complessità del problema e di conseguenza i modelli basati su quelle feature hanno la possibilità di generalizzare il problema posto.

Sono state eseguite diverse strategie di feature selection, ognuna delle quali ha prodotto un modello che è stato infine validato in modo da poter scegliere per ogni descrittore la combinazione più appropriata. Sulle feature, è stata eseguita la feature selection per cross correlazione con le soglie pari a 99% e 95%. Per ognuno di questi tentativi, segue la feature selection tramite test statistico con soglie pari a 0.05 e 0.01. Infine, è stata eseguita la PCA per rendere le feature estratte ortogonali tra loro, per ognuna delle combinazioni precedenti sono state impostate diverse soglie per la varianza spiegata dalla pca, esse sono 99.9%, 99%, 95%. Per la ricerca degli iperparametri si sono

impostati 3 metodi per valutare l'errore di classificazione e poter scegliere il modello migliore, essi sono l'accuratezza bilanciata, l'accuratezza e l'f1-score. Sono stati creati 10 topologie di modelli di classificazione, quindi il totale di modelli creati per ogni descrittore è 360, tra essi viene scelto quello con l'accuratezza bilanciata maggiore.

Tutti i modelli sono stati validati con la nested cross-validation aventi 5 fold esterne; pertanto, otterremo 5 modelli ottimizzati ognuno sul proprio dataset e testati sulla restante parte. Le predizioni sui test set dei diversi split vengono infine concatenate per creare un'unica matrice di confusione legata al modello.

Frequenza

Il miglior modello ottenuto per il descrittore frequenza ha ottenuto il 77.2% di accuratezza bilanciata sul test set e 85.1% sul validation set. È stato ridotto il numero di feature eseguendo le 3 tipologie di feature selection in cascata. La prima tramite mutua correlazione è stata eseguita con un valore di soglia pari al 95%. La seconda tramite test statistici è stata eseguita scartando le feature aventi p-val maggiore di 0.01. La terza è la PCA, la varianza spiegata è del 99.9%. A seguito della feature selection sono rimaste mediamente 191.4 feature per l'allenamento dei modelli, in particolare 192 nella prima fold, 184 nella seconda, 194 nella terza, 194 nella quarta, 193 nella quinta. Siccome sono stati rimossi alcuni pazienti per i motivi sopra descritti l'allenamento del modello è stata eseguita su un totale di 603 pazienti. La ricerca del modello migliore è stata effettuata utilizzando la accuratezza come metrica di valutazione dell'errore cercando di massimizzare quest'ultima durante l'ottimizzazione degli iperparametri.

Il modello utilizza un classificatore media di tutti i modelli con gli iperparametri riportati in tabella 3:

LOG

l1 ratio	C
0.127788091	0.005888421
0.081674888	0.031100064
0.022368284	0.010890307
0.026306743	0.014655809
0.012612826	0.009944311

LDA

shrinkage	solver
0.053792013	lsqr
0.196700076	lsqr
0.001283398	lsqr
0.036638718	lsqr
0.001787747	lsqr

SVM

C	gamma	kernel
0.00129369	60.58617987	linear
0.0048812	7.755346468	linear
0.004768669	8.951213705	linear
0.009197356	282.8718522	linear
0.001030076	353.4919305	linear

KNN

n neighbors	leaf size	p
6	158	2
4	121	2
6	122	2
12	218	3
10	230	3

DTC

criterion	max depth	min samples split	min samples leaf	max features	ccp alpha
entropy	10	16	8	0.887251144	0.000358599
gini	7	16	13	0.501255401	0.000214549
entropy	5	28	9	0.521301681	0.027441776
gini	3	49	6	0.852818957	0.001535223
entropy	9	15	5	0.586842093	0.00024726

RF

n estimators	criterion	Max depth	min samples split	min samples leaf	max features	ccp alpha
26	gini	10	21	7	0.523424475	0.000245374
27	entropy	9	19	8	0.598980396	0.000246921
22	entropy	9	18	15	0.57757496	0.002049486
18	entropy	9	23	6	0.5287671	0.000234892
27	gini	9	15	9	0.501266823	0.000268522

AB

criterion	max depth	min samples split	min samples leaf	max features	ccp alpha	n estimators	learning rate
entropy	10	28	11	0.9210	0.00036378	25	0.0106743
entropy	8	26	10	0.5871	0.000355336	19	0.0134408
entropy	8	39	9	0.6909	0.000221513	23	0.0165410
entropy	6	27	12	0.5989	0.000594857	23	0.0183513
entropy	9	41	18	0.7919	0.000144128	23	0.0095834

MLP

n layer	alpha	learning rate	learning rate init	power t	momentum	Hidden Layer sizes
5	0.297917531	invscaling	0.011427172	0.4492	0.050425114	[17, 17, 13, 4, 3]
5	0.17616314	adaptive	0.031018913	0.2572	0.121232079	[26, 13, 5, 3, 3]
2	0.010480549	invscaling	0.012920223	0.0103	0.010947512	[20, 5]
3	0.128541518	adaptive	0.011738478	0.1872	0.967852732	[28, 14, 10]
3	0.850547428	invscaling	0.012054451	0.0169	0.001582084	[30, 27, 14]

Tabella 3 Iperparametri di tutti i modelli considerati nella media per il descrittore frequenza

Le media delle probabilità a posteriori di questi modelli vengono mediate e infine arrotondate per ottenere la classificazione finale. Si crea infine le matrici di confusione riassuntive del modello, a sinistra quella ottenuta sul validation set, a destra quella ottenuta sul test set (figura 24).

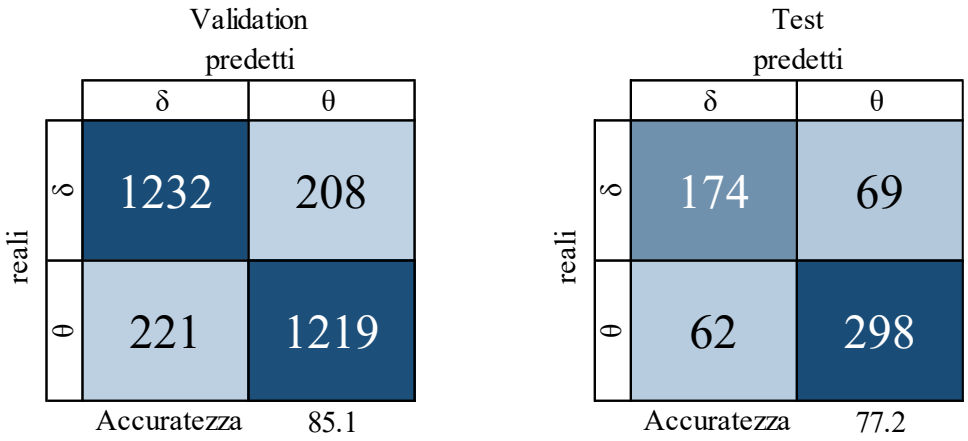


Figura 24 Matrice di confusione per il descrittore frequenza. Quella di sinistra è stata ottenuta sul dataset di validazione, si ricorda che esso viene utilizzato per l'ottimizzazione dei parametri, pertanto risulta bilanciato tra le classi. A destra la matrice di confusione ottenuta sul dataset di test.

Voltaggio

Il miglior modello ottenuto per il descrittore voltaggio ha ottenuto il 57.8% di accuratezza bilanciata sul test set e 97.5% sul validation set.

È stato ridotto il numero di feature eseguendo le 3 tipologie di feature selection in cascata. La prima tramite mutua correlazione è stata eseguita con un valore di soglia pari al 95%. La seconda tramite test statistici è stata eseguita scartando le feature aventi p-val maggiore di 0.01. La terza è la PCA, la varianza spiegata è del 99.9%. A seguito della feature selection sono rimaste mediamente 59.6 feature per l'allenamento dei modelli, in particolare 59 nella prima fold, 59 nella seconda, 58 nella terza, 63 nella quarta, 59 nella quinta. Siccome sono stati rimossi alcuni pazienti per i motivi sopra descritti l'allenamento del modello è stata eseguita su un totale di 601 pazienti. La ricerca del modello migliore è stata effettuata utilizzando la f1s come metrica di valutazione dell'errore cercando di massimizzare quest'ultima durante l'ottimizzazione degli iperparametri.

Il modello utilizza un classificatore AB di tutti i modelli con gli iperparametri riportati in tabella 4:

AB

criterion	max depth	min samples split	min samples leaf	max features	ccp alpha	n estimators	learning rate
entropy	10	21	18	0.666079	0.001727602	29	0.0402348
entropy	10	19	8	0.669402	0.000604838	23	0.0607489
entropy	9	28	10	0.731104	0.000563391	27	0.0313941
entropy	10	28	7	0.729212	0.000261632	27	0.0778597
entropy	9	17	10	0.829764	0.000390405	28	0.0408641

Tabella 4 Iperparametri del modello AdaBoost per il descrittore voltaggio

Si crea infine le matrici di confusione riassuntive del modello, a sinistra quella ottenuta sul validation set, a destra quella ottenuta sul test set (figura 25).

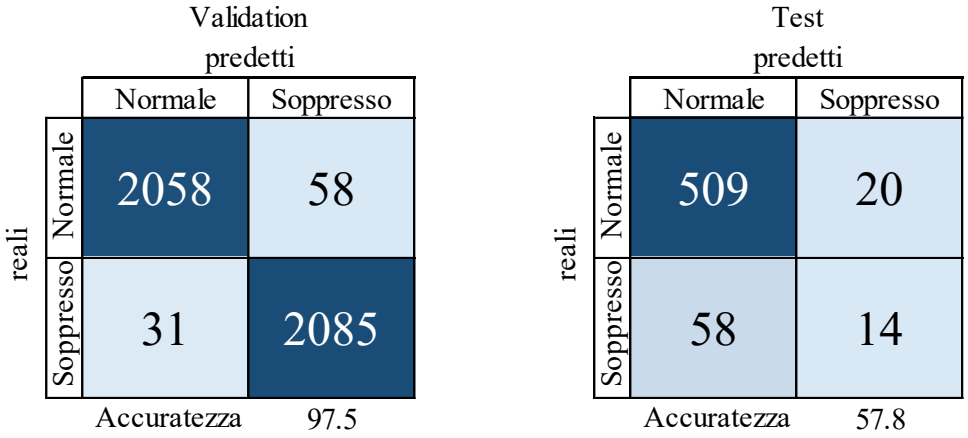


Figura 25 Matrice di confusione per il descrittore voltaggio. Quella di sinistra è stata ottenuta sul dataset di validazione, si ricorda che esso viene utilizzato per l'ottimizzazione dei parametri, pertanto risulta bilanciato tra le classi. A destra la matrice di confusione ottenuta sul dataset di test.

Simmetria

Il miglior modello ottenuto per il descrittore simmetria ha ottenuto il 75.6% di accuratezza bilanciata sul test set e 80.3% sul validation set.

È stato ridotto il numero di feature eseguendo le 3 tipologie di feature selection in cascata. La prima tramite mutua correlazione è stata eseguita con un valore di soglia pari al 99%. La seconda tramite test statistici è stata eseguita scartando le feature aventi p-val maggiore di 0.01. La terza è la PCA, la varianza spiegata è del 95%. A seguito della feature selection sono rimaste mediamente 29.6 feature per l'allenamento dei modelli, in particolare 30 nella prima fold, 29 nella seconda, 29 nella terza, 30 nella quarta, 30 nella quinta. Siccome sono stati rimossi alcuni pazienti per i motivi sopra descritti l'allenamento del modello è stata eseguita su un totale di 607 pazienti. La ricerca del modello migliore è stata effettuata utilizzando la acuratezza come metrica di valutazione dell'errore cercando di massimizzare quest'ultima durante l'ottimizzazione degli iperparametri.

Il modello utilizza un classificatore media di tutti i modelli con gli iperparametri riportati in tabella 5:

LOG

l1 ratio	C
0.074101485	0.55643548
0.453632292	0.117150758
0.562314665	0.121128567
0.059029294	0.627346188
0.280329487	0.255438909

LDA

shrinkage	solver
0.087131054	svd
0.623017692	lsqr
0.005293646	svd
0.003413664	svd
0.01915383	svd

SVM

C	gamma	kernel
0.019926	187.1903568	linear
0.007911959	5.585639287	linear
0.004427846	46.01284142	linear
0.009965715	5.618371319	linear
0.009182676	201.6592235	linear

KNN

n neighbors	leaf size	p
4	80	1
12	89	2
7	180	2
4	174	3
6	104	3

DTC

criterion	max depth	min samples split	min samples leaf	max features	ccp alpha
entropy	10	42	13	0.789044071	0.00047313
gini	3	44	17	0.902422225	0.000376054
gini	7	34	14	0.992668158	0.007155464
gini	5	46	10	0.535750577	0.000365533
entropy	4	19	11	0.684585107	0.02017338

RF

n estimators	criterion	Max depth	min samples split	min samples leaf	max features	ccp alpha
30	entropy	6	15	9	0.528088438	0.002253314
28	entropy	7	39	7	0.532990639	0.000906251
26	gini	9	18	9	0.647960711	0.000262205
30	gini	7	31	6	0.585875604	0.00297106
20	entropy	9	27	8	0.513712667	0.000304654

AB

criterion	max depth	min samples split	min samples leaf	max features	ccp alpha	n estimators	learning rate
entropy	10	18	7	0.587719	0.000142383	29	0.0103034
entropy	10	18	13	0.893311	0.007429838	30	0.0136732
entropy	10	28	11	0.731490	0.001090968	29	0.0718700
entropy	6	27	7	0.709341	0.000407602	22	0.0066121
entropy	8	30	19	0.824052	0.000498676	29	0.0269984

MLP

n layer	alpha	learning rate	learning rate init	power t	momentum	Hidden Layer sizes
3	0.056305822	constant	0.035789864	0.27830	0.207385138	[55, 16, 7]
4	0.076049819	invscaling	0.045895767	0.06012	0.009943987	[8, 7, 7, 4]
2	0.089684425	invscaling	0.031927521	0.25892	0.001427361	[38, 9]
2	0.105227715	constant	0.021852597	0.02245	0.002083739	[16, 4]
4	0.565268464	invscaling	0.040080486	0.59977	0.254601567	[29, 6, 5, 4]

Tabella 5 Iperparametri di tutti i modelli considerati nella media per il descrittore simmetria

Le media delle probabilità a posteriori di questi modelli vengono mediate e infine arrotondate per ottenere la classificazione finale. Si crea infine le matrici di confusione riassuntive del modello, a sinistra quella ottenuta sul validation set, a destra quella ottenuta sul test set (figura 26).

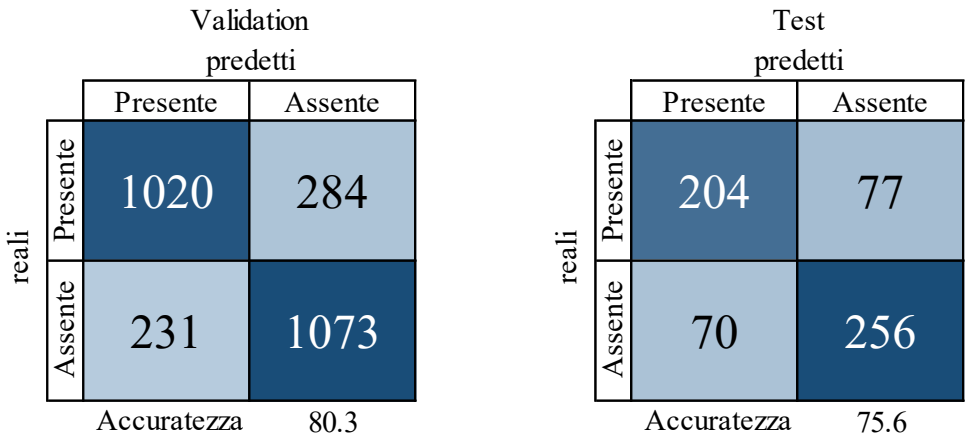


Figura 26 Matrice di confusione per il descrittore simmetria. Quella di sinistra è stata ottenuta sul dataset di validazione, si ricorda che esso viene utilizzato per l'ottimizzazione dei parametri, pertanto risulta bilanciato tra le classi. A destra la matrice di confusione ottenuta sul dataset di test.

Gradiente antero posteriore

Il miglior modello ottenuto per il descrittore gradiente antero-posteriore ha ottenuto il 57.6% di accuratezza bilanciata sul test set e 95.5% sul validation set.

È stato ridotto il numero di feature eseguendo le 3 tipologie di feature selection in cascata. La prima tramite mutua correlazione è stata eseguita con un valore di soglia pari al 99%. La seconda tramite test statistici è stata eseguita scartando le feature aventi p-val maggiore di 0.01. La terza è la PCA, la varianza spiegata è del 99.9%. A seguito della feature selection sono rimaste mediamente 93 feature per l'allenamento dei modelli, in particolare 91 nella prima fold, 95 nella seconda, 95 nella terza, 93 nella quarta, 91 nella quinta. Siccome sono stati rimossi alcuni pazienti per i motivi sopra descritti l'allenamento del modello è stata eseguita su un totale di 556 pazienti. La ricerca del modello migliore è stata effettuata utilizzando la f1s come metrica di valutazione dell'errore cercando di massimizzare quest'ultima durante l'ottimizzazione degli iperparametri.

Il modello utilizza un classificatore AB di tutti i modelli con gli iperparametri riportati in tabella 6:

AB

criterion	max depth	min samples split	min samples leaf	max features	ccp alpha	n estimators	learning rate
entropy	10	28	8	0.553470	0.000142961	28	0.0508058
entropy	9	21	9	0.560912	0.000187685	28	0.0339891
entropy	10	27	11	0.906469	0.000160444	29	0.0597406
entropy	9	37	13	0.670818	0.000698833	28	0.0715488
entropy	9	24	12	0.822178	0.000321045	29	0.0553401

Tabella 6 Iperparametri del modello AdaBoost per il descrittore gradiente antero-posteriore

Si crea infine le matrici di confusione riassuntive del modello, a sinistra quella ottenuta sul validation set, a destra quella ottenuta sul test set (figura 27).

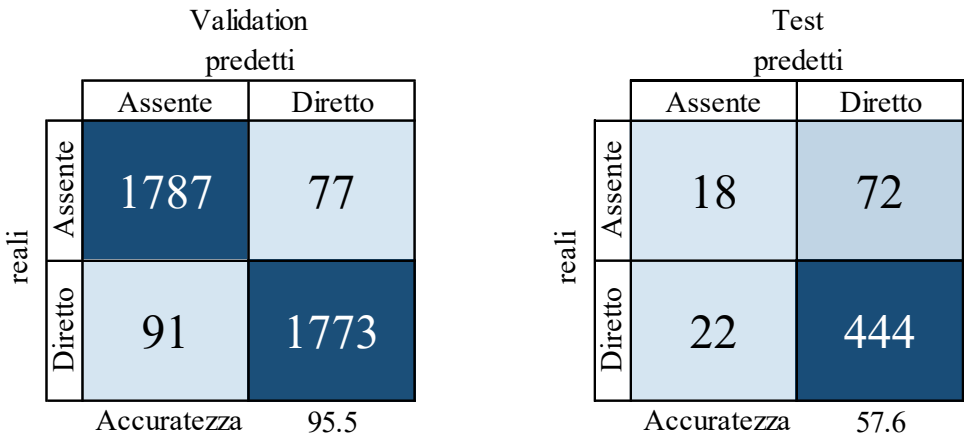


Figura 27 Matrice di confusione per il descrittore gradiente antero-posteriore. Quella di sinistra è stata ottenuta sul dataset di validazione, si ricorda che esso viene utilizzato per l'ottimizzazione dei parametri, pertanto risulta bilanciato tra le classi. A destra la matrice di confusione ottenuta sul dataset di test.

Anomalie lente delta

Il miglior modello ottenuto per il descrittore anomalie lente delta ha ottenuto il 69.6% di accuratezza bilanciata sul test set e 79.5% sul validation set.

È stato ridotto il numero di feature eseguendo le 3 tipologie di feature selection in cascata. La prima tramite mutua correlazione è stata eseguita con

un valore di soglia pari al 95%. La seconda tramite test statistici è stata eseguita scartando le feature aventi p-val maggiore di 0.05. La terza è la PCA, la varianza spiegata è del 99.9%. A seguito della feature selection sono rimaste mediamente 181.4 feature per l'allenamento dei modelli, in particolare 195 nella prima fold, 184 nella seconda, 176 nella terza, 184 nella quarta, 168 nella quinta. Siccome sono stati rimossi alcuni pazienti per i motivi sopra descritti l'allenamento del modello è stata eseguita su un totale di 590 pazienti. La ricerca del modello migliore è stata effettuata utilizzando la f1s come metrica di valutazione dell'errore cercando di massimizzare quest'ultima durante l'ottimizzazione degli iperparametri.

Il modello utilizza un classificatore AB di tutti i modelli con gli iperparametri riportati in tabella 7:

AB

criterion	max depth	min samples split	min samples leaf	max features	ccp alpha	n estimators	learning rate
entropy	8	31	5	0.704459	0.000453282	27	0.0133758
entropy	8	26	14	0.576190	0.000380527	23	0.0131333
entropy	9	24	18	0.648639	0.000305343	30	0.0207719
entropy	9	17	8	0.602505	0.003602715	23	0.0134542
entropy	7	18	11	0.845496	0.004480877	29	0.0102351

Tabella 7 Iperparametri del modello AdaBoost per il descrittore anomalie lente delta

Si crea infine le matrici di confusione riassuntive del modello, a sinistra quella ottenuta sul validation set, a destra quella ottenuta sul test set (figura 28).

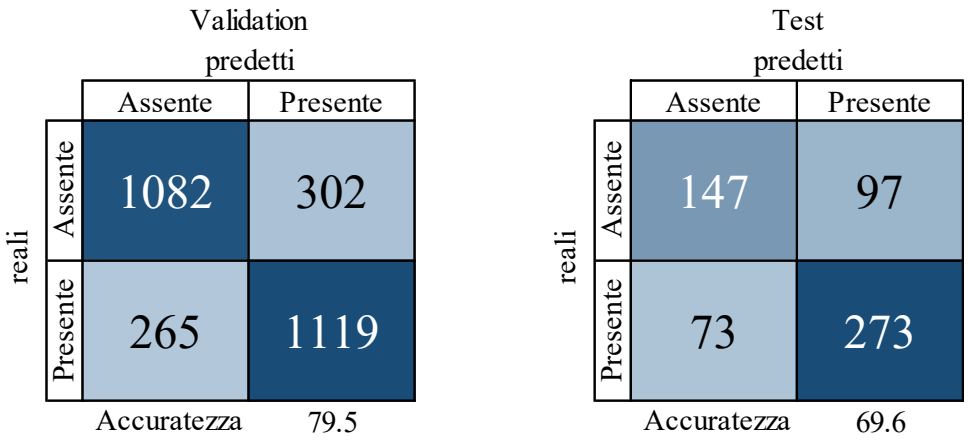


Figura 28 Matrice di confusione per il descrittore anomalie lente delta. Quella di sinistra è stata ottenuta sul dataset di validazione, si ricorda che esso viene utilizzato per l'ottimizzazione dei parametri, pertanto risulta bilanciato tra le classi. A destra la matrice di confusione ottenuta sul dataset di test.

Grafoelementi epilettici

Il miglior modello ottenuto per il descrittore grafoelementi epilettici ha ottenuto il 65.9% di accuratezza bilanciata sul test set e 93.2% sul validation set.

È stato ridotto il numero di feature eseguendo le 3 tipologie di feature selection in cascata. La prima tramite mutua correlazione è stata eseguita con un valore di soglia pari al 95%. La seconda tramite test statistici è stata eseguita scartando le feature aventi p-val maggiore di 0.05. La terza è la PCA, la varianza spiegata è del 99%. A seguito della feature selection sono rimaste mediamente 113.8 feature per l'allenamento dei modelli, in particolare 110 nella prima fold, 112 nella seconda, 122 nella terza, 113 nella quarta, 112 nella quinta. Siccome sono stati rimossi alcuni pazienti per i motivi sopra descritti l'allenamento del modello è stata eseguita su un totale di 607 pazienti. La ricerca del modello migliore è stata effettuata utilizzando la accuratezza come metrica di valutazione dell'errore cercando di massimizzare quest'ultima durante l'ottimizzazione degli iperparametri.

Il modello utilizza un classificatore AB di tutti i modelli con gli iperparametri riportati in tabella 8:

AB

criterion	max depth	min samples split	min samples leaf	max features	ccp alpha	n estimators	learning rate
entropy	9	16	7	0,895694	0,000884741	30	0,0195964
entropy	10	20	6	0,972083	0,001808831	25	0,0171864
entropy	10	26	15	0,806801	0,000234187	29	0,0795072
entropy	9	32	5	0,570160	0,003432159	23	0,0331996
entropy	10	23	9	0,521233	0,000920886	27	0,0570676

Tabella 8 Iperparametri del modello AdaBoost per il descrittore grafoelementi epilettici

Si crea infine le matrici di confusione riassuntive del modello, a sinistra quella ottenuta sul validation set, a destra quella ottenuta sul test set (figura 29).

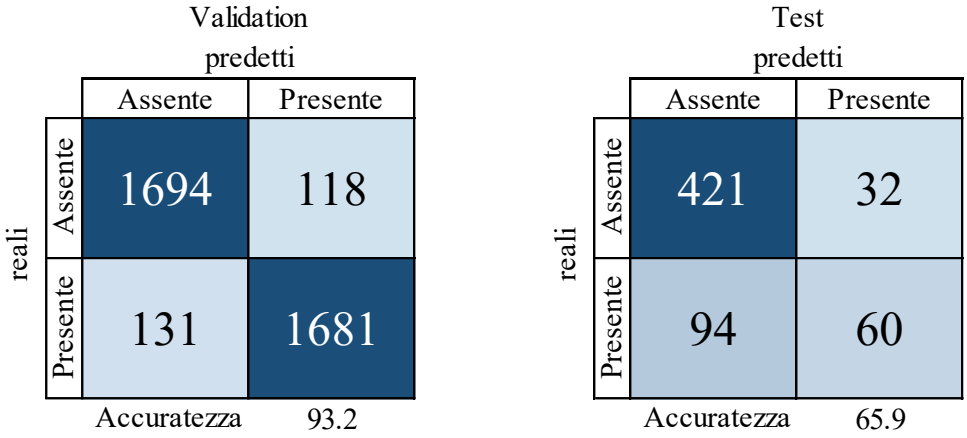


Figura 29 Matrice di confusione per il descrittore grafoelementi epilettici. Quella di sinistra è stata ottenuta sul dataset di validazione, si ricorda che esso viene utilizzato per l'ottimizzazione dei parametri, pertanto risulta bilanciato tra le classi. A destra la matrice di confusione ottenuta sul dataset di test

Interfaccia grafica

Si riporta nella figura 30 la rappresentazione dei modelli in cui ogni punto rappresenta un modello con diverse impostazioni di feature selection e di soglia applicata per la classificazione. Sulle ascisse viene riportata la percentuale di pazienti classificati, mentre sulle ordinate l'accuratezza bilanciata ottenuta sul test set.

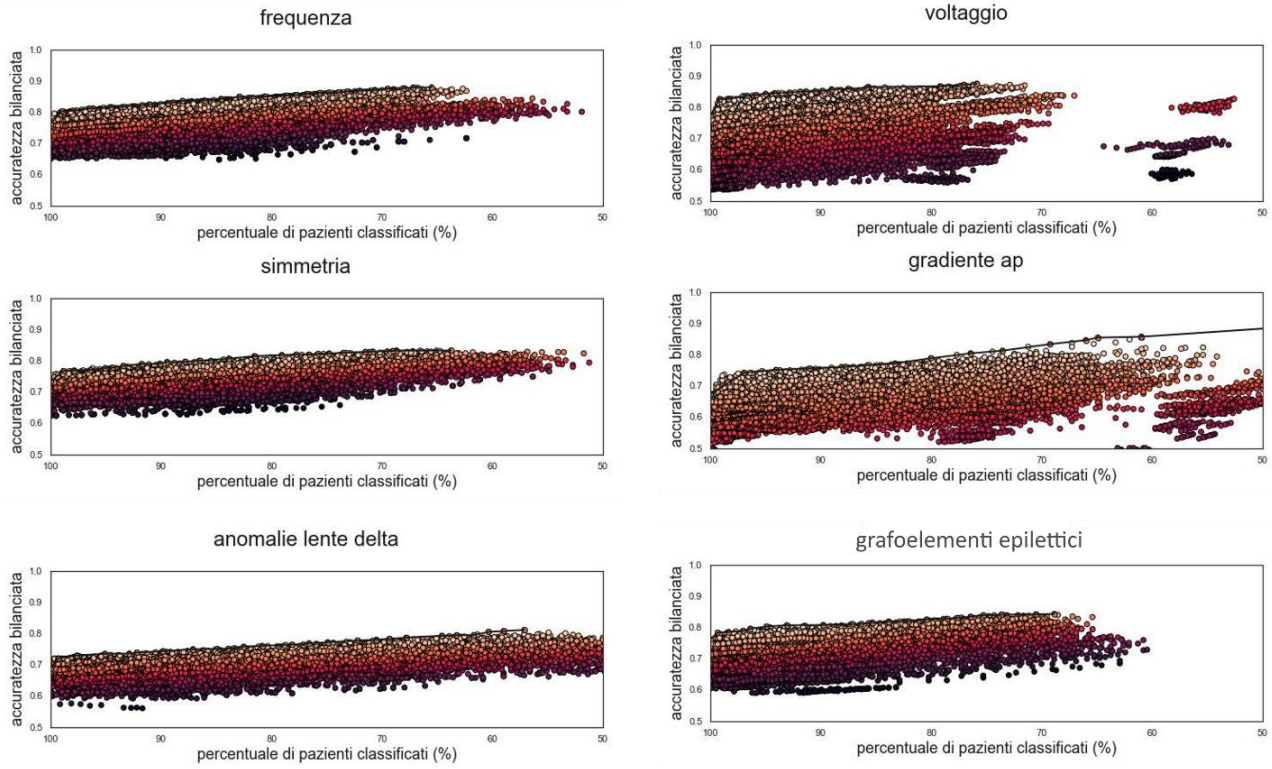


Figura 30 grafico a dispersione dei modelli creati e posizionati in base alla propria accuratezza e percentuale di pazienti classificati. La linea nera intercetta i migliori modelli selezionati per l'implementazione pratica. Per un facile confronto gli assi sono stati normalizzati tra il 50% e 100% dei pazienti classificati e il 50% e il 100% di accuratezza

In totale vengono prodotti 360 modelli ognuno valutato con coppie di soglie diverse. Nell'applicazione pratica vengono selezionati 30 modelli per ogni descrittore, dal migliore in accuratezza in modo assoluto al migliore in accuratezza avente tutti i pazienti classificati.

Essa necessita del caricamento della registrazione da parte di un operatore e in modo automatico lo analizza e lo classifica. Sulla parte di sinistra vengono

mostrati alcuni dati relativi all'EEG come frequenza di campionamento, canali presenti, data della registrazione e altri. Subito sotto invece ci sono una serie di barre progresso in cui viene mostrato cosa il software sta analizzando. Al termine dell'analisi del segnale esso verrà automaticamente classificato. Nella sezione di destra vi è una selezione opzionale del modello di classificazione, per ogni descrittore si può scegliere da una serie di 30 modelli, dal più accurato in modo assoluto, fino al modello migliore che classifica però tutti i pazienti (figura 31). Al termine dell'estrazione delle feature dal segnale verrà abilitato il pulsante valuta per classificare il segnale a partire dal modello desiderato. La scelta è opzionale, perché di default viene selezionato il modello con migliore accuratezza assoluta come punto di partenza. La rappresentazione dei risultati prevede sia il valore della predizione come testo, sia la probabilità a posteriori dei modelli alla quale il clinico può fare riferimento espressa come indice su una barra colorata. Il valore del cursore sulla barra rappresenta la probabilità a posteriori, più il cursore è vicino a un estremo più il modello è sicuro della predizione, il centro colorato in rosso rappresenta l'indecisione del modello. L'interfaccia grafica prevede inoltre delle figure topografiche del segnale in analisi, esse rappresentano con una scala cromatica la media dell'intensità del segnale per ogni canale e la potenza media in ogni banda. Se il puntatore del mouse va sopra l'icona "info" affianco a ogni valutazione compariranno i dati del modello utilizzato per la classificazione, come accuratezza bilanciata ottenuta sul test set e la percentuale di pazienti classificati (figura 32). Al termine si riportano i parametri del modello utilizzato.

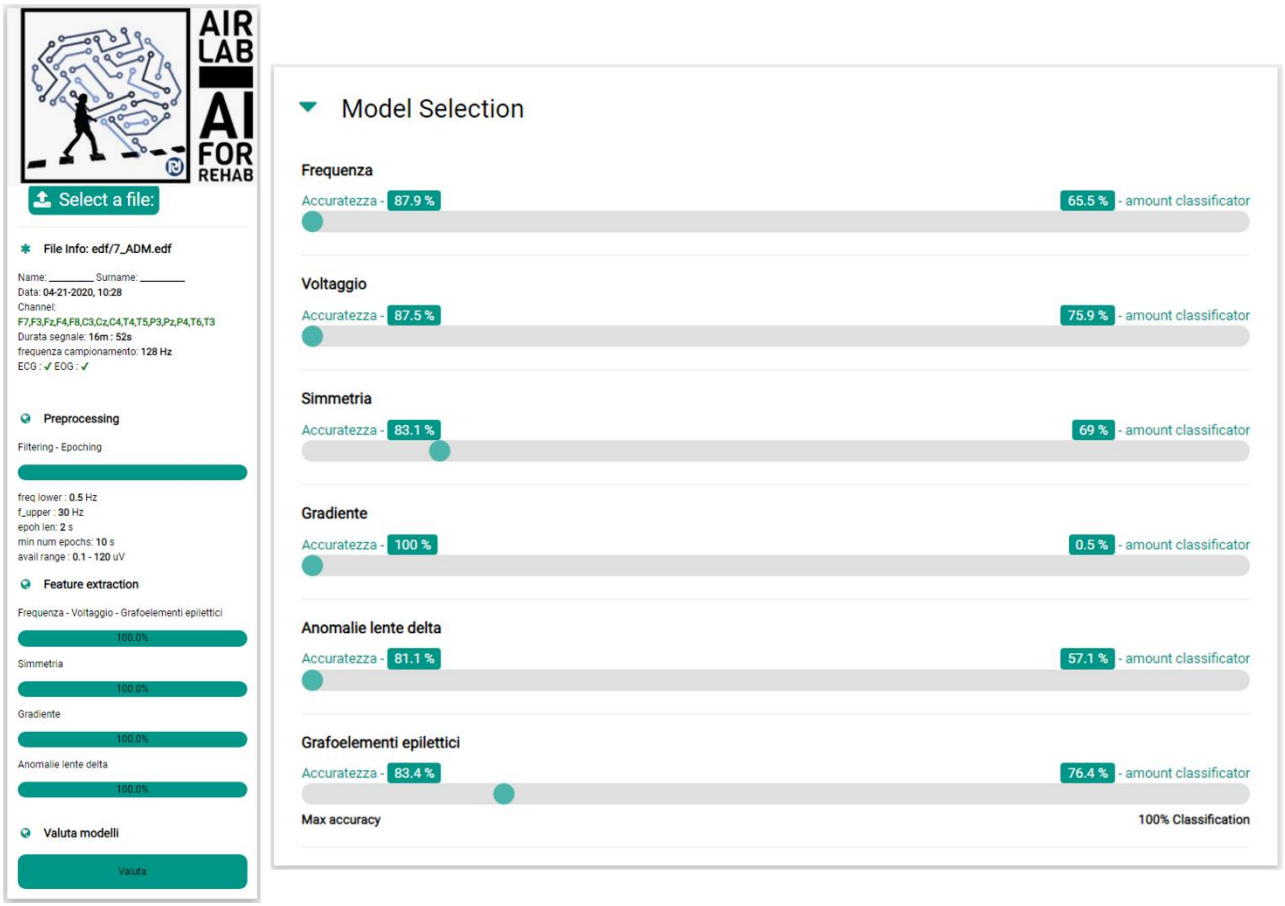


Figura 31 Pagina principale di interfaccia grafica. Sulla sinistra il caricamento del file e a scendere i dati estrapolati da esso. Sulla destra le impostazioni per la scelta del modello di ML da utilizzare. La scelta di quest'ultimo può essere lasciata alle impostazioni di default

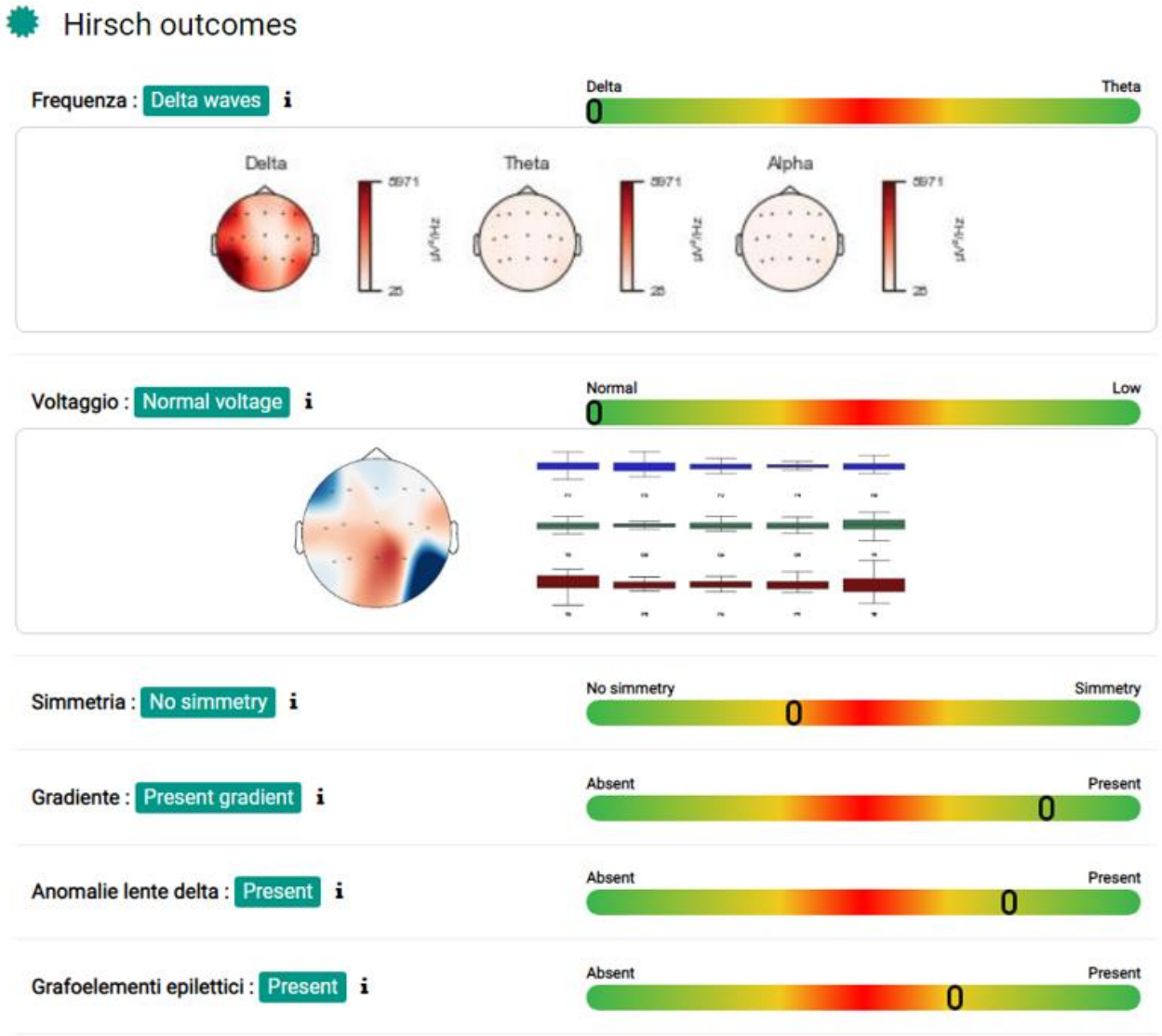


Figura 32 pagina di rappresentazioni dei risultati. Per ogni descrittore vi è il valore testuale del risultato, una barra di rappresentazione della probabilità a posteriori e alcune figure dimostrative.

Si riporta inoltre le matrici di confusione dei modelli aventi la migliore accuratezza assoluta e quelle dei modelli migliori classificando tutti i pazienti. Sotto a ogni matrice vi è riportata l'accuratezza massima e la percentuale di pazienti classificati per quel modello. Si può notare che la somma dei pazienti per le matrici con accuratezza migliore sia differente da quella che restituisce una classificazione sicura (figura 33).

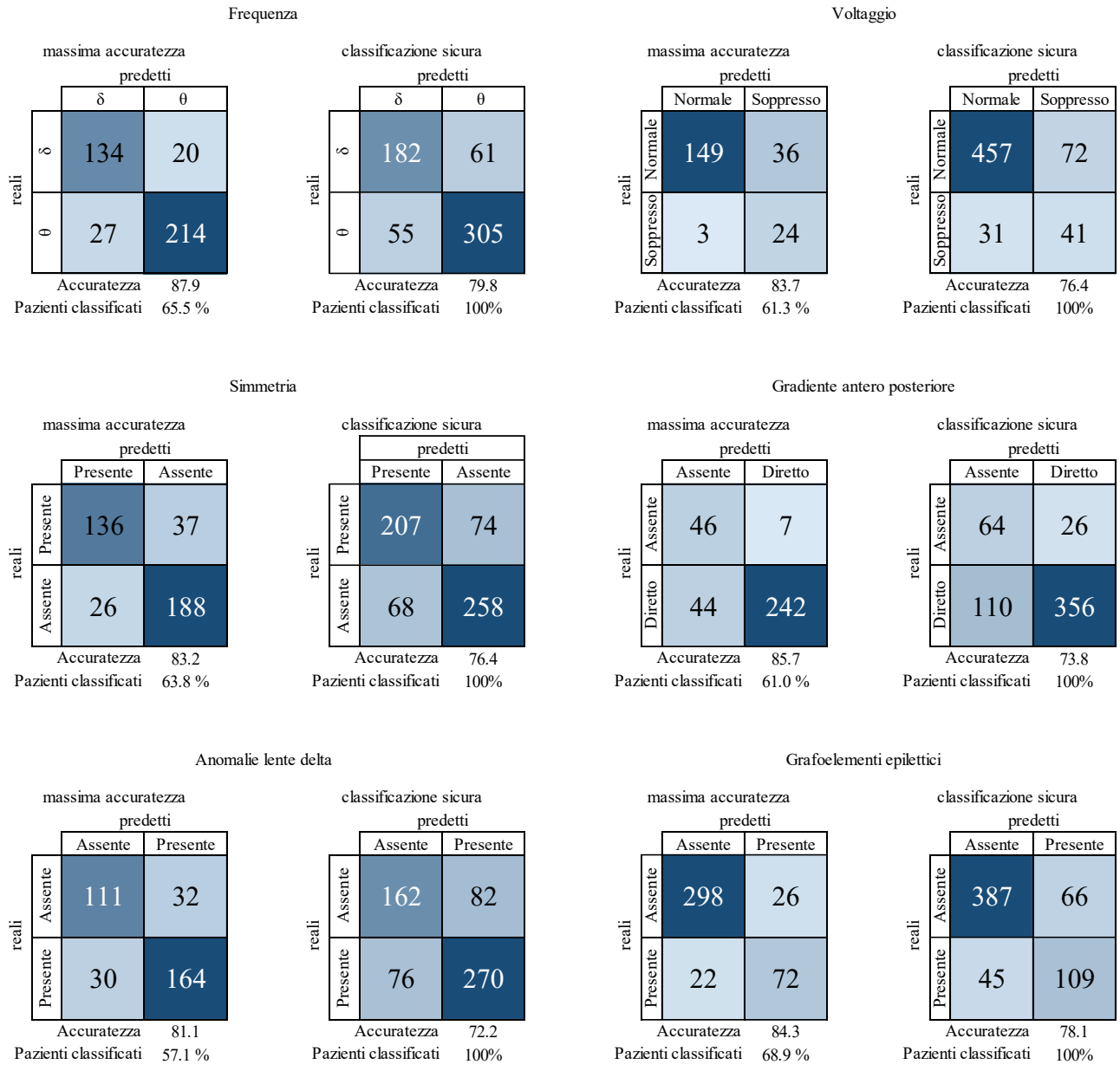


Figura 33 Matrici di confusione dei modelli implementati nell'interfaccia grafica

Discussione

Lo scopo dello studio è quello di creare un software per la refertazione automatica di alcuni predittori dell'ACNS. Gli EEG raccolti sono clinici e non a scopo di ricerca; quindi, i modelli sono stati allenati su segnali artefattati da tutti i problemi del prelievo clinico e senza la particolare attenzione della ricerca. I segnali sono stati raccolti tramite il sistema internazionale di prelievo 10-20 a 19 canali, molto comune in tutte le strutture cliniche senza aver bisogno di canali per EOG ed ECG che rendono più complessa la registrazione. Inoltre, l'analisi del segnale non prevede l'analisi delle componenti ICA del segnale, per l'interpretazione del potenziale evocato cardiaco e del battito delle ciglia, che sono interpretabili esclusivamente di clinici esperti. Questo rende il sistema completamente automatico e di facile translazionabilità tra i centri.

Il sistema creato analizza il segnale e emette una valutazione multi-obiettivo sul paziente. Le grandi dimensioni del dataset iniziale (621 EEG) su questa tipologia rara di pazienti diminuisce la probabilità di overfittare i modelli e la convalida tramite il recente metodo di nested cross-validation rende i modelli ottenuti robusti e assicurano un ottimo grado di generalizzazione. I modelli creati ottengono risultati validi, confrontati con la letteratura e la complessità del problema rende il sistema di classificazione utile per valutazioni cliniche. La complessità del problema si basa anche sul classificare l'intero EEG avendo un solo valore di riferimento per ogni segnale. Questo impedisce il riconoscimento de segnale epoca per epoca, ma una stima di tutta la registrazione.

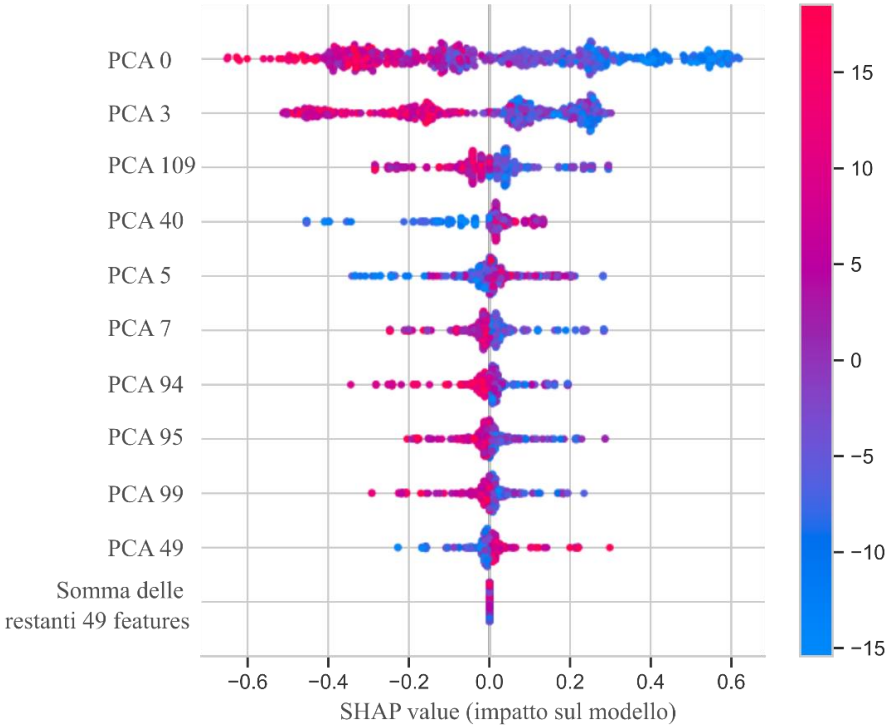
I risultati ottenuti per i descrittori di frequenza e di voltaggio sono eccellenti, indice della semplicità del problema in confronto agli altri descrittori. I modelli sono riusciti a generalizzare il problema e le feature estratte sono state sufficientemente rappresentative del problema. Analizzando il risultato ottenuto per la simmetria esso è ottimo poiché la sua accuratezza ricade all'interno della variabilità inter-operatore, questo rende il modello consultabile nella pratica clinica. Anche il risultato ottenuto sul rilevamento di grafoelementi è un ottimo pur non specificando i grafotipo presente. Per la ricerca di quest'ultimo si consiglia di utilizzare la strategia di template

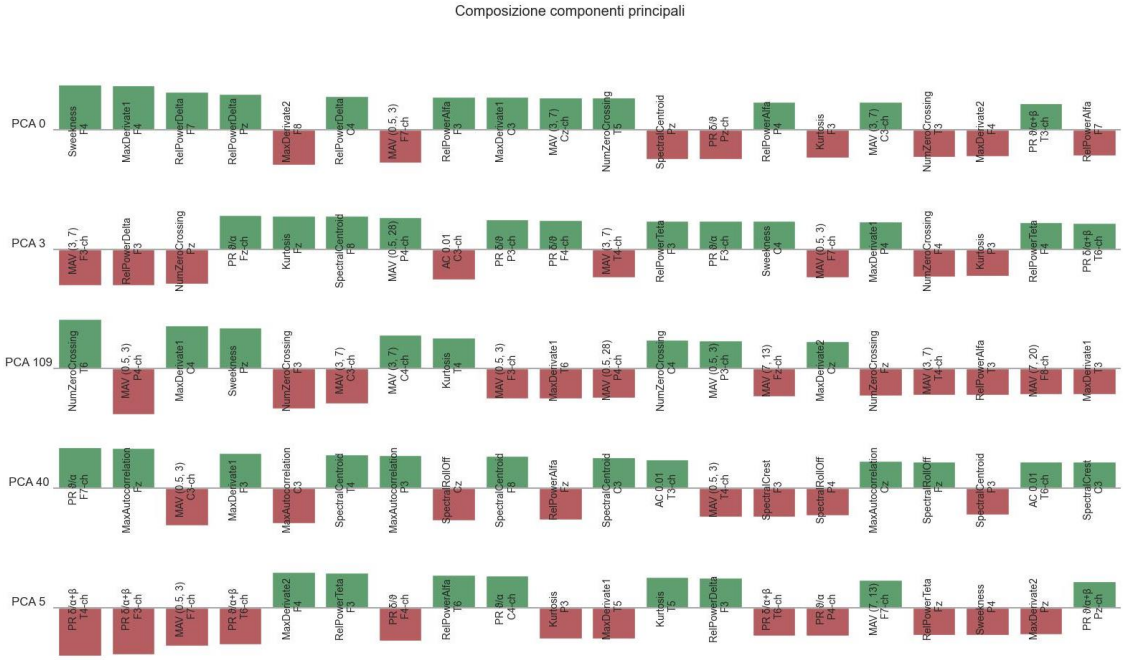
matching con il template opportuno. Per quanto riguarda le anomale lente delta, esse non si presentano in tutte le epoche e non si ripetono a frequenza costante. Per questo motivo estrarre un valore mediato tra tutte le epoche come rappresentativo del paziente potrebbe limitare il manifestarsi di una differenza tra segnali in cui le anomalie sono presenti da quelle in cui non lo sono. Tuttavia, i risultati per il descrittore sono buoni. In modo simile il gradiente antero-posterire si può manifestare solo per pochi minuti per fare in modo che il gradiente venga classificato come presente. Allo stesso modo delle anomalie lente delta la media rende più uniformi i valori di feature estratti. Per compensare queto problema è stata abbassata la soglia sulla probabilità a posteriori in modo da rendere più sensibile il modello.

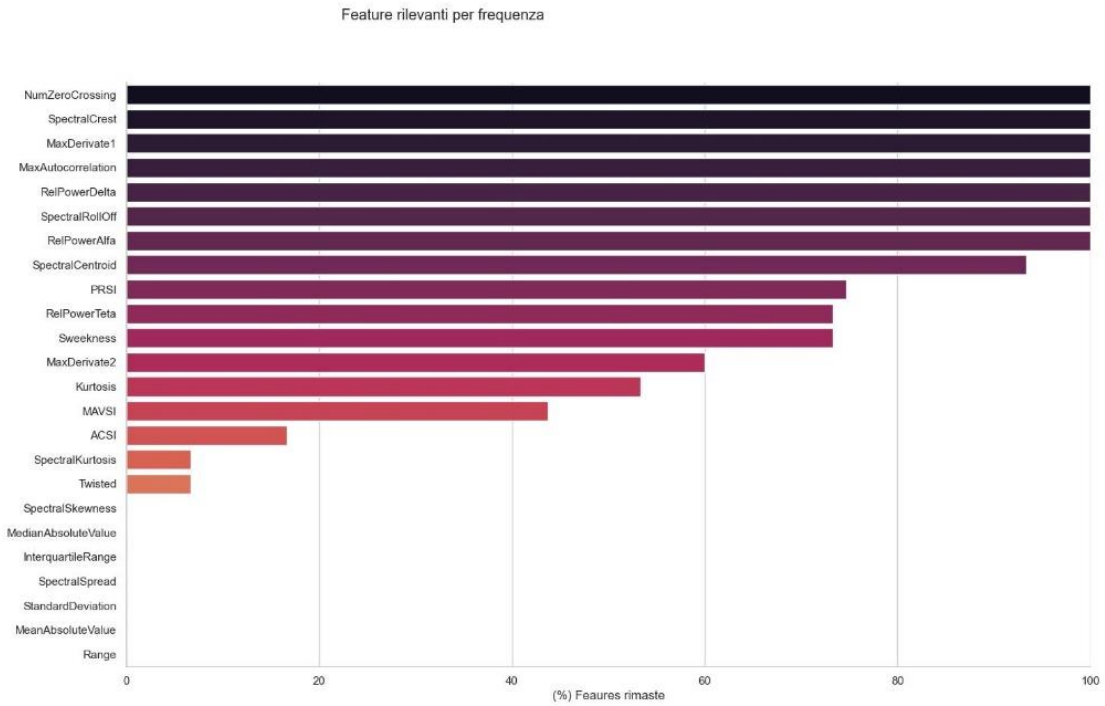
L'interfaccia grafica realizzata è di facile interpretazione e rende il sistema facilmente utilizzabile, esso prevede il caricamento del segnale da parte dell'operatore e in modo automatico mostra i risultati delle predizioni. La rappresentazione dei risultati prevede sia il valore della predizione come testo, sia la probabilità a posteriori dei modelli alla quale il clinico può fare riferimento espressa come indice su una barra colorata. Il valore del cursore sulla barra rappresenta la probabilità a posteriori, più il cursore è vicino a un estremo più il modello è sicuro della predizione, il centro colorato in rosso rappresenta l'indecisione del modello. L'interfaccia grafica prevede inoltre delle figure topografiche del segnale in analisi, esse rappresentano con una scala cromatica la media dell'intensità del segnale per ogni canale e la potenza media in ogni banda. I disegni aiutano il clinico nell'interpretazione e nella valutazione del segnale fornendo una prova della classificazione effettuata.

Appendice

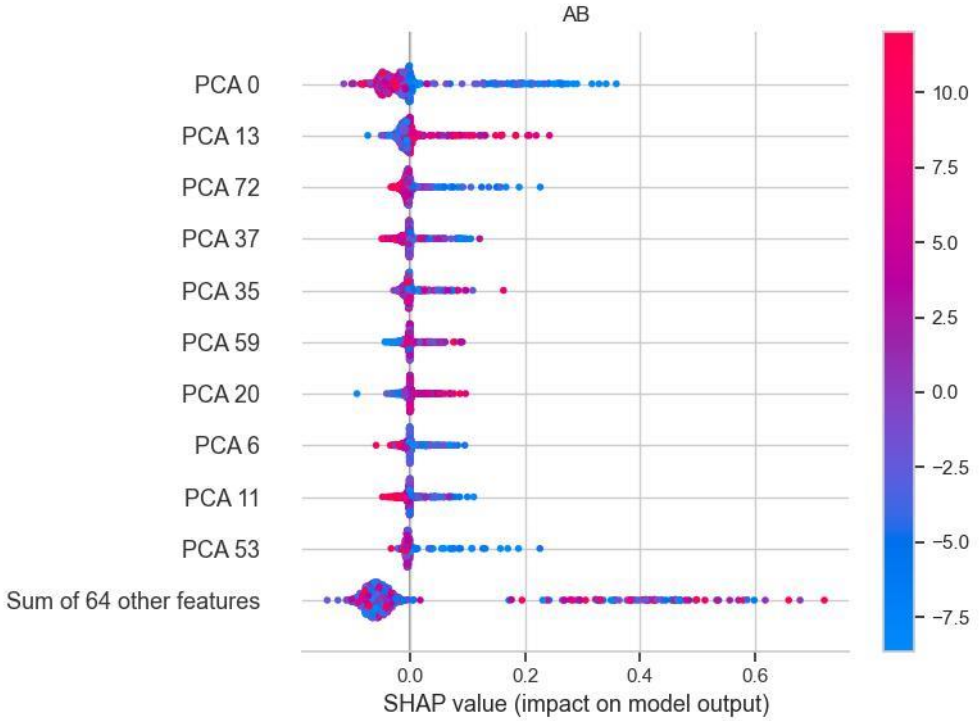
Frequenza

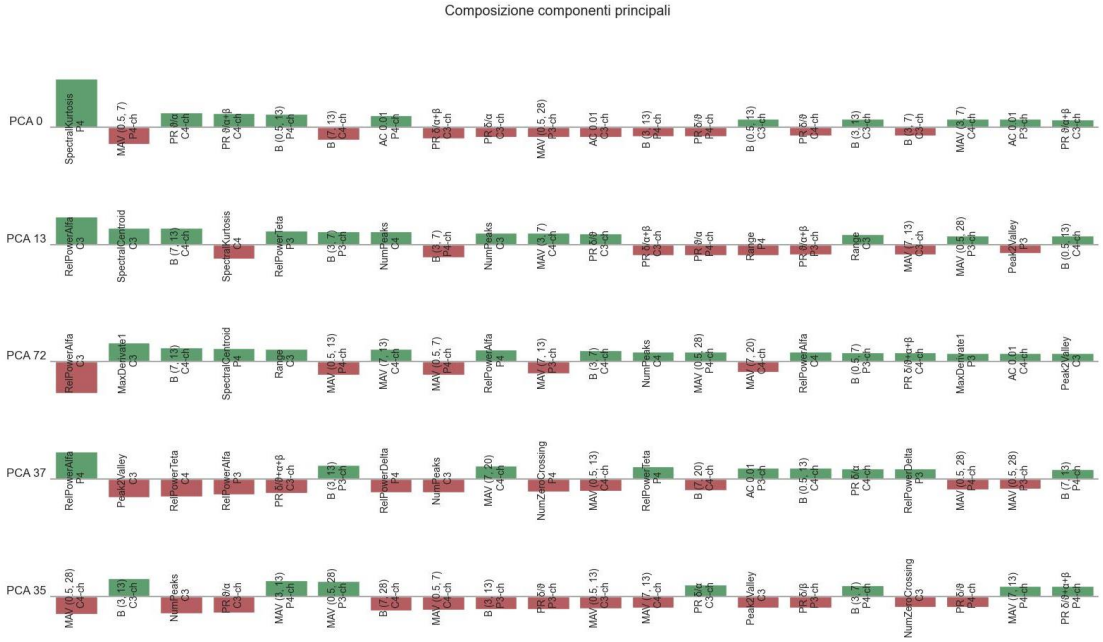


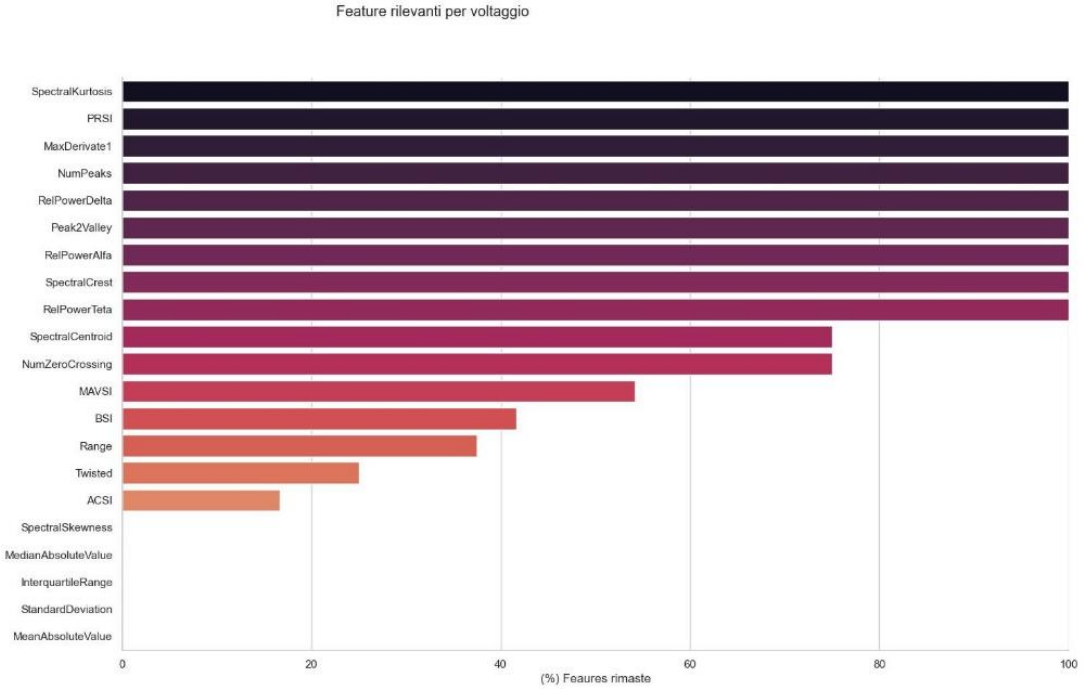



Feature rilevanti per frequenza
NumZeroCrossing
SpectralCrest
MaxDerivate1
MaxAutocorrelation
RelPowerDelta
SpectralRollOff
RelPowerAlfa
SpectralCentroid
PRSI
RelPowerTeta
Sweekness
MaxDerivate2
Kurtosis
MAVSI
ACSI
SpectralKurtosis
Twisted
SpectralSkewness
MedianAbsoluteValue
InterquartileRange
SpectralSpread
StandardDeviation
MeanAbsoluteValue
Range
0
20
40
60
80
100
(%) Feaures rimaste

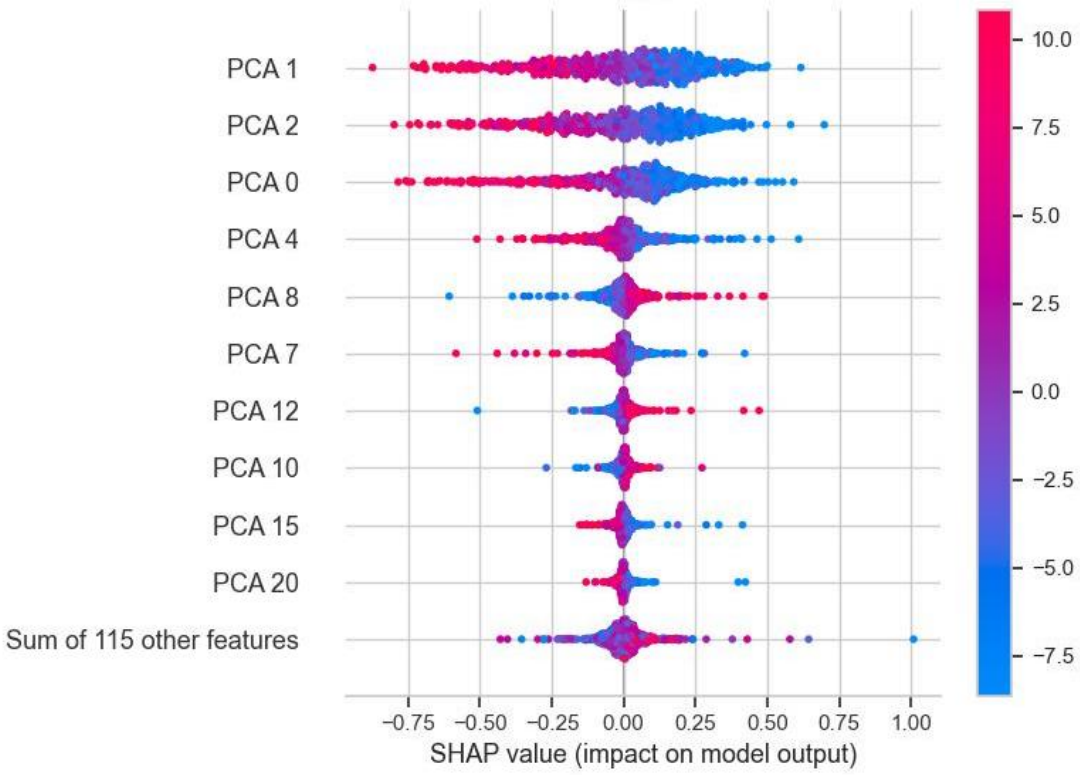
Voltaggio

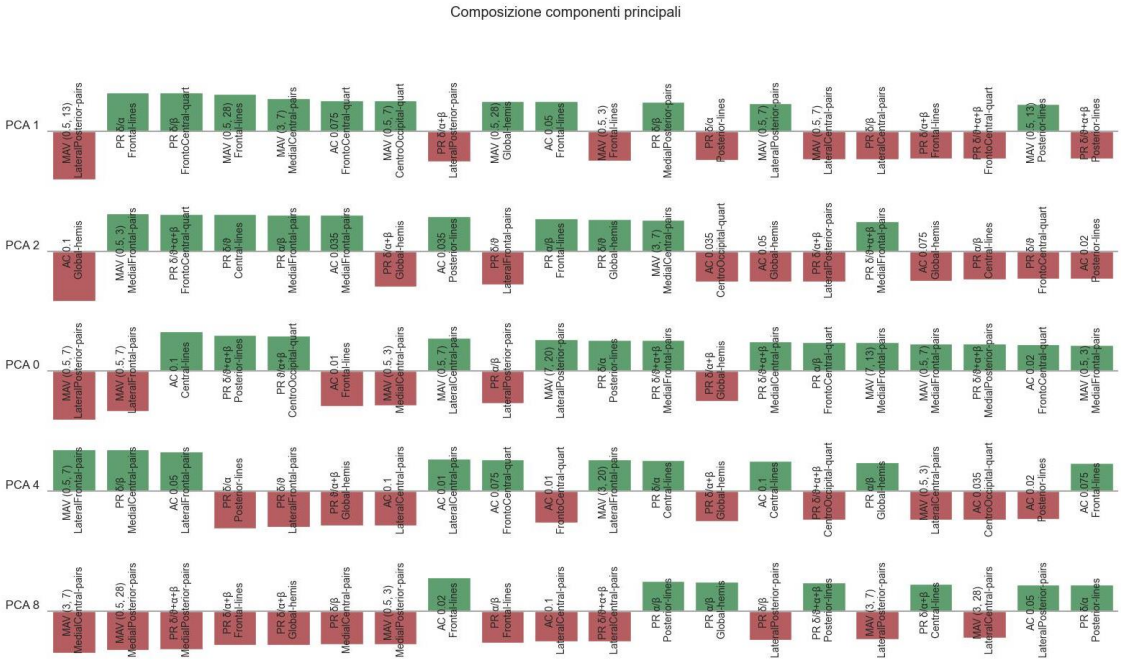

AB
PCA 0
PCA 13
PCA 72
PCA 37
PCA 35
PCA 59
PCA 20
PCA 6
PCA 11
PCA 53
Sum of 64 other features
0.0
0.2
0.4
0.6
SHAP value (impact on model output)
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5

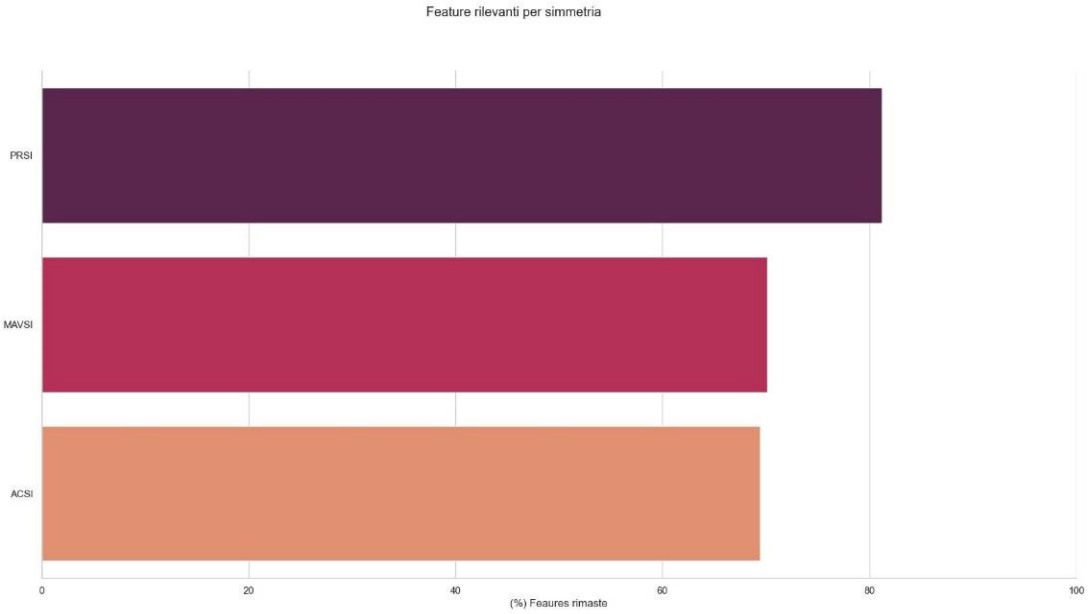

Composizione componenti principali
PCA 0
PCA 13
PCA 72
PCA 37
PCA 35


Feature rilevanti per voltaggio
SpectralKurtosis
PRSI
MaxDerivate1
NumPeaks
RelPowerDelta
Peak2Valley
RelPowerAlfa
SpectralCrest
RelPowerTeta
SpectralCentroid
NumZeroCrossing
MAVSI
BSI
Range
Twisted
ACSI
SpectralSkewness
MedianAbsoluteValue
InterquartileRange
StandardDeviation
MeanAbsoluteValue
0
20
40
60
80
100
(%) Feaures rimaste

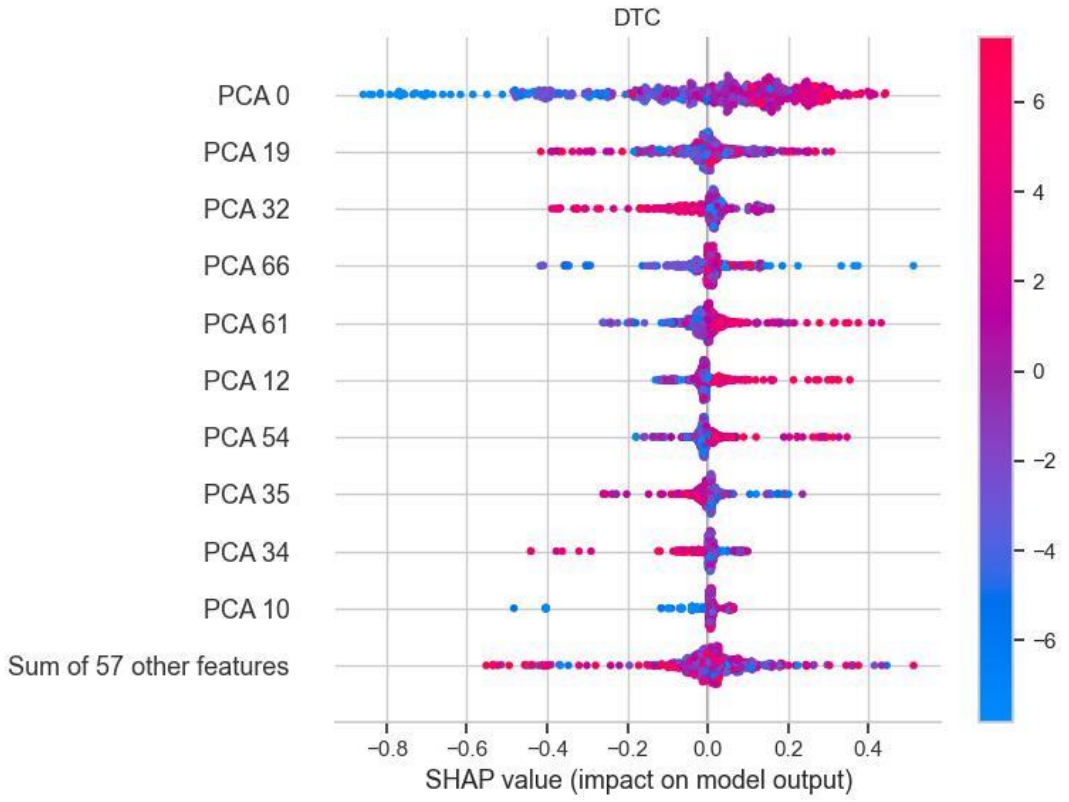
Simmetria

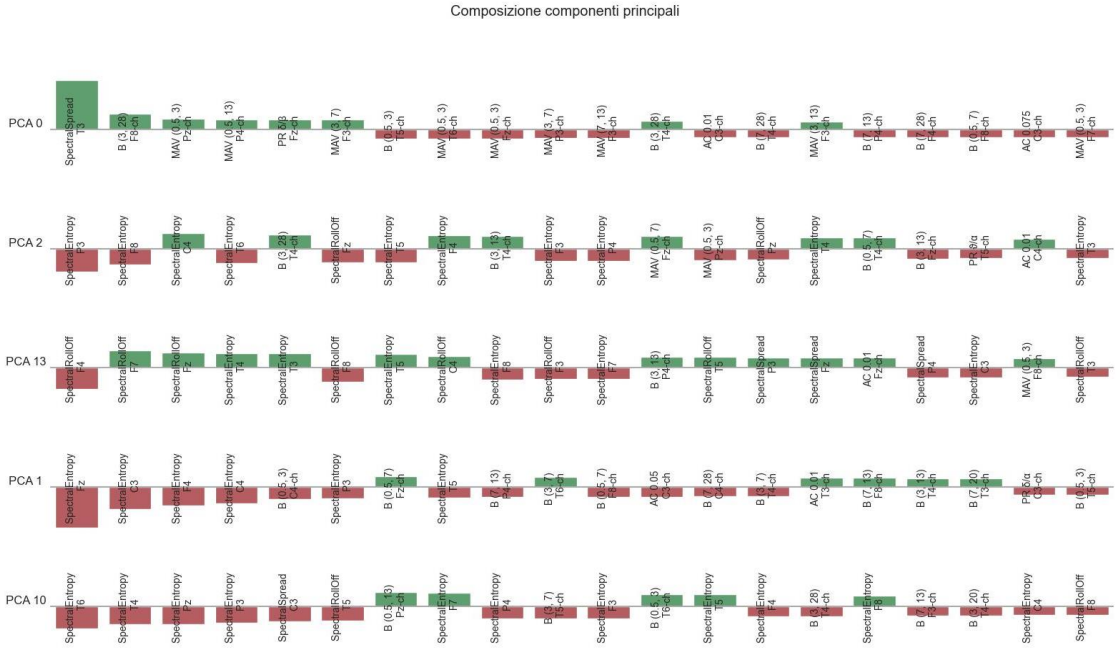

PCA 1
PCA 2
PCA 0
PCA 4
PCA 8
PCA 7
PCA 12
PCA 10
PCA 15
PCA 20
Sum of 115 other features
−0.75 −0.50 −0.25 0.00 0.25 0.50 0.75 1.00
SHAP value (impact on model output)
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5

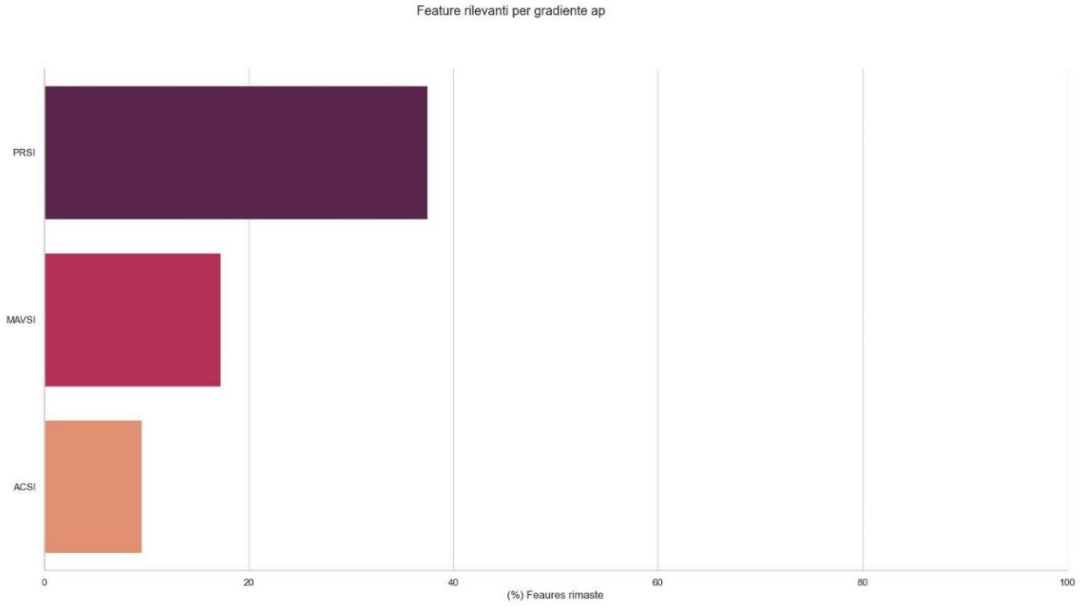

Composizione componenti principali
PCA 1
PCA 2
PCA 0
PCA 4
PCA 8



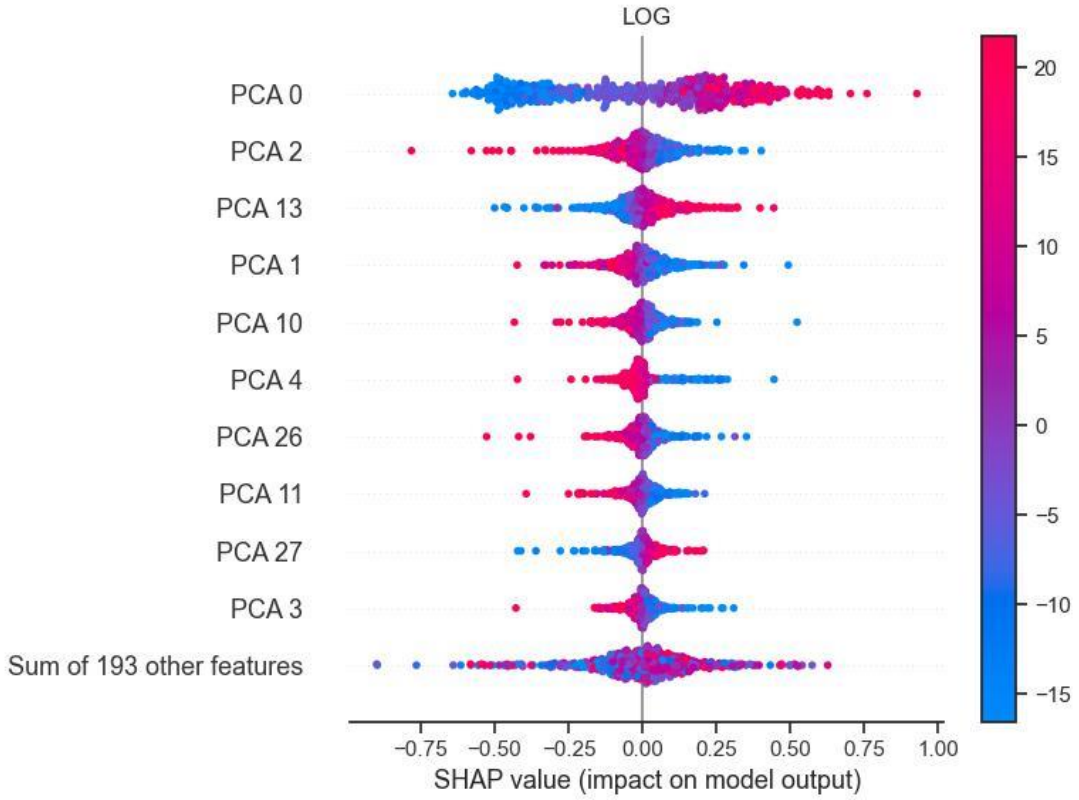
Gradiente antero posteriore

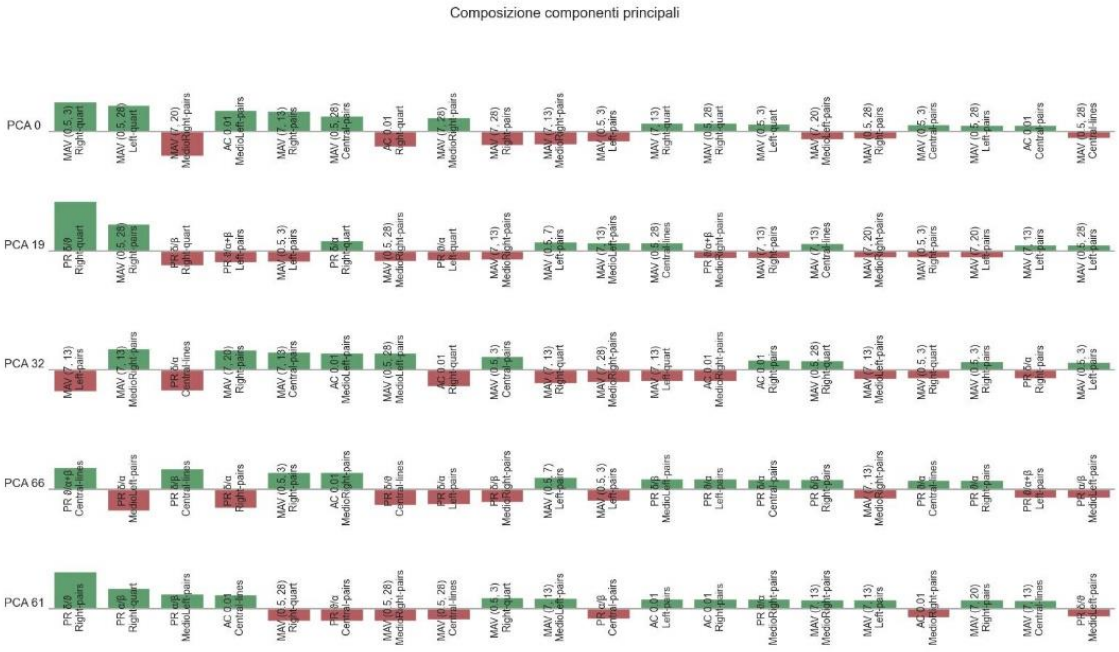


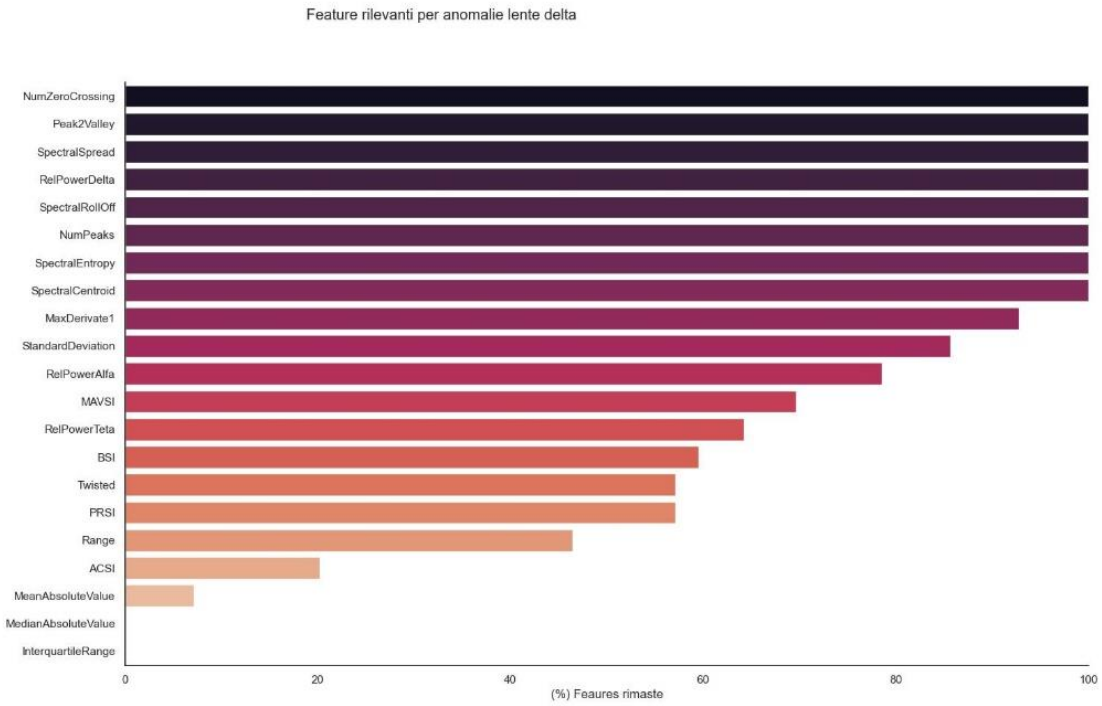

Composizione componenti principali
PCA 0
PCA 2
PCA 13
PCA 1
PCA 10


Feature rilevanti per gradiente ap
PRSI
MAVSI
ACSI
0
20
40
60
80
100
(%) Feaures rimaste

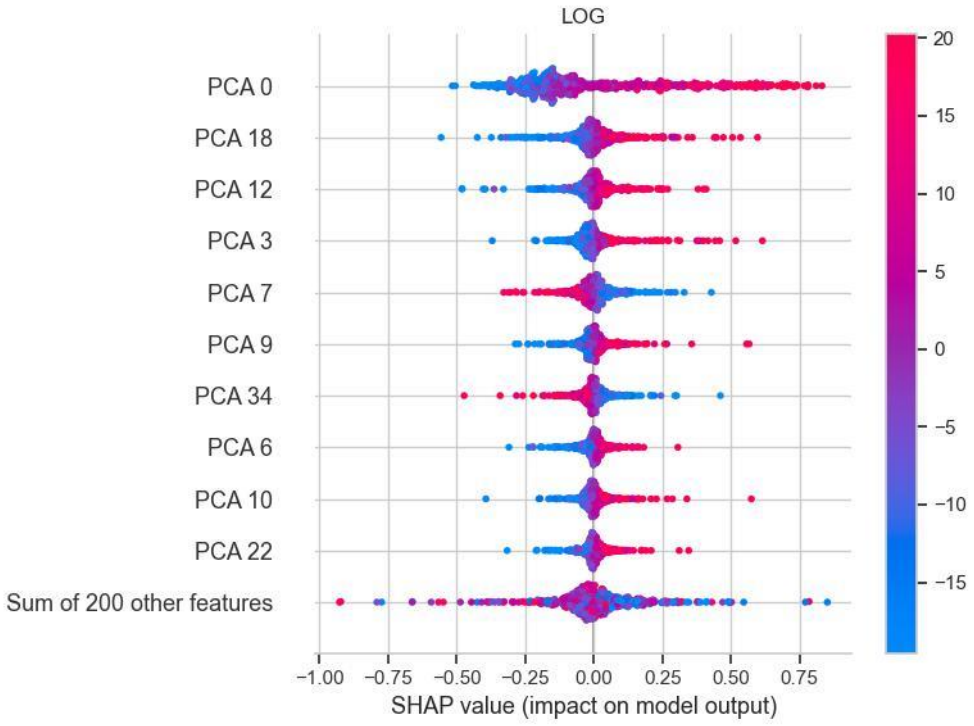
Anomali lente delta

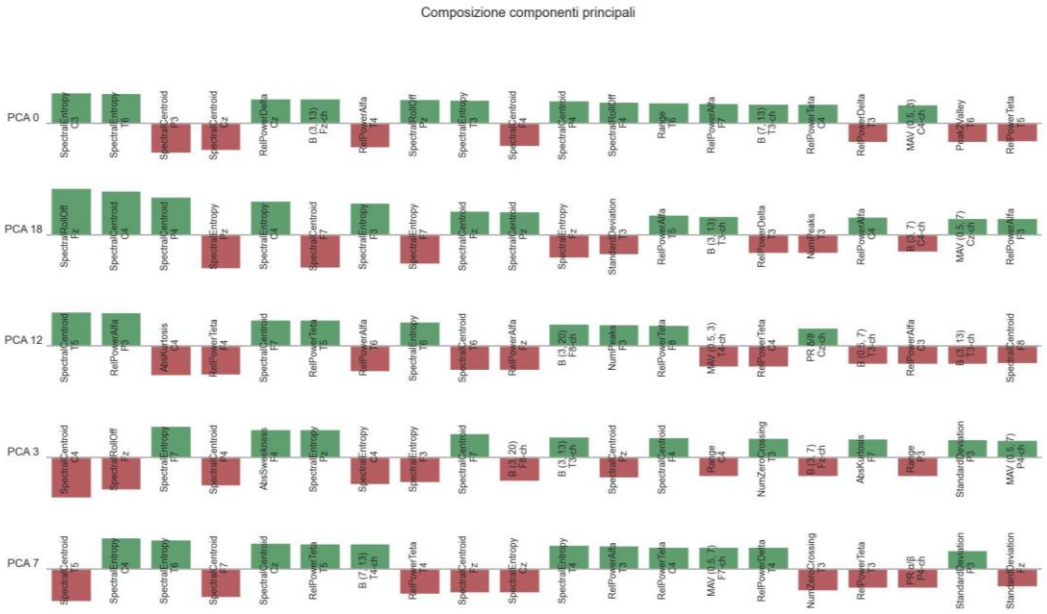


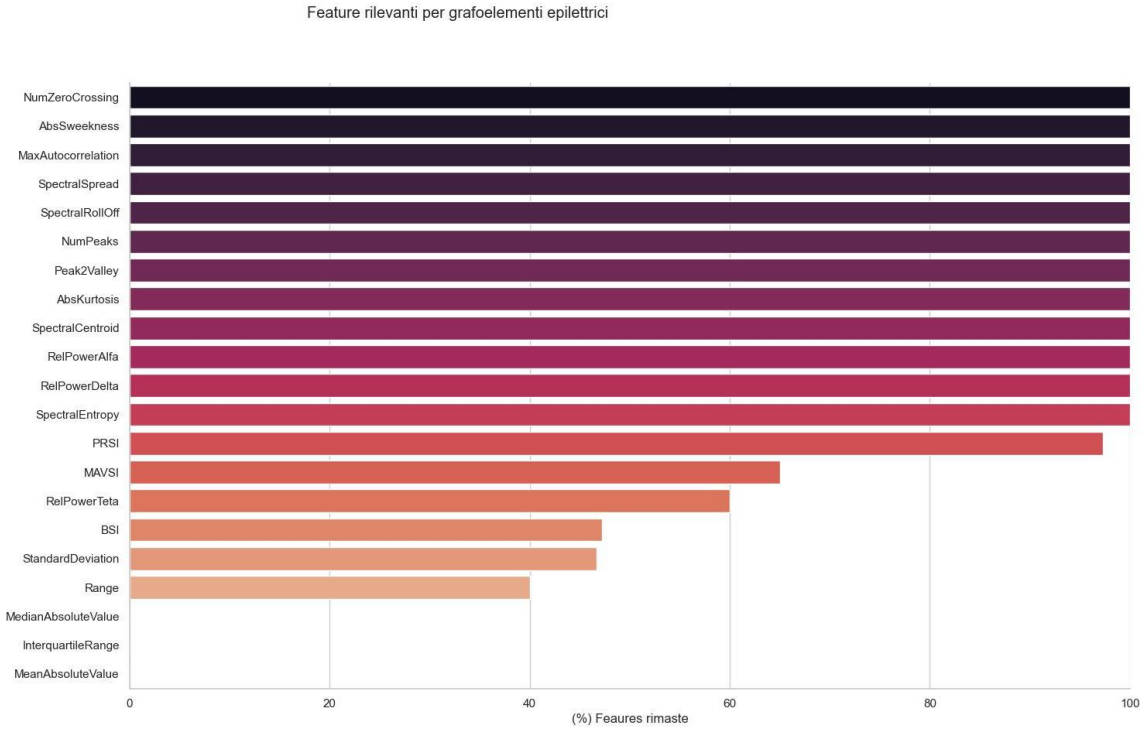




Grafoelementi epilettici




Composizione componenti principali
PCA 0
PCA 18
PCA 12
PCA 3
PCA 7


Feature rilevanti per grafoelementi epilettrici
NumZeroCrossing
AbsSweekness
MaxAutocorrelation
SpectralSpread
SpectralRollOff
NumPeaks
Peak2Valley
AbsKurtosis
SpectralCentroid
RelPowerAlfa
RelPowerDelta
SpectralEntropy
PRSI
MAVSI
RelPowerTeta
BSI
StandardDeviation
Range
MedianAbsoluteValue
InterquartileRange
MeanAbsoluteValue
0
20
40
60
80
100
(%) Feaures rimaste

Bibliografia

[1] C.-P. F, C.-G. M, D. P, R.-M. Jm, Z.-H. Jm, and undefined, '[Acquired brain injury: a proposal for its definition, diagnostic criteria and classification].', *Rev. Neurol.*, vol. 54, no. 6, pp. 357–366, Mar. 2012.

[2] J. T. Giacino, 'The vegetative and minimally conscious states: Consensus-based criteria for establishing diagnosis and prognosis', *NeuroRehabilitation*, vol. 19, no. 4, pp. 293–298, Jan. 2004, doi: 10.3233/NRE-2004-19405.

[3] M.-A. Bruno, A. Vanhaudenhuyse, A. Thibaut, G. Moonen, and S. Laureys, 'From unresponsive wakefulness to minimally conscious PLUS and functional locked-in syndromes: recent advances in our understanding of disorders of consciousness', *J. Neurol.*, vol. 258, no. 7, pp. 1373–1384, Jul. 2011, doi: 10.1007/s00415-011-6114-x.

[4] P. Guldenmund, J. Stender, L. Heine, and S. Laureys, 'Mindsight: Diagnostics in Disorders of Consciousness', *Crit. Care Res. Pract.*, vol. 2012, p. e624724, Nov. 2012, doi: 10.1155/2012/624724.

[5] 'Practice Guideline Update Recommendations Summary: Disorders of Consciousness: Report of the Guideline Development, Dissemination, and Implementation Subcommittee of the American Academy of Neurology; the American Congress of Rehabilitation Medicine; and the National Institute on Disability, Independent Living, and Rehabilitation Research - ScienceDirect'. https://www.sciencedirect.com/science/article/abs/pii/S0003999318304465 (accessed May 16, 2022).

[6] 'Assessment Scales for Disorders of Consciousness: Evidence-Based Recommendations for Clinical Practice and Research - ScienceDirect'. https://www.sciencedirect.com/science/article/abs/pii/S0003999310006039 (accessed May 16, 2022).

[7] 'American Clinical Neurophysiology Society's Standardized Critical Care EEG Terminology: 2021 Version - PMC'. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8135051/ (accessed May 16, 2022).

[8] 'European Academy of Neurology guideline on the diagnosis of coma and other disorders of consciousness - Kondziella - 2020 - European

Journal of Neurology - Wiley Online Library'. https://onlinelibrary.wiley.com/doi/full/10.1111/ene.14151 (accessed May 16, 2022).

[9] 'The misdiagnosis of prolonged disorders of consciousness by a clinical consensus compared with repeated coma-recovery scale-revised assessment | SpringerLink'. https://link.springer.com/article/10.1186/s12883-020-01924-9 (accessed May 16, 2022).

[10] 'Predicting outcome in patients with moderate to severe traumatic brain injury using electroencephalography | SpringerLink'. https://link.springer.com/article/10.1186/s13054-019-2656-6 (accessed May 16, 2022).

[11] M. Scarpino *et al.*, 'EEG and Coma Recovery Scale-Revised prediction of neurological outcome in Disorder of Consciousness patients', *Acta Neurol. Scand.*, vol. 142, no. 3, pp. 221–228, 2020, doi: 10.1111/ane.13247.

[12] M. Scarpino *et al.*, 'Prognostic value of post-acute EEG in severe disorders of consciousness, using American Clinical Neurophysiology Society terminology', *Neurophysiol. Clin.*, vol. 49, no. 4, pp. 317–327, Sep. 2019, doi: 10.1016/j.neucli.2019.07.001.

[13] B. Hermann *et al.*, 'Multimodal FDG-PET and EEG assessment improves diagnosis and prognostication of disorders of consciousness', *NeuroImage Clin.*, vol. 30, p. 102601, Jan. 2021, doi: 10.1016/j.nicl.2021.102601.

[14] N. Gaspard, L. J. Hirsch, S. M. LaRoche, C. D. Hahn, M. B. Westover, and the C. C. E. M. R. Consortium, 'Interrater agreement for Critical Care EEG Terminology', *Epilepsia*, vol. 55, no. 9, pp. 1366–1373, 2014, doi: 10.1111/epi.12653.

[15] 'Recent Advances in Quantitative EEG as an Aid to Diagnosis and as a Guide to Neurofeedback Training for Cortical Hypofunctions, Hyperfunctions, Disconnections, and Hyperconnections: Improving Efficacy in Complicated Neurological and Psychological Disorders | SpringerLink'. https://link.springer.com/article/10.1007/s10484-009-9107-0 (accessed May 16, 2022).

[16] U. R. Acharya, S. L. Oh, Y. Hagiwara, J. H. Tan, H. Adeli, and D. P. Subha, 'Automated EEG-based screening of depression using deep convolutional neural network', *Comput. Methods Programs Biomed.*, vol. 161, pp. 103–113, Jul. 2018, doi: 10.1016/j.cmpb.2018.04.012.

[17] J. Jing *et al.*, 'Development of Expert-Level Automated Detection of Epileptiform Discharges During Electroencephalogram Interpretation', *JAMA Neurol.*, vol. 77, no. 1, pp. 103–108, Jan. 2020, doi: 10.1001/jamaneurol.2019.3485.

[18] F. S. Bao, D. Y.-C. Lie, and Y. Zhang, 'A New Approach to Automated Epileptic Diagnosis Using EEG and Probabilistic Neural Network', in *2008 20th IEEE International Conference on Tools with Artificial Intelligence*, Nov. 2008, vol. 2, pp. 482–486. doi: 10.1109/ICTAI.2008.99.

[19] A. Harati, S. López, I. Obeid, J. Picone, M. P. Jacobson, and S. Tobochnik, 'The TUH EEG CORPUS: A big data resource for automated EEG interpretation', in *2014 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, Dec. 2014, pp. 1–5. doi: 10.1109/SPMB.2014.7002953.

[20] L. R. Rabiner, 'A tutorial on hidden Markov models and selected applications in speech recognition', *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989, doi: 10.1109/5.18626.

[21] R. Agarwal, J. Gotman, D. Flanagan, and B. Rosenblatt, 'Automatic EEG analysis during long-term monitoring in the ICU', *Electroencephalogr. Clin. Neurophysiol.*, vol. 107, no. 1, pp. 44–58, Jul. 1998, doi: 10.1016/S0013-4694(98)00009-1.

[22] V. Jahmunah *et al.*, 'Automated detection of schizophrenia using nonlinear signal processing methods', *Artif. Intell. Med.*, vol. 100, p. 101698, Sep. 2019, doi: 10.1016/j.artmed.2019.07.006.

[23] G. C. Cawley and N. L. C. Talbot, 'On Over-fitting in Model Selection and Subsequent Selection Bias in Performance Evaluation', p. 29.

[24] R. W. Homan, J. Herman, and P. Purdy, 'Cerebral location of international 10–20 system electrode placement', *Electroencephalogr. Clin. Neurophysiol.*, vol. 66, no. 4, pp. 376–382, Apr. 1987, doi: 10.1016/0013-4694(87)90206-9.

[25] C. R. Harris *et al.*, 'Array programming with NumPy', *Nature*, vol. 585, no. 7825, Art. no. 7825, Sep. 2020, doi: 10.1038/s41586-020-2649-2.

[26] P. Virtanen *et al.*, 'SciPy 1.0: fundamental algorithms for scientific computing in Python', *Nat. Methods*, vol. 17, no. 3, Art. no. 3, Mar. 2020, doi: 10.1038/s41592-019-0686-2.

[27] A. Gramfort *et al.*, 'MNE software for processing MEG and EEG data', *NeuroImage*, vol. 86, pp. 446–460, Feb. 2014, doi: 10.1016/j.neuroimage.2013.10.027.

[28] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, 'Optuna: A Next-generation Hyperparameter Optimization Framework', in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, New York, NY, USA, Jul. 2019, pp. 2623–2631. doi: 10.1145/3292500.3330701.

[29] F. Pedregosa *et al.*, 'Scikit-learn: Machine Learning in Python', *Mach. Learn. PYTHON*, p. 6.

[30] G. V. den Broeck, A. Lykov, M. Schleich, and D. Suciu, 'On the Tractability of SHAP Explanations', *J. Artif. Intell. Res.*, vol. 74, pp. 851–886, Jun. 2022, doi: 10.1613/jair.1.13283.

[31] P. Welch, 'The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms', *IEEE Trans. Audio Electroacoustics*, vol. 15, no. 2, pp. 70–73, Jun. 1967, doi: 10.1109/TAU.1967.1161901.

[32] D. Slepian, 'Prolate spheroidal wave functions, fourier analysis, and uncertainty — V: the discrete case', *Bell Syst. Tech. J.*, vol. 57, no. 5, pp. 1371–1430, May 1978, doi: 10.1002/j.1538-7305.1978.tb02104.x.

[33] 'Spectral Analysis for Physical Applications - Donald B. Percival, Andrew T. Walden, Percival Donald B., Walden Andrew T. - Google Libri'. https://books.google.it/books?hl=it&lr=&id=FubniGJ0ECQC&oi=fnd&pg=PR15&dq=Donald+B.+Percival+and+Andrew+T.+Walden.+Spectral+Analysis+for+Physical+Applications:+Multitaper+and+Conventional+Univariate+Techniques.+Cambridge+University+Press,+Cambridge&ots=mwEaKgvGCN&sig=Z6ew1XucMcUYLpV8HqapaAh1pKo#v=on

epage&q=Donald%20B.%20Percival%20and%20Andrew%20T.%20Walden.%20Spectral%20Analysis%20for%20Physical%20Applications%3A%20Multitaper%20and%20Conventional%20Univariate%20Techniques.%20Cambridge%20University%20Press%2C%20Cambridge&f=false (accessed May 16, 2022).

[34] 'Reproducibility and clinical relevance of quantitative EEG parameters in cerebral ischemia: A basic approach - ScienceDirect'. https://www.sciencedirect.com/science/article/abs/pii/S1388245709002375 (accessed May 16, 2022).

[35] 'Bi-Frequency Symmetry Difference EIT-Feasibility and Limitations of Application to Stroke Diagnosis - PubMed'. https://pubmed.ncbi.nlm.nih.gov/31869810/ (accessed Jun. 07, 2022).

[36] W. H. Kruskal and W. A. Wallis, 'Use of Ranks in One-Criterion Variance Analysis', *J. Am. Stat. Assoc.*, vol. 47, no. 260, pp. 583–621, Dec. 1952, doi: 10.1080/01621459.1952.10483441.

[37] I. Rodríguez-Fdez, A. Canosa, M. Mucientes, and A. Bugarín, 'STAC: A web platform for the comparison of algorithms using statistical tests', in *2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Aug. 2015, pp. 1–8. doi: 10.1109/FUZZ-IEEE.2015.7337889.

[38] S. Parvandeh, H.-W. Yeh, M. P. Paulus, and B. A. McKinney, 'Consensus features nested cross-validation', *Bioinformatics*, vol. 36, no. 10, pp. 3093–3098, May 2020, doi: 10.1093/bioinformatics/btaa046.

[39] R. Duda, P. Hart, and D. G.Stork, 'Pattern Classification', in *Wiley Interscience*, vol. xx, 2001.

[40] *The Elements of Statistical Learning*. Accessed: May 20, 2022. [Online]. Available: https://link.springer.com/book/10.1007/978-0-387-21606-5

[41] O. Ledoit and M. Wolf, 'Honey, I Shrunk the Sample Covariance Matrix', *J. Portf. Manag.*, vol. 30, no. 4, pp. 110–119, Jul. 2004, doi: 10.3905/jpm.2004.110.

[42] L. Breiman, 'Random Forests', *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.

[43] L. Breiman, 'Rejoinder: Arcing Classifiers', *Ann. Stat.*, vol. 26, no. 3, pp. 841–849, 1998.

[44] ‘A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting - ScienceDirect’. https://www.sciencedirect.com/science/article/pii/S002200009791504X (accessed May 18, 2022).