

POLITECNICO DI MILANO
Facoltà di Ingegneria dell'Informazione
Corso di Laurea in Ingegneria Informatica
Laurea Magistrale



RICONOSCIMENTO DELL'ATTENZIONE IN UNA BCI P300

AI & R Lab
Laboratorio di intelligenza artificiale e robotica

Relatore: Prof. Matteo Matteucci
Correlatore: Ing. Fabio Veronese

Tesi di Laurea di:
Leonardo Volpe
matr. 724874

Anno Accademico 2010-2011

Indice

Sommario	1
1 Introduzione	3
1.1 La ricerca nel campo delle BCI	3
1.2 Scopo della tesi	5
1.3 Struttura della tesi	6
2 BCI basate su P300	7
2.1 Cenni di neurofisiologia	7
2.1.1 Encefalo	7
2.1.2 Neuroni	9
2.1.3 Sinapsi	10
2.2 Monitorare l'attività cerebrale	11
2.2.1 L'elettroencefalografia	11
2.2.2 Analisi di un tracciato EEG	13
2.2.3 Potenziali evocati evento correlati e P300	15
2.3 Brain-Computer Interface	16
2.3.1 Classificazione delle BCI	17
2.3.2 Le componenti di una BCI	18
2.4 Controllo asincrono delle BCI	21
2.5 Stato dell'arte	22
2.5.1 BCI P300	22
2.5.2 BCI Self-paced	22
2.5.3 BCI P300 Self-paced	23
3 Materiali e metodi	25
3.1 Acquisizione	25
3.1.1 Hardware e software	25
3.1.2 Speller P300	27
3.1.3 Set up	28
3.1.4 Protocollo di acquisizione	29
3.1.5 Soggetti testati	31
3.1.6 Dati acquisiti	31
3.2 Train e test dello Speller	32
3.2.1 Dataset	32
3.2.2 Algoritmo Genetico	32

3.2.3	Classificatore Logistico	34
3.2.4	Test	34
3.3	Realizzazione del filtro asincrono	36
3.3.1	Dataset	36
3.3.2	Estrazione delle feature	38
3.3.3	Selezione delle feature	41
3.3.4	Classificazione	43
3.3.5	Algoritmo di selezione automatica	46
3.3.6	Valutazione del classificatore	47
4	Risultati sperimentali	51
4.1	Combinazioni di test	52
4.2	Confronto classificatori	54
4.3	Confronto metodi feature selection	56
4.4	Confronto soggetti	58
4.5	Selezione automatica di feature e classificatore	59
4.6	Confronto con altri sistemi presenti in letteratura	59
5	Conclusioni e sviluppi futuri	63
	Ringraziamenti	65
	Bibliografia	67

Sommario

Una interfaccia cervello-computer (generalmente indicata col termine BCI dall'inglese *Brain-Computer Interface*) consente di monitorare l'attività cerebrale di un utente, nel nostro caso tramite EEG, e utilizzare alcuni pattern per attuarne la volontà.

Tra questi pattern ci sono i cosiddetti potenziali evento-correlati. Un potenziale evento-correlato è un segnale cerebrale che si forma in risposta a uno stimolo; in particolare il segnale P300 è un potenziale positivo che si manifesta circa 300 ms dopo lo stimolo.

Un importante aspetto che caratterizza i vari protocolli BCI è la possibilità di effettuare il controllo asincrono (*self-paced*); una BCI con controllo asincrono è in grado di dedurre dal segnale EEG se l'utente intende utilizzare l'interfaccia oppure non sta prestando attenzione alla stessa.

Gli studi sul controllo asincrono di BCI basate su potenziali evento-correlati (ERP) sono ancora scarsi.

Lo scopo della presente tesi è quello di investigare la possibilità di un controllo asincrono di BCI basate su ERP e in particolare sulla possibilità di rendere asincrona la BCI sviluppata al Politecnico di Milano: lo Speller P300 (un software che permette la scrittura di un testo, lettera per lettera).

Per applicare un controllo asincrono a un sistema basato su ERP si è sfruttata la presenza o l'assenza della risposta allo stimolo proposto come discriminante per capire se ci si trova in uno stato in cui l'utente intende utilizzare il dispositivo oppure se l'utente non sta prestando attenzione allo stesso. In particolare nello Speller P300 ci si è ricondotti alla ricerca di un metodo per rintracciare la presenza o l'assenza del segnale P300 a partire da alcuni score assegnati dalla BCI ad ogni lettera.

Si è creato un filtro che, se applicato allo Speller P300, permette di realizzare il controllo asincrono.

I buoni risultati, ottenuti su esperimenti pilota, mostrano che questo approccio non solo è percorribile, ma anche efficace come meccanismo di correzione degli errori.

1 Introduzione

Gli uomini hanno da sempre utilizzato strumenti e macchine per essere aiutati nello svolgimento del proprio lavoro. Fino a pochi decenni fa l'unico modo che aveva un operatore umano per utilizzarli era l'uso di leve, bottoni, pedali. Con i progressi dell'elettronica e con l'avvento dei personal computer, le interfacce sono diventate più usabili e le vecchie leve e manopole sono state sostituite da mouse, tastiere, touch screen e riconoscitori vocali. Tuttavia, per quanto possano essere avanzate, tutte le interfacce in uso nei moderni dispositivi richiedono l'uso da parte dell'operatore di un qualche muscolo per poterli azionare.

Recentemente sono stati effettuati studi che possono rendere possibile quello che fino a ora è stato solo un sogno dell'uomo, ossia la possibilità di comandare un dispositivo direttamente con le onde cerebrali. Questo tipo di interfacce, oltre a rappresentare una possibile futura alternativa a tutti gli strumenti utilizzati oggi, assumono grande importanza per tutte quelle persone per le quali l'uso del sistema muscolare rappresenta un problema come, ad esempio, coloro che sono affetti da patologie quali SLA, Sclerosi Multipla ed estese lesioni cerebrali.

A rendere possibile questo tipo di comunicazione sono nuovi tipi di interfaccia, per l'appunto interfacce cervello-computer, generalmente indicate col termine BCI dall'inglese Brain-Computer Interface.

1.1 La ricerca nel campo delle BCI

Una BCI è un dispositivo in grado di monitorare l'attività cerebrale di una persona e utilizzare alcuni segnali, come ad esempio l'EEG, per interpretarne e attuarne la volontà. In maniera schematica possiamo dire che un'interfaccia cervello-computer è composta dai seguenti moduli:

- un modulo di acquisizione che permette di registrare il segnale generato dal cervello dell'utente;

- un modulo di analisi del segnale, che, processando e classificando i dati ricevuti dal modulo precedente, è in grado di estrarre informazione utile dal segnale grezzo registrato dal cervello dell'utente;
- infine, un modulo che, sfruttando l'informazione utile estratta durante l'analisi del segnale, produce un output che è reinviato all'utente come feedback.

La ricerca nel campo delle BCI affonda le sue radici nelle indagini e negli studi riguardanti l'anatomia e la fisiologia del cervello. Una delle ipotesi fondamentali su cui si basano molte ricerche nel campo delle neuro-scienze è l'assunto che esista una relazione tra stati mentali e attività neurale [17]. In altre parole, sarebbe possibile riconoscere un'esperienza mentale analizzando l'attività elettrica generata dai neuroni all'interno del cervello. In verità, questa ipotesi è tuttora molto dibattuta e la connessione tra stati mentali e attività neurale, ammesso che esista, è ben lungi dall'essere nota. Di fronte a questo problema, i ricercatori nel campo delle interfacce cervello-computer hanno seguito un approccio più pragmatico. Consapevoli della difficoltà di dare una spiegazione teorica completa e coerente all'attività neurale, molti hanno preferito limitarsi a studiare in quale modo utilizzare e sfruttare i segnali prodotti dal cervello piuttosto che darne un'interpretazione [52]. Pertanto l'obiettivo principale della ricerca sulle interfacce cervello-computer è quello di captare ed elaborare segnali cerebrali [3].

Per quanto vi siano vari metodi per captare questi segnali, quello che fino a oggi è utilizzato in maniera prevalente è l'elettroencefalografia, ovvero la registrazione dei potenziali encefalici attraverso elettrodi applicati sullo scalpo. Questo metodo ha il merito di essere non invasivo, tecnicamente poco impegnativo e disponibile a un costo relativamente basso.

Sono stati sviluppati diversi prototipi di BCI basati sui segnali EEG, ma, in particolare, questo lavoro di tesi si basa sui precedenti studi in questo campo del Politecnico di Milano [9].

Tali studi hanno portato alla realizzazione di uno Speller P300, ossia una BCI che consente di scrivere sfruttando il segnale P300 prodotto dal cervello sulla base di stimoli visivi.

Un importante aspetto delle BCI basate su EEG è il concetto di controllo asincrono; generalmente le BCI che ne fanno uso sono dette *asynchronous BCI* o *self-paced BCI*. In questo tipo di approccio, la BCI deve essere in grado di dedurre dal segnale EEG se l'utente intende utilizzare l'interfaccia (questo stato è detto *intentional control state*, IC) oppure non farlo (*noncontrol state*, NC) [43]. Tuttavia la maggior parte delle BCI basate su EEG lavorano in modalità sincrona, ossia assumono che l'utente sia sempre in stato di controllo. Una BCI sincrona è costantemente impegnata nella traduzione simultanea di segnali EEG in comandi di controllo: se l'utente non intende controllare l'interfaccia si verificano quindi false interpretazioni del segnale cerebrale e di conseguenza false azioni di controllo. Una BCI asincrona, per essere davvero tale, deve quindi essere in grado di far passare liberamente l'utente da uno stato di controllo a uno di non controllo senza l'aiuto di un input esterno.

Gli ultimi anni hanno visto un incremento dell'interesse di ricerca nel campo del controllo asincrono. Ma quasi tutti si sono concentrati sulla rilevazione di un tipo di segnale, il segnale detto *motor imagery* (per maggiori informazioni si rimanda a [40]), mentre ci sono stati solo pochi studi sul controllo asincrono utilizzando un potenziale evento correlato, nel nostro caso il segnale P300.

Chiariamo cosa si intende per potenziale evento correlato. Un potenziale evento-correlato, con acronimo ERP dall'inglese *event-related potential*, è un segnale cerebrale misurabile, che si forma in risposta a uno stimolo interno oppure esterno e quindi non è sempre presente in una registrazione EEG. In particolare il segnale P300 è un potenziale positivo che si manifesta circa 300 ms dopo uno stimolo come reazione ad esso.

1.2 Scopo della tesi

Lo scopo della presente tesi è quello di investigare sulla possibilità di un controllo asincrono di BCI basate su potenziali evento correlati e in particolare sulla possibilità di rendere asincrono lo Speller P300 sviluppato al Politecnico di Milano.

Si è cercato di capire come applicare un controllo asincrono ad un sistema basato su ERP. Generalmente il controllo di BCI asincrone è determinato dagli stati (*state-driven*), ossia l'utente ha la possibilità di passare da uno stato IC ad uno NC e viceversa; questo è permesso quando il fenomeno neurologico alla base della BCI è un valore continuo con una soglia [43]. Nelle BCI ERP questo valore non esiste poiché quello che viene riconosciuto è il segnale elicitato dall'evento raro. Pertanto si rende necessario un approccio diverso al problema che non può essere *state-driven* ma deve essere *event-driven* ossia determinato dagli eventi. Per questo motivo si è deciso di sfruttare la presenza o l'assenza della risposta allo stimolo proposto come discriminante per capire se ci si trova in uno stato in cui l'utente intende utilizzare il dispositivo oppure non sta prestando attenzione allo stesso.

Tornando al caso di studio dello Speller P300, l'utente è stimolato una serie di volte con una matrice di lettere e la BCI assegna un valore numerico, influenzato dalla forma dell'onda dopo lo stimolo, ad ogni lettera per ogni serie; somma i vari punteggi e la lettera col punteggio più alto è data in output al sistema. Nel momento in cui l'utente non sta prestando la sua attenzione, lo Speller continua lo stesso a stimolare e dare una lettera in uscita. Pertanto si è studiato un metodo per rintracciare la presenza o l'assenza del segnale P300 a partire dai valori assegnati dalla BCI dopo lo stimolo.

A tal fine sono stati raccolti in laboratorio diversi dati di soggetti attenti agli stimoli proposti e di soggetti non attenti agli stessi e si sono elaborati questi dati con tecniche di *pattern recognition* al fine di creare un filtro che possa essere usato per rendere asincrono lo Speller P300.

1.3 Struttura della tesi

La presente tesi è costituita da cinque capitoli strutturati in maniera tale da guidare il lettore nella comprensione delle scelte effettuate e delle metodologie utilizzate.

Il presente capitolo offre una sintetica descrizione del settore di ricerca delle interfacce cervello-computer e spiega quale sia il ruolo, l'obiettivo e la motivazione di questa tesi.

Il secondo capitolo spiega meglio l'ambito di applicazione della tesi. Parte da un insieme di nozioni elementari di biologia. Studiando un argomento che affonda le sue radici tanto nel campo dell'informatica quanto nel campo della biologia, sarebbe impossibile procedere nella ricerca senza cognizioni basilari di neurofisiologia. Si passa poi alle moderne tecniche di registrazione dell'attività neurale, con particolare attenzione all'encefalografia, e si descrivono le caratteristiche delle interfacce cervello-computer. La fine del capitolo è dedicata allo stato dell'arte delle BCI, approfondendo maggiormente gli studi riguardanti le BCI asincrone.

Il terzo capitolo descrive il sistema sviluppato nel corso della tesi e gli esperimenti condotti per la validazione dello stesso. Sono presentati con precisione le configurazioni sperimentali adottate, i protocolli di laboratorio utilizzati durante gli esperimenti, l'ambiente software all'interno del quale sono state condotte le analisi dei dati raccolti durante gli esperimenti e i metodi utilizzati per la realizzazione del software.

Nel quarto capitolo vengono presentati i risultati ottenuti nel corso degli esperimenti.

Il quinto ed ultimo capitolo è dedicato alla discussione dei risultati e sono indicate possibili proposte per uno sviluppo futuro del progetto.

2 BCI basate su P300

Come affermato, il fine di una interfaccia cervello-computer è quello di costituire una via di comunicazione che dipenda esclusivamente dai segnali prodotti dall'attività neurale del cervello. E' opportuno dunque introdurre alcune nozioni elementari riguardo la natura e la struttura del cervello, delineando quegli aspetti che saranno di maggior interesse per le interfacce cervello-computer.

Nel seguito del capitolo si presenteranno varie tecniche di monitoraggio dell'attività cerebrale, con particolare riferimento all'elettroencefalografia.

La parte finale del capitolo è, invece, dedicata all'introduzione delle BCI ed allo stato dell'arte delle stesse.

2.1 Cenni di neurofisiologia

2.1.1 Encefalo

L'encefalo è il centro del sistema nervoso centrale umano, contenuto all'interno della scatola cranica. Esso è responsabile di tutte le attività del corpo umano, dalla regolazione del battito cardiaco alla gestione della memoria, dalla generazione di pensieri alla formazione di sogni e speranze [20]. La struttura del cervello ha una complessità seconda a nessuna degli altri organi del corpo umano e, tuttora, buona parte delle sue reali funzioni sono ancora oggetto di studio [20]. L'encefalo propriamente detto è costituito da (vedi Figura 2.1) [27, 42]:

Proencefalo: il cervello costituisce la parte anteriore e superiore dell'encefalo; con la sua massa esso occupa quasi interamente la scatola cranica;

Tronco encefalico (mesencefalo): il tronco encefalico è situato nella parte inferiore dell'encefalo, tra il proencefalo e il cervelletto; riceve input dai nervi cranici e trasferisce impulsi sensori e motori tra il cervello e la spina dorsale; contiene inoltre le strutture responsabili del controllo del battito cardiaco, della respirazione e della pressione sanguigna;

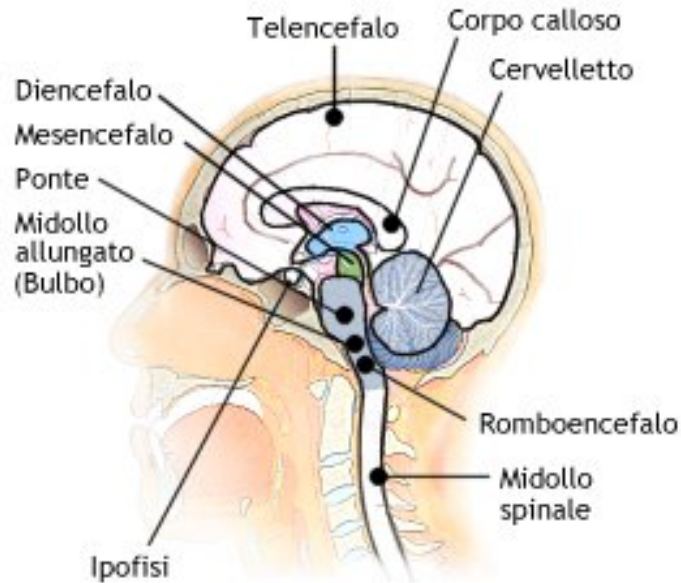


Figura 2.1: Struttura dell'encefalo

Cervelletto: il cervelletto è situato nella parte posteriore e inferiore dell'encefalo; ha un ruolo fondamentale nell'apprendimento e nella memorizzazione dei movimenti, nonché nella pianificazione e nel coordinamento.

Analizzando più a fondo il prosencefalo, è possibile suddividerlo in [27, 50]:

Diencefalo: collocato tra la parte superiore dell'encefalo e il mesencefalo, il diencefalo è costituito da strutture simmetriche specializzate, quali il terzo ventricolo, il talamo, il subtalamo, l'epitalamo e l'ipotalamo.

Telencefalo: al di sopra del diencefalo trova posto il telencefalo che costituisce la parte esterna dell'encefalo; il telencefalo si compone di due emisferi cerebrali, bilaterali e simmetrici, separati tra loro dalla scissura interemisferica longitudinale, e di alcune formazioni interemisferiche.

I due emisferi cerebrali che formano il telencefalo constano di una superficie esterna di materia grigia, la corteccia cerebrale e di un nucleo interno di materia bianca; inoltre, in profondità, all'interno dei due emisferi sono presenti anche altre strutture di materia grigia, chiamate gangli della base [27, 50]. La corteccia cerebrale è filogeneticamente la parte più giovane dell'encefalo. La superficie esterna appare irregolare, caratterizzata da fessure (solchi) e rilievi (giri); i solchi più profondi sono detti scissure e permettono di suddividere la superficie della corteccia in regioni chiamate lobi (vedi Figura 2.2) [23].

La corteccia è preposta a un elevato numero di funzioni, tra cui funzioni sensorie, motorie, cognitive, affettive e mnemoniche. Più precisamente, separando l'area della corteccia cerebrale in lobi, è possibile identificare aree funzionali all'interno di ciascuno lobo [21]:

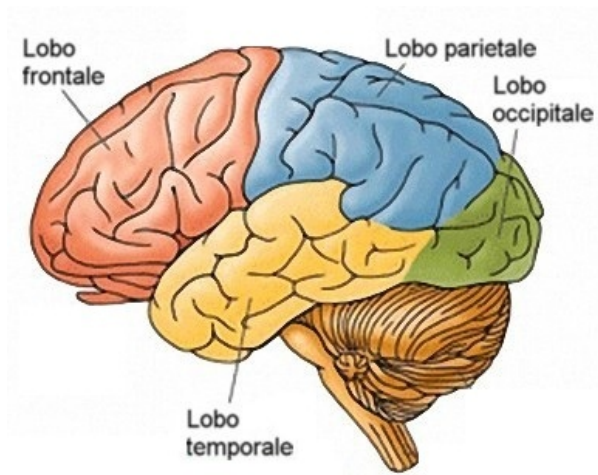


Figura 2.2: Suddivisione della corteccia in lobi

Lobo frontale: l'area compresa tra la parte anteriore del cervello e il solco centrale contiene le aree funzionali preposte all'elaborazione delle informazioni motorie (corteccia motoria), visive, linguistiche e intellettive;

Lobo parietale: l'area compresa tra il solco centrale e il solco parieto-occipitale contiene le aree funzionali preposte all'elaborazione delle informazioni sensorie (corteccia somatosensoria);

Lobo temporale: l'area delimitata dal solco laterale contiene le aree funzionali preposte all'elaborazione delle informazioni uditorie (corteccia uditiva) e alla gestione della memoria;

Lobo occipitale: l'area delimitata dal solco parieto occipitale contiene le aree funzionali preposte all'elaborazione delle informazioni visive (corteccia visiva).

2.1.2 Neuroni

La materia grigia che forma la corteccia cerebrale ha uno spessore variabile da 1.5 mm a 4.5 mm e viene generalmente suddivisa in sei strati secondo la densità e la disposizione delle cellule che la costituiscono [23]. Le cellule nervose si distinguono in [1, 23]:

Neuroni: sono le cellule nervose in grado di gestire l'informazione, elaborando e trasferendo segnali elettrici; si conta che nel cervello umano vi siano oltre 10^{10} neuroni [17].

Cellule gliali: sono cellule nervose senza alcun ruolo nella comunicazione elettrica; sono più numerose dei neuroni e il loro unico compito è quello di supporto nutrizionale, ionico e meccanico ai neuroni .

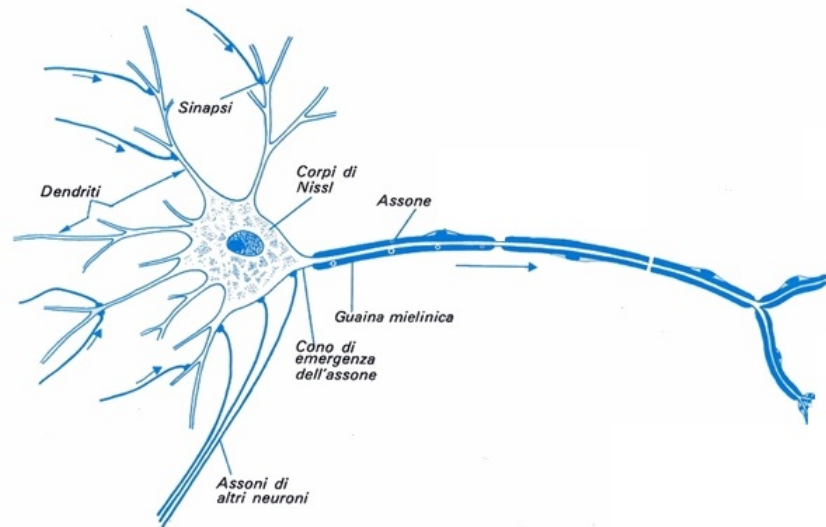


Figura 2.3: Struttura di un neurone

Sebbene sia possibile distinguere ulteriormente i neuroni in diverse sottoclassi, tutti i neuroni condividono una struttura simile. Ciascun neurone è costituito da un corpo cellulare (soma o perikaryon) da cui dipartono dei filamenti citoplasmatici (dendriti e assone). I dendriti si estendono dal soma del neurone e formano una struttura ad albero la cui funzione è quella di portare i segnali elettrici di altre cellule nervose al corpo cellulare del neurone. L'assone è, invece, un unico filamento citoplasmatico rivestito di mielina, da cui possono comunque generarsi rami secondari, responsabili per la trasmissione dell'informazione dal soma del neurone ad altre cellule nervose; l'efficienza nella trasmissione del segnale elettrico lungo l'assone è determinata dal diametro dell'assone e dal livello di isolamento garantito dal rivestimento di mielina (vedi Figura 2.3) [23]. I punti di giunzione tra due neuroni, attraverso i quali è possibile trasferire informazione da un neurone all'altro, prendono il nome di sinapsi.

2.1.3 Sinapsi

Una sinapsi è una struttura altamente specializzata che consente la comunicazione delle cellule del tessuto nervoso tra loro (neuroni) o con altre cellule (cellule muscolari, sensoriali o ghiandole endocrine). Attraverso la trasmissione sinaptica, l'impulso nervoso può viaggiare da un neurone all'altro o da un neurone ad una fibra p.es. muscolare (giunzione neuromuscolare).

La comunicazione sinaptica tra neuroni può avvenire in due modi [36]:

Comunicazione sinaptica elettrica: ha luogo in presenza di sinapsi elettriche, ovvero quei punti in cui due cellule nervose entrano in contatto diretto; la comunicazione sinaptica elettrica è veloce e adattabile e permette il passaggio diretto di impulsi elettrici in entrambe le direzioni.

Comunicazione sinaptica chimica: si realizza in presenza di sinapsi chimiche, ovvero in quei punti in cui tra due cellule vi sia un piccolo spazio; gli impulsi elettrici sono veicolati tra le due cellule per mezzo di neurotrasmettitori chimici; la comunicazione sinaptica chimica è una forma di comunicazione unidirezionale, in cui la cellula nervosa pre-sinaptica si specializza nella creazione e nel rilascio di neurotrasmettitori chimici in presenza di un impulso elettrico e la cellula nervosa post-sinaptica si specializza nel riconoscimento dei neurotrasmettitori e nella generazione di un impulso elettrico in presenza di neurotrasmettitori chimici.

Si conta che nel cervello umano vi siano migliaia di sinapsi e oltre 100 diversi tipi di neurotrasmettitori [17]. L'informazione tra i neuroni è scambiata sotto forma di impulsi elettrici, chiamati potenziali di azione. Il flusso di corrente è reso possibile dal passaggio di ioni carichi positivamente o negativamente attraverso la membrana cellulare. Il potenziale di azione è una breve inversione del potenziale di membrana di una cellula nervosa della durata di meno di 1 ms [37].

2.2 Monitorare l'attività cerebrale

Lo studio dell'attività cerebrale ha interessato gli studiosi fin dal primo '900. Esistono numerosi metodi per monitorarla: dall'elettroencefalografia (EEG) a metodi d'indagine più sofisticati e di imaging.

La magnetoencefalografia (MEG) è una tecnica che consente di studiare la funzionalità cerebrale tramite la misura del campo magnetico generato dalle correnti intracellulari dei neuroni piramidali corticali.

La risonanza magnetica funzionale (fMRI) è una tecnica di imaging biomedico che consiste nell'uso della risonanza magnetica per valutare la funzionalità di un organo o un apparato, in maniera complementare all'imaging morfologico.

La NIRS (in inglese Near Infrared Spectroscopy) è una tecnica non invasiva in grado di misurare l'ossigenazione dei tessuti attraverso la radiazione nel vicino infrarosso (700-950 nm); l'assorbimento di questa radiazione da parte del tessuto biologico è basso ed è principalmente dovuto all'emoglobina; l'emoglobina ossigenata e l'emoglobina deossigenata hanno spettri di assorbimento differenti e questa caratteristica permette di misurare la concentrazione d'ossigeno nei tessuti studiati, input di un sistema di BCI [35].

Nel paragrafo seguente è esposto il metodo che riguarda più da vicino il lavoro di tesi: l'elettroencefalografia.

2.2.1 L'elettroencefalografia

L'elettroencefalografia (EEG) è la registrazione dell'attività elettrica dell'encefalo. La tecnica è stata inventata nel 1929 da Hans Berger, il quale scoprì che vi era una

differenza di potenziale elettrico tra aghi infissi nello scalpo oppure tra due piccoli dischi di metallo (elettrodi) nel momento in cui erano posti a contatto sulla cute sgrassata del cuoio capelluto.

L'elettroencefalografia è una misura del flusso di corrente extracellulare che è generato dalla somma delle attività di un elevato numero di neuroni. I potenziali di superficie sono, principalmente, il risultato dell'attività dei neuroni corticali piramidali disposti in corrispondenza dell'area corticale sottostante l'elettrodo. L'EEG rileva la differenza di potenziale elettrico tra un elettrodo attivo, posto sopra la sede, dove si svolge l'attività neurale, e un elettrodo indifferente, collocato a una certa distanza dal primo. Un EEG misura quindi la differenza di potenziale tra aree dello scalpo ed è proporzionale alla corrente che scorre nella corteccia cerebrale durante l'eccitazione sinaptica dei dendriti di molti neuroni piramidali, che giacciono proprio sotto il cranio. Il contributo elettrico di ciascun neurone corticale è straordinariamente piccolo e il segnale deve attraversare diversi strati di tessuto non neurale, incluse le meningi (il sistema di membrane che protegge l'encefalo), i liquidi che si trovano tra le meningi, le ossa del cranio e la pelle, prima di raggiungere gli elettrodi. Di conseguenza, sono necessarie migliaia di neuroni attivati contemporaneamente per generare un segnale EEG abbastanza grande da poter essere rilevato. Questo fenomeno ha un'interessante conseguenza sull'ampiezza del segnale EEG. Il segnale dipende in grande misura da quanto è sincronizzata l'attività dei neuroni implicati. Infatti, se ciascuna cellula riceve la stessa quantità di eccitazione, ma in tempi diversi, i segnali sommati sono esigui e irregolari. Se tutte le cellule ricevono, invece, la stessa eccitazione contemporaneamente, i singoli segnali possono sommarsi, dando origine ad un campo elettrico più intenso. In questo caso il numero di cellule attivate e la quantità totale di attivazione possono rimanere invariate, ciò che cambia è solo la sincronizzazione dell'attività.

Per facilitare il contatto con l'elettrodo e per aumentare il livello di conducibilità è spesso utilizzato un gel conduttore abrasivo che è applicato direttamente sul cuoio capelluto. Per ricostruire in maniera accurata l'attività corticale, gli elettrodi sono applicati sullo scalpo secondo la disposizione standard chiamato sistema internazionale 10-20 (vedi Figura 2.4). 10% oppure 20% si riferisce alla distanza tra due punti di riferimento cranici: inion (prominenza alla base dell'osso occipitale) e nasion (attaccatura superiore del naso). Questa distanza di solito va da 30 a 36 cm con grande variabilità interpersonale (Figura 1.4).

Vengono collocati da 10 a 20 elettrodi e una massa, lungo cinque linee: longitudinale esterna longitudinale interna di destra, centrale, longitudinale esterna e longitudinale interna di sinistra. La linea trasversa T4-C4-Cz-C3-T3 (risultante delle precedenti) anch'essa deve seguire la regola del 10-20%. Gli elettrodi fronto-polari sono collocati al 10% (3-4 cm) della distanza I-N, sopra la sopracciglia, i frontali vengono collocati sulla stessa linea dei fronto-polari, più sopra del 20%, poi vengono i centrali (+ 20%), infine i parietali (+ 20%) e gli occipitali (+ 20%), con questi si arriva al 90% della distanza nasion-inion, a una distanza del 10% dall'inion. Alla posizione che ogni elettrodo occupa sullo scalpo fa riferimento una sigla. Le sigle che individuano

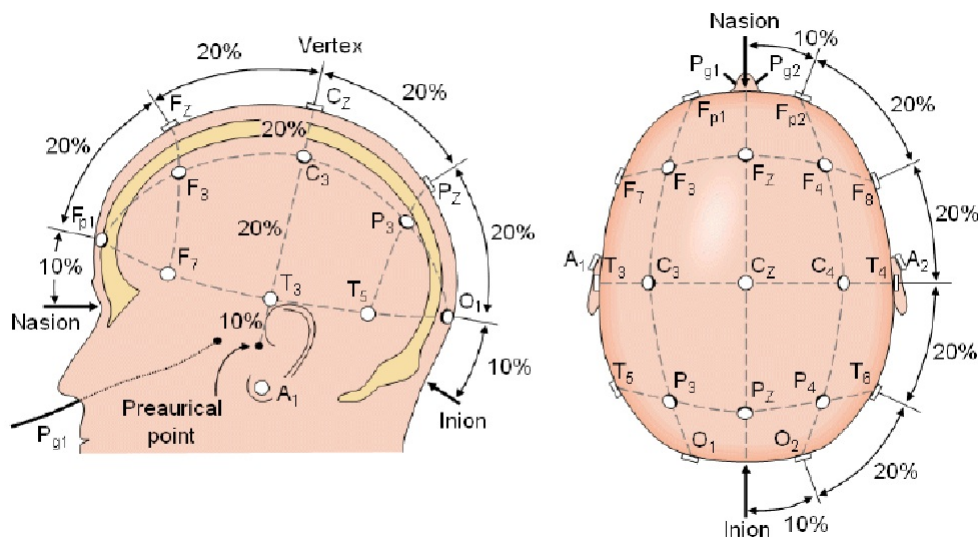


Figura 2.4: Disposizione degli elettrodi nel sistema internazionale 10-20

la posizione di un elettrodo sono formate da una/due lettere, che permettono di identificare la regione della corteccia esplorata (Fp: frontopolare; F: frontale; C: centrale; P: parietale; T: temporale; O: occipitale) e da un numero (o una z) che identifica l'emisfero (numeri dispari: sinistra; numeri pari: destra; z: linea mediana).

Per quanto riguarda il riferimento possono essere impiegate due tecniche: metodo del riferimento comune e del riferimento medio. Con il primo i potenziali di tutti gli elettrodi sono misurati rispetto a quelli di un singolo elettrodo comune, solitamente attaccato al lobo di un orecchio. Lo svantaggio in tal caso è la possibile presenza di attività vicina all'elettrodo, potenzialmente distorta per la registrazione. Col secondo metodo invece il valore di ogni canale corticale è dato con riferimento alla media di tutti i canali risolvendo così il problema del primo metodo, ma ottenendo così pattern spaziali che tendono inevitabilmente a influenzarsi.

2.2.2 Analisi di un tracciato EEG

I parametri fondamentali dell'EEG sono la frequenza (misurata in Hz, numero di onde al secondo) e l'ampiezza (misurata in μV) delle oscillazioni di potenziale, o ritmi EEG. Secondo tali parametri si distinguono onde a diversa frequenza e ampiezza: alfa, beta, delta e teta. La variazione di tali onde si correla specificamente a eventi fisiologici (attività, concentrazione, sonno, stimolazioni sensoriali etc.) e patologici (tumori, ematomi, epilessia etc.). L'interpretazione di un tracciato EEG si basa sullo studio dei suddetti parametri.

I ritmi EEG possono essere suddivisi generalmente in cinque categorie distinte (vedi Figura 2.5):

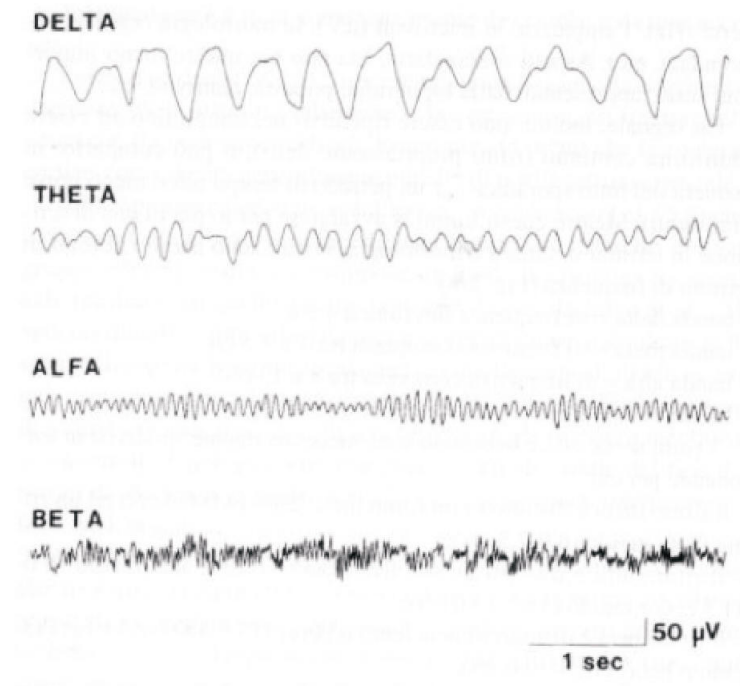


Figura 2.5: Ritmi EEG

Ritmi Gamma (γ): frequenza associata maggiore di 30Hz con ampiezza tra 1 e 20 μV ; indicano una corteccia altamente attivata; durante l'attività mentale intensa presentano l'ampiezza minima.

Ritmi Beta (β): frequenza associata tra 14 e 30Hz con ampiezza tra 1 e 20 μV ; indicano anch'essi una corteccia altamente attivata e sono perlopiù concentrati nell'area frontale e centro-parietale.

Ritmi Alpha (α) o di Berger: frequenza associata fra gli 8 e i 13Hz con ampiezza tra i 10 e i 200 μV ; sono associati a stati di veglia rilassata (lobi parietali ed occipitali).

Ritmi Theta (θ): frequenza associata fra i 3 e i 7Hz con ampiezza dell'ordine dei 5-100 μV ; si registrano durante alcuni stati di sonno.

Ritmi Delta (δ): frequenza associata tra 1Hz e 4Hz, ampiezza elevata dell'ordine dei 20-200 μV ; sono associati a stati di sonno profondo o a condizioni patologiche quali coma e forme tumorali.

In stato di veglia le zone corticali primarie motorie o sensitive visualizzano tipicamente l'attività ritmica di EEG con una frequenza bassa di 8-12 Hz; quest'attività, denominata ritmo Mu(μ) [50] è distinta dal ritmo alfa visivo. Numerose analisi computerizzate hanno indicato come il ritmo μ sia presente in una grande maggioranza di soggetti adulti. Tali analisi inoltre hanno indicato come quest'attività contenga una varietà di ritmi differenti tra gli 8-12 Hz, distinti gli uni dagli altri in termini

di zona di attivazione, frequenza e relazioni con input sensoriali e output motori. Si pensa che la modulazione di tale ritmo rifletta l'output elettrico risultato dell'attività di grandi porzioni di neuroni piramidali della corteccia motoria che controllano il moto articolare. In generale i ritmi ad alta frequenza e bassa ampiezza sono associati a stati di vigilanza e di veglia poiché quando la corteccia è fortemente impegnata nell'elaborazione delle informazioni, il livello di attività dei neuroni corticali è relativamente alto ma anche relativamente desincronizzato su ampie zone della corteccia. Al contrario i ritmi a bassa frequenza e grande ampiezza sono associati a stati del sonno o a uno stato patologico. In tal caso i neuroni corticali non sono implicati nell'elaborazione delle informazioni e un grosso numero di essi è ritmicamente eccitato da un input comune, lento e fasico consentendo di avere una alta sincronia che di conseguenza determina un'ampiezza EEG alta.

2.2.3 Potenziali evocati evento correlati e P300

Una dell'attività elettriche rilevabili tramite elettroencefalografia sono i potenziali evento correlati la cui individuazione, e analisi, pone le fondamenta della ricostruzione dell'attività corticale di un soggetto sottoposto a stimolazione esterna. Il primo passo dell'analisi consiste nel separare la parte dell'EEG correlata con il fenomeno (EEG evento-correlato) da quella parte che consegue a un'attività indipendente da quella sotto osservazione (EEG di base), a tal fine occorre quindi fornire un certo numero di stimolazioni, registrare un breve tratto di attività elettrica, che segue lo stimolo, ed effettuare una media su tutti i segnali registrati; in questo modo si potrà mettere in evidenza l'attività EEG sincronizzata con lo stimolo rispetto a quella spontanea che tenderà ad annullarsi con l'operazione di media.

La denominazione di potenziali correlati ha bisogno di essere specificata meglio attraverso l'introduzione di due nuove classi di segnale:

- Potenziali evocati stimolo correlati
- Potenziali evento correlati o ERP (Event Related Potential)

La differenza principale tra questi due tipi di potenziale si traduce nel fatto che mentre i primi dipendono dalle caratteristiche fisiche di uno stimolo esterno (quale può essere un flash luminoso, un suono, una sensazione tattile), i secondi sono evocati quando al soggetto è richiesta l'attivazione di specifiche funzioni cognitive. In questo secondo caso tali potenziali sono dipendenti dal contenuto informativo dello stimolo e compaiono solo quando il soggetto presta attenzione a quest'ultimo e vi attribuisce un "significato". Una delle principali caratteristiche di tali potenziali è sicuramente la stretta relazione temporale tra la stimolazione e la risposta allo stimolo stesso.

Un ERP può essere, in particolare, visto come il risultato di un rifasamento delle oscillazioni di fondo rispetto all'evento di interesse: le variazioni prodotte dall'evento esterno (sia esso uno stimolo visivo o acustico ad esempio) avvengono sempre ad un distanza fissa nel tempo (latenza) rispetto all'evento di interesse. Tale tipo di attività

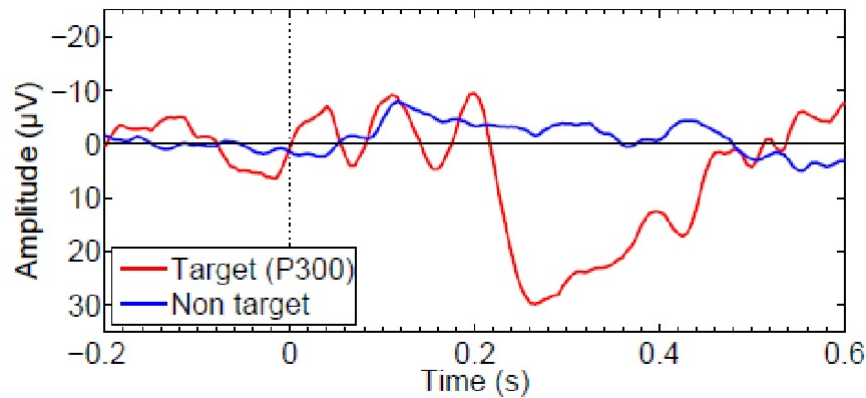


Figura 2.6: Grafico di una P300, ottenuto dalla media di molte ripetizioni

generalmente si riscontra nello spettro che va dai 4-6 Hz ai 60-70 Hz ed è costituita da oscillazioni di potenziale elettrico e da forme d'onda caratterizzate da una serie di deflessioni positive o negative.

In questo ambito, ricopre una notevole importanza la componente P300, un potenziale positivo (maggiore di $10 \mu\text{V}$) endogeno che per definizione compare solamente in seguito a stimoli “target” (vedi Figura 2.6). Si tratta di un fenomeno elettrofisiologico che si colloca al termine di una complessa sequenza di elaborazione cognitiva, attivata dal paradigma di stimolazione e conclusa periodicamente dalla presentazione dello stimolo significativo. La latenza della P300 esprime il tempo impiegato dal soggetto per completare il pieno riconoscimento dello stimolo atteso (il picco di potenziale si evidenzia dopo circa 300 ms dalla presentazione dello stimolo, da cui il nome). L'ampiezza, invece, è funzione inversa della probabilità di comparsa (sia oggettiva che soggettiva) dello stimolo significativo e dalla quantità di informazione da esso trasmessa al soggetto.

Il paradigma “oddball” è quello più utilizzato per indurre tale potenziale, esso è composto da un treno di stimoli standard (detti anche stimoli “non target”) all'interno del quale vengono inseriti degli stimoli rari (“target”). Tale paradigma genera di conseguenza un potenziale P300 nella zona parietocentrale dello scalpo nel momento in cui si distingue lo stimolo target da quello standard. L'ampiezza dell'onda P300 cresce se diminuisce la probabilità di presentazione dello stimolo target. Per quanto riguarda la latenza invece essa cresce quando i target sono più difficili da discriminare dai no-target.

2.3 Brain-Computer Interface

Un'interfaccia cervello-computer (in inglese, Brain-Computer Interface, abbreviato in BCI, o anche Brain-Machine Interface, BMI) è un sistema di comunicazione in cui

i messaggi o i comandi che l'utente normalmente invia per agire sul mondo esterno non passano attraverso la canonica via costituita dai nervi periferici e dai muscoli, bensì usa come codificatore e attuatore il computer e i dispositivi ad esso collegati. Per far ciò, è necessaria un'analisi numerica che interpreti l'attività cerebrale letta dall'interfaccia.

2.3.1 Classificazione delle BCI

Una prima distinzione delle BCI prevede due classi: BCI dipendenti e indipendenti.

- Le BCI dipendenti non usano la normale via neuromuscolare, ma utilizzano vie neuromuscolari alternative, senza la necessità quindi di bypassare completamente il tratto muscolare periferico; essenzialmente sostituiscono una via interrotta o deteriorata con un'altra preesistente, che serva per generare l'attività cerebrale che codifica il messaggio da inviare al computer. È il caso, ad esempio, dei sistemi di BCI caratterizzati da una matrice di lettere che s'illuminano una alla volta: il soggetto seleziona la lettera scelta allineandovi lo sguardo, così da elicitare un potenziale evocato visivo (VEP), che risulta essere più ampio dei potenziali evocati dalle altre lettere [16].
- Le BCI indipendenti non dipendono in alcun modo dalle normali vie d'uscita neuromuscolari. Il messaggio in questi sistemi non è convogliato in vie neuromuscolari o sensorie, e non è necessario l'intervento di queste vie per far emergere l'attività cerebrale necessaria alla codifica del messaggio. È il caso, ad esempio, dei sistemi di BCI in cui tramite una matrice di lettere che si illuminano in modo sequenziale viene letto il potenziale evocato P300, correlato e sincronizzato con la scelta del soggetto [19, 12]. In questo caso, l'output registrato, utile alla codifica del messaggio, è il segnale EEG, che non è veicolato da nessuna via neuromuscolare periferica, ma dipende solo dalle intenzioni dell'utente, e non dalla direzione dello sguardo [13, 48, 41]. Bypassando l'attività periferica, i sistemi di BCI indipendenti risultano di maggior interesse teorico sulle funzioni del cervello rispetto ai sistemi BCI dipendenti, e di maggiore utilità, specialmente per pazienti con più gravi disabilità neuromuscolari.

L'altra classica distinzione è tra le BCI invasive e non invasive, originate da tradizioni di ricerca, discipline e soprattutto obiettivi diversi.

- Le BCI invasive consistono essenzialmente in matrici di elettrodi impiantate nella corteccia motoria o premotoria di soggetti paralizzati [14]. Ereditando l'approccio robotico, ovvero l'uso di esecutori mecatronici direttamente interfacciati al soggetto, si basano sull'ipotesi che le abilità motorie possano essere ricostruite a partire dalla codifica diretta dei segnali neuronali indirizzati ai tratti periferici neuromuscolari [25]; sviluppate nella pratica clinica, sono state un tentativo di sostituzione totale o parziale di funzioni motorie, principalmente a livello di arti superiori [32].

Le BCI invasive richiedono l'impianto di microelettrodi tramite intervento chirurgico; se l'attività che si vuole registrare comprende un'area cerebrale ampia, possono essere utilizzate delle vere e proprie griglie applicate in porzioni d'interesse sulla corteccia [28]; l'utilizzo di una matrice di elettrodi consente la registrazione del segnale su una superficie di corteccia maggiore: questo permette di distinguere diversi pattern d'attivazione corticale e di utilizzare quest'informazione come punto di partenza per una BCI. La necessità di un rischioso e complesso intervento chirurgico attualmente rende auspicabile la scelta di una BCI non invasiva, se non per soggetti con disabilità estremamente gravi [51].

- Le BCI non invasive riprendono e fanno proprie le tradizioni dell'elaborazione dei segnali, dell'intelligenza artificiale e della neuropsicologia come basi teoriche e strumenti tecnologici per il controllo e la comunicazione tra cervello e macchina, non più semplice dispositivo esecutore del movimento, ma interprete delle intenzioni del soggetto, fino alle più alte funzioni cognitive e di pianificazione [8].

L'intento è usare i segnali definiti di neurofeedback, segnali di condizionamento del sistema nervoso centrale (CNS), tramite cui è possibile apprendere e/o modificare funzioni motorie, senza alcuna mediazione del sistema somatico-muscolare [34]. L'impulso più importante lo hanno dato: la scoperta dell'autoregolazione delle onde cerebrali [26], ovvero la modifica di caratteristiche del segnale EEG a seconda del comportamento, delle condizioni ambientali e dell'addestramento del soggetto, senza la mediazione delle vie periferiche somatiche e neuromuscolari; la scoperta dell'apprendimento e di una forma di controllo volontario nei segnali cerebrali dopo una fase di training [31].

Nella pratica clinica, l'uso di BCI non invasive è stato quindi proposto inizialmente come alternativa, fino a diventare una delle poche (se non l'unica) soluzione per quei soggetti affetti da stati patologici (SLA, estese lesioni cerebrali, Sclerosi Multipla e altre) che portano alla condizione di locked-in state, ovvero una condizione in cui si ha un rudimentale controllo di al massimo un muscolo, fino a condizioni estreme in cui si ha la totale scomparsa di risposta muscolare, condizione definita come completely locked-in state [4].

2.3.2 Le componenti di una BCI

Nella costruzione di un sistema di BCI è necessario fare delle scelte preliminari, a cominciare dal segnale biologico da acquisire, passando per la tecnica di acquisizione, la modalità con cui il segnale biologico viene elicitato, per finire con la registrazione ed elaborazione del segnale e il metodo di analisi e traduzione del segnale. Due sono le componenti essenziali di cui necessita ogni BCI [18, 51]:

1. Un sistema hardware dedicato alla presentazione degli stimoli sensoriali, all'acquisizione del segnale EEG (elettrodi, amplificatore, filtri, ecc.), alla registrazione di risposte comportamentali e, facoltativamente, alla presentazione di un feedback (visivo, acustico, tattile);

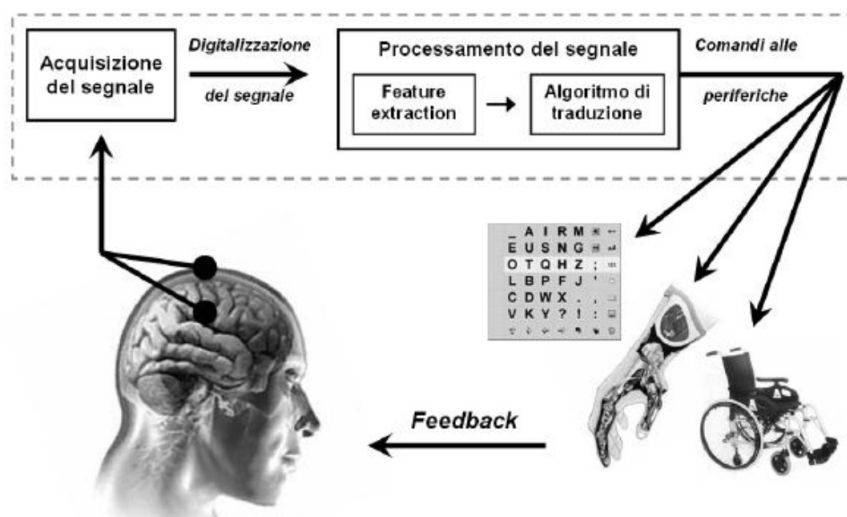


Figura 2.7: Struttura di una BCI

2. Un sistema per l'elaborazione e la classificazione del segnale.

Nello specifico, essendo un sistema di comunicazione e controllo, una BCI ha: un ingresso, un'uscita, delle componenti per convertire l'ingresso in uscita e, nel caso di BCI sincrone, un protocollo che definisce il timing delle operazioni.

I blocchi fondamentali sono i seguenti (vedi Figura 2.7) :

1. **Acquisizione del segnale:** in questo caso vanno sottolineate le differenze tra segnali di risposte evocate e attività spontanee; le risposte evocate (come la P300) sono risposte del cervello a stimoli sensori stereotipati provenienti in questo caso dal sistema di BCI, quindi l'onset è a carico del sistema di acquisizione; l'attività spontanea (come i ritmi EEG) non dipende dalla generazione di stimoli da parte del sistema di BCI, quindi l'onset è dato dall'utente; nella parte di acquisizione del segnale, l'input scelto è acquisito dagli elettrodi, amplificato e digitalizzato.
2. **Processing del segnale:** le BCI possono usare caratteristiche del segnale nel dominio del tempo o della frequenza, oppure usare entrambi i domini, soluzione che potrebbe portare a un miglioramento delle performance [44]; in genere, possiamo dire che gli attuali BCI estraggono e riconoscono caratteristiche che riflettono eventi cerebrali ben definiti e identificabili (particolari onde o ritmi); è possibile per una BCI usare caratteristiche che non siano strettamente o direttamente correlate a uno specifico evento cerebrale, ma che comunque risultino di rilevanza per stabilire il momento e il tipo di intento dell'utente; in questo caso, è particolarmente importante garantire che la caratteristica scelta non sia contaminata da EMG, EOG o altri artefatti del Sistema Nervoso Centrale.

3. L'algoritmo di traduzione: la fase successiva è interessata alla traduzione delle caratteristiche, estratte o riconosciute, in comandi ai dispositivi a valle, in modo da poter esprimere in maniera corretta l'intento dell'utente. Questo algoritmo può usare metodi lineari (la classica analisi statistica), o non lineari (ad esempio le reti neurali); qualunque sia la sua natura, ogni algoritmo ha come obiettivo la traduzione dei segnali elettrofisiologici in segnali di controllo. L'efficienza dell'algoritmo ad adattarsi a ogni singolo utente si può valutare su tre livelli:
 - a) Al primo accesso del nuovo utente, l'algoritmo si adatta alle particolari caratteristiche del segnale; come esempio concreto, parlando di BCI basate su P300, l'algoritmo si adatta alle caratteristiche di ampiezza e latenza della P300 dell'utente connesso; una BCI che ha solo questo livello di adattamento, continuerà ad avere performance costanti se e solo se le risposte ai segnali da parte dell'utente resteranno costanti;
 - b) Poiché i segnali EEG e altri segnali elettrofisiologici mostrano in genere variazioni a breve e lungo termine, legati ai ritmi circadiani e alle condizioni fisiche, ormonali o ambientali, è necessario che sia presente un secondo livello di adattamento, ovvero un periodico adeguamento per ridurre l'impatto di variazioni spontanee; un buon algoritmo di traduzione deve essere in grado di adattarsi in modo da ottenere la massima corrispondenza tra l'attuale range di valori delle caratteristiche di segnale dell'utente e il range di valori disponibili dei dispositivi da controllare.
 - c) L'ultimo livello di adattamento è in realtà l'adattamento auspicabile per ogni sistema di BCI, ovvero riflettere le capacità adattative del cervello e rispondervi in modo rapido e affidabile: il cervello così potrà modificare il segnale d'uscita in modo da migliorare le operazioni BCI, come se fosse un canale neuromuscolare, ottenendo non solo un costante miglioramento delle performance del sistema di BCI, ma incoraggiando l'utente a sviluppare una nuova abilità e, in ultima istanza, un nuovo metodo di comunicazione.
4. Il dispositivo d'uscita: per la maggior parte delle attuali BCI, il dispositivo d'uscita è il monitor di un computer e l'uscita (auspicabile) è la selezione di un obiettivo, una lettera o un'icona presente su di esso; in altre BCI l'uscita è il movimento di un cursore verso l'alternativa prescelta. L'uscita è anche un feedback che l'utente usa per testare la capacità del sistema di rispondere al suo intento, espresso con le caratteristiche del segnale che la BCI traduce in comandi.
5. Il protocollo operativo: ogni BCI ha un protocollo che guida le operazioni, pianificate in modo specifico in relazione all'interfaccia o allo stimolo presentato all'utente, al segnale registrato e alla tipologia di BCI con cui si lavora, che definisce l'interazione tra utente e sistema e quale feedback è restituito all'utente, ammesso che esso sia presente.

2.4 Controllo asincrono delle BCI

Un importante aspetto delle BCI basate su EEG è il concetto di controllo asincrono; generalmente le BCI che ne fanno uso sono dette asincrone (*asynchronous BCI* o *self-paced BCI*), mentre quelle che non lo utilizzano sono dette sincrone (*paced BCI* o *cue-based BCI*).

Bisogna distinguere due stati [43]:

- il primo è lo stato di controllo (o *intentional control state*, IC), nel quale l'utente intende utilizzare l'interfaccia che, quindi, deve essere attiva;
- il secondo è lo stato di non controllo (*noncontrol state*, NC), nel quale l'utente non intende utilizzare l'interfaccia che, pertanto, non deve restituire alcun output.

Sistemi di BCI sincroni: in questa tipologia di BCI il soggetto è vincolato a inviare i propri comandi al computer soltanto nei momenti “sincronizzati” con l'attività di stimolazione esterna. Una BCI sincrona assume che l'utente sia sempre in stato di controllo e pertanto è costantemente impegnata nella traduzione simultanea di segnali cerebrali in comandi: se l'utente non intende controllare l'interfaccia, quest'ultima deve essere spenta per evitare false interpretazioni del segnale cerebrale e di conseguenza false azioni di controllo.

Sistemi di BCI asincroni: in questo secondo caso l'utente può comunicare con l'elaboratore in un qualsiasi momento a suo piacimento, senza alcuna restrizione temporale. Una BCI asincrona è in grado di far passare liberamente l'utente da uno stato di controllo a uno di non controllo senza l'aiuto di un input esterno. La sfida principale, in questo caso, consiste nel discernere fra gli stati mentali voluti dall'utente per segnalare uno specifico compito o comando per l'elaboratore, e gli stati non intenzionali potenzialmente interpretabili come falsi positivi, generando così comandi non voluti dal soggetto ma rilevati dall'elaboratore.

La distinzione tra interfacce cervello-computer sincrone e asincrone è importante per poter valutare l'usabilità di tali dispositivi nel mondo reale; è infatti chiaro che una BCI di entrambi i tipi può funzionare senza problemi in un ambiente di test come un laboratorio, ma è altrettanto evidente che nella vita quotidiana risulta molto più sensato averne una asincrona che autonomamente riconosca le intenzioni di un paziente in qualsiasi istante questo le voglia esprimere.

2.5 Stato dell'arte

2.5.1 BCI P300

La P300 è un potenziale positivo, ampio, studiato in forma estensiva nel contesto del paradigma oddball.

La prima BCI che abbia utilizzato la P300, chiamata P300-speller, è stata sviluppata da Farwell e Donchin. Essa prevedeva una matrice 6×6 di simboli (26 lettere dell'alfabeto e alcuni comandi funzionali), di colore grigio su sfondo nero, le cui colonne e righe erano alternativamente e in modo casuale rese più luminose: una P300 veniva prodotta quando la colonna o la riga del simbolo scelto erano illuminate; il simbolo scelto era selezionato tramite la combinazione delle risposte su righe e colonne [19].

Anche se il paradigma che utilizza una matrice di lettere e simboli è stato quello più esplorato ed indagato nel campo delle P300-BCI, sono stati proposti anche altri tipi di interfaccia. Alcuni esempi interessanti sono quelli riportati nei lavori di Bayliss e Ballard con un ambiente virtuale (un appartamento con oggetti): le prestazioni ottenute sono in linea con i precedenti lavori sulla P300-BCI [2].

Un metodo alternativo, sviluppato da Sellers e Donchin, [46] prevede un'interfaccia a quattro frecce direzionali o quattro scelte (nel caso specifico: "YES", "NO", "PASS", "END") ai lati dello schermo, con il classico lampeggiamento casuale come stimolo; per la P300-BCI con questa interfaccia, sono state confrontate le prestazioni (su pazienti malati di SLA) tra una modalità visiva, una modalità acustica e una terza modalità che prevedeva la simultanea presentazione di stimoli acustici e visivi [46]. Le conclusioni principali di questo lavoro sono state che una BCI P300 può essere usata per pazienti con SLA e che sia stimoli visivi che uditivi elicitano la P300.

2.5.2 BCI Self-paced

Gli ultimi anni hanno visto un incremento dell'interesse di ricerca nel campo del controllo asincrono.

Le tecnologie elencate in questa sezione non verranno esplicitate, pertanto, per eventuali approfondimenti, si rimanda alla bibliografia indicata.

Un possibile approccio è quello di Mason e Birch [33], che utilizzano un sistema di cambiamento di stato chiamato LF-ASD (*low-frequency asynchronous switch design*). Questo sistema è basato su un set di feature connesso al *motor imagery* in un intervallo di frequenza tra 1-4 Hz.

Un altro lavoro interessante è quello di Scherer, Schloegl e Lee che hanno sviluppato la Graz BCI [45]. Questa BCI è in grado di effettuare un controllo asincrono per mezzo di ritmi indotti dalla motor imagery. L'interfaccia è testata con una applicazione-gioco in cui i soggetti devono raccogliere delle monete esplorando un ambiente virtuale.

Un metodo differente è invece quello utilizzato da Faradji, Ward e Birch: in questo caso la BCI è basata sui *mental task*. L'output della BCI è attivato solo quando il soggetto è impegnato in un determinato mental task, mentre resta inattiva per gli altri.

2.5.3 BCI P300 Self-paced

Come detto in precedenza, gli studi nel campo delle BCI self-paced basate su ERP sono ancora scarsi.

Un primo tentativo di creazione di una BCI P300 con controllo asincrono è stato fatto da Zhang, Guan e Wang nel giugno 2008 [54]. In questo lavoro il segnale EEG è acquisito da 15 canali (F,Fc,C,CP,P lungo le posizioni emisferiche 3,z,4), elaborato al fine di trovare dei vettori di feature (tramite segmentazione nel dominio del tempo e della frequenza) ed infine classificato tramite SVM. Le prestazioni riportate, misurate in ITR (*information transfer rate*), sono di approssimativamente 20 bit/min con un tasso di falsi positivi di 1 evento/min.

Il significato di questa misura di prestazioni sarà esplicito nella sezione 4.6.

Di recente (giugno 2011) è stata proposta da Panicker, Puthusserypady e Ying Sun, una BCI che combina onde P300 e SSVEP (*steady-state visually evoked potentials*) [38]. Gli SSVEP sono segnali di risposta del cervello a stimoli visuali a specifiche frequenze.

Il sistema base è basato sul segnale P300 mentre la discriminazione tra gli stati di controllo e non controllo è effettuata tramite SSVEP.

L'accuratezza della discriminazione tra gli stati riportata è di circa l'88%, con un ITR di 19.05 bits/min.

3 Materiali e metodi

La parte sperimentale del lavoro di tesi può essere suddivisa in due parti distinte: una parte è costituita dall'attività in laboratorio, in cui si è proceduto all'acquisizione dei dati necessari, un'altra è costituita dalla loro successiva elaborazione.

Gli esperimenti sono stati da noi condotti all'interno dei laboratori del Politecnico di Milano, in particolare parte delle acquisizioni sono avvenute nel Laboratorio di Intelligenza Artificiale e Robotica e parte nel Laboratorio di Bioingegneria. Essi sono consistiti nell'utilizzo, da parte di volontari sani, dell'interfaccia cervello-computer e nell'acquisizione dei relativi dati elettroencefalografici.

Una volta acquisiti i dati si è proceduto a una prima elaborazione con il software già in dotazione al Politecnico, che permette di effettuare una fase di training e una di test al fine di addestrare la BCI.

L'output di questo software è utilizzato come base di partenza per la realizzazione del filtro asincrono oggetto della tesi. Nella Figura 3.1 è possibile vedere lo schema a blocchi del sistema.

Nel presente capitolo si espongono innanzitutto le configurazioni sperimentali e i protocolli di laboratorio utilizzati per la fase sperimentale; in seguito, l'attenzione è volta alla descrizione del software prodotto.

3.1 Acquisizione

3.1.1 Hardware e software

Per la parte di acquisizione del segnale si è fatto uso dell'hardware e del software utilizzato per i precedenti studi del Politecnico di Milano riguardanti lo Speller P300.

La componente principale dell'hardware di acquisizione EEG è l'elettroencefalografo che rileva l'attività elettrica cerebrale proveniente dagli elettrodi, la amplifica e la invia al computer. Nel nostro caso, l'unità è l'EEG Digital BE Light di EBNeuro

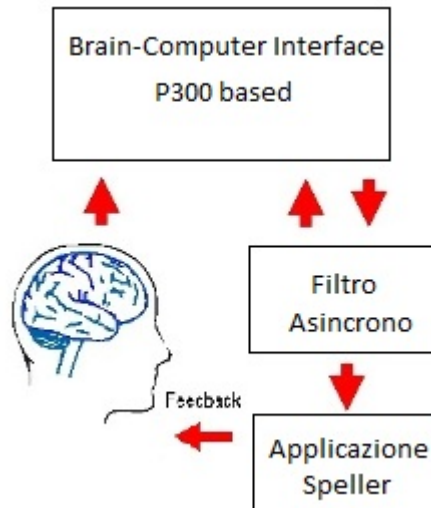


Figura 3.1: Schema a blocchi del sistema

mostrata in Figura 3.2. Ha 28 canali: 21 mono-polari, 4 bipolari, e 3 mono-polari con un riferimento separato. La frequenza massima di campionamento è di 8 kHz. L'unità è collegata a un comune computer portatile attraverso una fibra ottica e un adattatore PC Card, vedi Figura 3.3.

L'uso della fibra ottica isola il computer dall'amplificatore e dal soggetto. Il computer è utilizzato anche per stimolare il soggetto tramite lo Speller P300 e si trova molto vicino a lui. Per questo motivo, il computer è scollegato dalla presa di corrente e funziona con la sua batteria interna per tutta la durata degli esperimenti. Il BE Light applica un filtro passa-alto a 0,1 Hz a tutti i canali EEG ed EOG. Frequenze superiori a 1 kHz vengono anche esse rimosse da un filtro anti-aliasing. L'amplificatore non è stato impostato per rimuovere le interferenze di linea a 50 Hz, ma gli esperimenti hanno dimostrato che tenere cavi di alimentazione, prese e adattatori AC ad almeno 50cm - 1 metro di distanza dal soggetto è sufficiente per avere solo trascurabili effetti sulla qualità del segnale.



Figura 3.2: Elettroencefalografo utilizzato negli esperimenti



Figura 3.3: Laptop utilizzato per le acquisizioni

E' stata utilizzata una frequenza di campionamento di 512 Hz per tutti i nostri esperimenti, pertanto la banda di frequenza utile in accordo alle specifiche è di 0,1-230 Hz. Poiché il segnale P300 non presenta componenti significative sopra i 20 Hz, i dati vengono ulteriormente filtrati via software.

L'amplificatore è dotato di una suite software, Galileo, che gestisce tutti i compiti inerenti l'acquisizione dell'EEG, che è stata utilizzata solo per registrare i dati ed esportarli per le ulteriori elaborazioni. Lo Speller P300 si basa su BCI2000, un generico sistema software sviluppato per la ricerca nel campo delle brain-computer interface presso il Wadsworth Center del New York State Department of Health ad Albany (USA).

3.1.2 Speller P300

Lo Speller P300 è molto semplice nell'aspetto e nel funzionamento. L'interfaccia è una griglia 6x6 le cui celle contengono le lettere dell'alfabeto, alcune cifre, lo spazio e il backspace, rappresentato come BS nell'angolo in basso a destra (vedi Figura 3.4).

I caratteri sono bianchi su sfondo nero. La griglia viene visualizzata su un display e ciascuna delle 6 righe e delle 6 colonne lampeggia una dopo l'altra in ordine casuale. La durata dello stimolo è di 125 ms e tra l'inizio di un flash e l'inizio del successivo è impostato un Intervallo Inter Stimolo (ISI) di 125ms. Ogni riga e ogni colonna lampeggiano esattamente una volta nei primi dodici stimoli (le sei righe e le sei colonne). Un blocco di 12 stimolazioni consecutive è chiamato ripetizione. Alla fine della ripetizione si riparte con altre 12 stimolazioni in un nuovo ordine casuale. Al soggetto è richiesto di focalizzare l'attenzione su una lettera e di contare il numero di volte che la riga e la colonna che la contengono si accende. Tali eventi sono

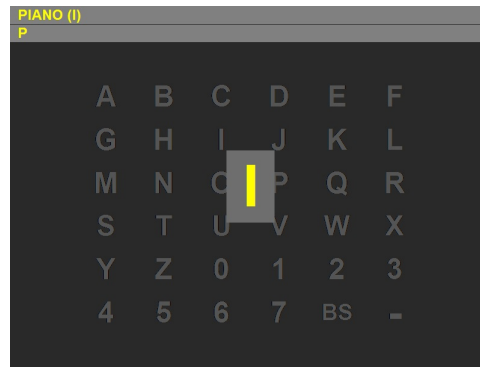


Figura 3.4: Speller P300

ovviamente gli stimoli target che elicitano la P300 e che vengono riconosciuti nelle successive fasi di elaborazione dei dati.

3.1.3 Set up

Per procedere all'acquisizione è necessaria una fase di set up del software e il montaggio, ossia il posizionamento degli elettrodi sul soggetto da testare e il collegamento dei cavi.

Nel set-up vengono impostati i parametri di base necessari per lo svolgimento dell'esperimento, tra cui la scelta delle parole da far scrivere e il numero di ripetizioni per ogni lettera.

Per il montaggio si è fatto uso di una cuffia precablata per facilitare l'operazione, ma dato che il segnale P300 è più forte sulla linea mediana della testa sono stati utilizzati soltanto i quattro canali mono-polari centrali: Fz, Cz, Pz, Oz (vedi Figura 3.5). Il riferimento è stato posizionato al mastoide destro, e un elettrodo di massa al mastoide sinistro. Infine, due elettrodi Ag/AgCl bipolari sono stati posti vicino all'occhio destro per misurare l'elettrooculogramma (EOG), cioè un potenziale correlato con il movimento degli occhi; nel canale EOG sono visibili sia i movimenti

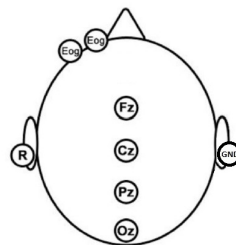


Figura 3.5: Applicazione degli elettrodi per l'acquisizione dei segnali elettroencefalografico (EEG) ed elettrooculografico (EOG),



Figura 3.6: Strumenti utilizzati per le acquisizioni

oculari sia battito delle ciglia. L'EOG non è usato in fase di classificazione, ma solo per filtrare eventuali artefatti oculari. Gli elettrodi sono stati applicati seguendo la procedura standard: sfregamento della pelle e applicazione della pasta conduttiva (alcuni strumenti utilizzati sono mostrati in Figura 3.6). Dopo il controllo dei livelli di impedenza e la calibrazione dell'encefalografo, si procede all'acquisizione vera e propria.

3.1.4 Protocollo di acquisizione

Il protocollo prevede alcune acquisizioni registrate in due giornate diverse. Questa scelta è fatta per due ragioni: la prima è mostrare la validità del training anche a distanza di tempo (anche mesi) e la seconda è che se tutto il protocollo fosse fatto in una sola giornata si andrebbe incontro alla stanchezza dell'utente con conseguente peggioramento dei risultati.

L'acquisizione è preceduta da una fase in cui si spiega brevemente al soggetto che si deve sottoporre al test il funzionamento dell'apparecchio e i task che deve eseguire per un corretto svolgimento dell'esperimento.

Si è proceduto ad acquisire dati con un numero di ripetizioni pari a 10. Il protocollo di acquisizione è stato pensato in maniera tale da permettere di acquisire le onde EEG dei soggetti sia nel caso in cui gli stessi prestavano attenzione alla griglia, e quindi di volta in volta alla singola lettera da scrivere, sia nel caso in cui, pur fissando lo Speller, erano disattenti e non cercavano di scrivere la lettera proposta. Il concetto di attenzione/disattenzione è meglio spiegato nel sottoparagrafo successivo.

La durata totale di ogni acquisizione è di circa 1 ora e 20 minuti, a cui si deve aggiungere qualche minuto di pausa facoltativo tra una parola e l'altra per far riposare l'utente e caricare il nuovo set di lettere.

Nel primo giorno, il software di acquisizione propone 10 parole diverse e permette all'utente di scrivere ogni parola segnalando di volta in volta la lettera corretta da scrivere. Data la durata del test e la difficoltà nel tenere alta la concentrazione si è scelto di sfruttare la freschezza iniziale del soggetto. In questa ottica si è scelto



Figura 3.7: Sequenza di acquisizione. Con la A sono indicate le parole in cui l'utente è attento, con la D le parole in cui l'utente è disattento, con la L le singole lettere.

di usare la seguente sequenza di attenzione/disattenzione: le prime 5 parole con soggetto attento e le restanti 5 alternando una parola con soggetto disattento ed una con soggetto attento. Ottenendo così in totale 7 parole con la registrazione delle onde EEG di un utente attento e 3 parole con la registrazione di onde di un utente disattento (vedi Figura 3.7).

Nella seconda giornata, vengono proposte sempre 10 parole, ma in questo caso si comincia fin da subito ad alternare una parola con utente attento ed una con utente disattento (vedi Figura 3.7).

Attenzione/Disattenzione

Chiariamo ora meglio il concetto di attenzione e quello di disattenzione.

Come spiegato precedentemente, il fine della tesi è quello di capire, dai dati rilevati tramite EEG, quando l'utente intende utilizzare la BCI alla quale è sempre collegato. Ossia, in ogni istante, il sistema deve essere in grado di distinguere due stati: il primo, che chiamiamo stato di attenzione, è quando l'utente vuole utilizzare la BCI (nel nostro caso vuole scrivere una lettera) e quindi si interessa all'interfaccia che ha davanti a sé; il secondo, detto di disattenzione, è relativo a quando l'utente non vuole utilizzarla (non vuole dunque scrivere nulla) e quindi non è concentrato su di essa.

Per poter trovare una soluzione al problema, vi è bisogno di disporre di registrazioni EEG relative a entrambi gli stati.

Ai soggetti che si sono sottoposti al test, si è dunque spiegato che dovevano avere due atteggiamenti diversi: per alcune parole si dovevano concentrare e cercare di scrivere la lettera proposta, e per altre parole, invece, pur fissando l'interfaccia dello Speller dovevano pensare ad altro e non dovevano cercare di scrivere. Nel caso dell'atteggiamento di disattenzione, si è detto all'utente di non leggere neppure la

lettera che veniva proposta a video in maniera tale da essere certi di non provocare P300 indesiderate sulla lettera stessa.

3.1.5 Soggetti testati

I soggetti che si sono sottoposti ai test, sono tutti sani, di sesso maschile e di età compresa tra i 20 e i 30 anni.

I soggetti sono 4 per un totale di 8 sessioni di acquisizione. In particolare questa è la lista dei soggetti con il relativo tipo di acquisizione:

Soggetto	1 ^a sessione di acquisizione	2 ^a sessione di acquisizione
sbj1	10 parole (10 ripetizioni)	10 parole (10 ripetizioni)
sbj2	10 parole (10 ripetizioni)	10 parole (10 ripetizioni)
sbj3	10 parole (10 ripetizioni)	10 parole (10 ripetizioni)
sbj4	10 parole (10 ripetizioni)	10 parole (10 ripetizioni)

3.1.6 Dati acquisiti

Riassumendo quanto detto in precedenza, per ciascuno dei soggetti, si hanno a disposizione le stimolazioni relative a 20 parole (12 attento e 8 disattento).

La lunghezza delle parole non è fissa, dato che sono state testate parole diverse, perciò anche il numero di lettere totali che si hanno a disposizione per i 4 soggetti potrebbe non essere uguale; nel seguito, però, per questioni di semplicità si considererà fisso per tutti questo numero.

La lunghezza media di ogni parola è di 10 lettere per cui possiamo dire con buona approssimazione che per ogni soggetto abbiamo a disposizione un totale di circa 200 lettere e quindi 200 set di stimolazioni. Anche in questo caso si farà riferimento ai 200 set come se fosse un numero fisso. I set in cui il soggetto era attento, sono ovviamente composti da una stimolazione relativa al target (la lettera su cui bisognava focalizzare l'attenzione) e dalle stimolazioni relative ai non-target; quando, invece, il soggetto non era attento non vi è alcun target.

I dati rilevati sono utilizzati per le successive elaborazioni, in particolare per il train e il test sia dello Speller sia del filtro a valle dello stesso, come verrà successivamente spiegato.

3.2 Train e test dello Speller

La prima elaborazione dei dati acquisiti è affidata al software sviluppato dal Politecnico di Milano che permette di effettuare il training ed il test dello Speller P300. Nel seguito verrà brevemente introdotto il funzionamento di questo software.

In primo luogo vengono analizzati i segnali EEG e si tenta di estrarre le caratteristiche peculiari del segnale: la ricerca della soluzione migliore è affidata a un algoritmo genetico progettato ad hoc per l'applicazione.

Quindi si procede alla classificazione dei segnali attraverso l'addestramento di un classificatore logistico, per mezzo delle caratteristiche trovate.

Infine si testano i risultati ottenuti.

3.2.1 Dataset

Delle 20 parole (200 set di stimolazioni), le prime 5 (tutte con soggetto attento) vanno a formare il train-set e sono pertanto utilizzate per l'addestramento dello Speller in modo tale che il relativo classificatore sia in grado di emettere uno score per ogni lettera, sperabilmente uno score maggiore per gli stimoli target e minore per gli stimoli non-target. Per quanto riguarda il test-set sono state utilizzate le parole dalla 5 alla 20. L'intero dataset e la sua suddivisione sono mostrati in Figura 3.8. Si noti che la quinta parola è stata utilizzata sia nel train che nel test dello Speller, ma il lievissimo overfit sui dati, che si crea in questo modo, è assolutamente ininfluente per il filtro a valle. Questa scelta è stata fatta per ottimizzare la quantità di dati a disposizione ed avere dei dati omogenei (8 parole in cui il soggetto era attento e 8 in cui il soggetto non lo era) in modo da addestrare e testare al meglio il filtro a valle, che sfrutterà l'output del test.

3.2.2 Algoritmo Genetico

Per estrarre delle caratteristiche utili dal segnale, si fa uso di un algoritmo genetico.

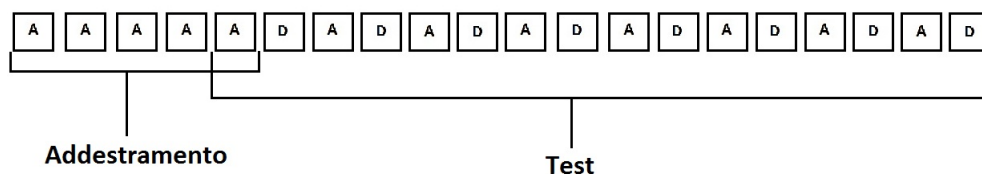


Figura 3.8: Dataset dello Speller. Le 16 parole utilizzate per il test sono le stesse che andranno a comporre l'intero dataset del filtro

Un algoritmo genetico è un algoritmo di ottimizzazione, ispirato al principio della selezione naturale di Charles Darwin che regola l'evoluzione biologica.

Le componenti chiave che definiscono un algoritmo genetico, con le soluzioni scelte, sono [22]:

Popolazione la popolazione di un algoritmo genetico in un dato istante è costituita da un insieme di individui, ciascuno dei quali rappresenta una possibile soluzione del problema;

Codifica la codifica specifica in quale modo le possibili soluzioni sono codificate nella popolazione. Ogni individuo nella popolazione è costituito da un cromosoma; ogni cromosoma contiene un numero variabile di geni, ognuno dei quali definisce una caratteristica.

Operatori genetici gli operatori genetici specificano in quale modo la ricerca procede nello spazio delle soluzioni nel corso dell'evoluzione; per simulare l'evoluzione biologica naturale, a ogni iterazione alcune funzioni per la manipolazione di stringhe, dette operatori genetici, possono essere applicate ai genotipi della popolazione per generare individui con un nuovo cromosoma. I due operatori genetici usati sono l'operatore di cross-over che permette di combinare i geni di due individui e l'operatore di mutazione che, in maniera casuale, cambia alcuni valori dei geni (per mezzo dell'operatore di mutazione è possibile evitare che il processo di ricerca della soluzione ottima si arresti in un minimo locale).

Funzione di fitness a ogni iterazione, la funzione di fitness restituisce per ogni individuo della popolazione un valore numerico proporzionale alla performance della soluzione codificata nel cromosoma dell'individuo stesso. Nel nostro caso la fitness è determinata misurando la performance di un classificatore logistico addestrato sulle caratteristiche che il cromosoma codifica.

L'obiettivo dell'algoritmo genetico è dunque quello di massimizzare la funzione di fitness, ovvero, generare nel tempo individui il cui cromosoma codifichi soluzioni sempre migliori nella speranza di raggiungere la soluzione ottima.

L'algoritmo base che definisce il funzionamento dell'algoritmo genetico può essere sinteticamente riassunto come segue [22]:

1. Inizializzazione: generazione casuale di una popolazione di partenza costituita da n individui;
2. Valutazione della fitness : per ogni individuo della popolazione viene valutata la sua fitness;
3. Evoluzione: a partire dalla popolazione attuale viene prodotta una nuova generazione di n individui ripetendo m volte le seguenti operazioni:
 - a) Selezione degli individui genitori: dalla popolazione attuale sono selezionati degli individui con tecniche di *tournament* (si rimanda a [11]) che garantiscono il principio della sopravvivenza dei più adatti, ossia quelli con una fitness maggiore;

- b) Applicazione degli operatori genetici in maniera probabilistica;
 - c) Inserimento nella nuova generazione: l'individuo figlio viene inserito nell'insieme degli individui della nuova generazione, insieme ai migliori della generazione precedente (principio dell'elitismo) che alla fine dello step 3 sostituiranno la popolazione attuale;
4. Valutazione delle condizioni di terminazione: se si riscontrano le condizioni per terminare l'algoritmo, si restituisce la migliore soluzione trovata; altrimenti si torna allo step 2.

3.2.3 Classificatore Logistico

Un classificatore logistico è un tipo di classificatore basato sulla regressione logistica, che può essere applicato nei casi in cui la variabile dipendente y sia di tipo dicotomico, come lo sono tutte le variabili che possono assumere esclusivamente due valori (nel nostro caso target/non-target). Nella regressione logistica, la risposta binaria y a un vettore di variabili indipendenti X , è considerata come una variabile casuale binomiale (bernoulliana) di parametro p .

Il modello viene descritto da:

$$\ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \dots + \beta_k x_k = X\beta,$$

da cui:

$$p(X) = \frac{e^{X\beta}}{1+e^{X\beta}},$$

che rappresenta la funzione di ripartizione di una variabile casuale logistica.

Per maggiori informazioni sui classificatori logistici si rimanda a [29] e per l'implementazione del caso in esame si rimanda a [47].

3.2.4 Test

Una volta terminata l'evoluzione dell'algoritmo genetico e finito l'addestramento del classificatore logistico, si passa alla fase di test.

Si è scelto di effettuare il test solo di alcuni individui della popolazione finale dell'algoritmo genetico: in particolare, si sono ordinati gli individui in base al risultato della funzione di fitness e sono stati scelti quelli la cui fitness era almeno il 90% della fitness del miglior individuo.

Per ogni individuo scelto, si effettua il test con le lettere del test-set e il classificatore dello Speller produce una serie di valori numerici che serviranno successivamente per la realizzazione del filtro asincrono.

In particolare, per ogni lettera che si vuole scrivere, l'output consiste in una matrice. Ogni posto della matrice contiene un punteggio (*score*) che è quello che ha ricevuto ogni stimolazione (l'illuminazione di una riga o una colonna) nel corso di ognuna delle 10 ripetizioni.

Attualmente, da questa matrice, sono estratte le lettere che vengono scelte dallo Speller per essere mostrate a video. La scelta di ogni lettera viene fatta nel modo seguente: si sommano gli score ottenuti da ogni stimolazione nel corso delle diverse ripetizioni e si selezionano la riga e la colonna le cui stimolazioni hanno ottenuto lo score più alto; una volta ottenuti la riga e la colonna si va a prendere la lettera che si trova in quella posizione della griglia (Figura 3.4) e si mostra a video.

3.3 Realizzazione del filtro asincrono

Dopo l'acquisizione e la prima elaborazione dei dati, si arriva alla parte più importante del lavoro di tesi: la realizzazione del filtro che permette di trasformare la BCI da sincrona in asincrona.

Il comportamento che noi vogliamo ottenere dal sistema è il seguente: la lettera scelta dallo Speller per essere mostrata a video, deve essere effettivamente mostrata solo se l'utente in quel momento prestava attenzione alla griglia.

La considerazione dalla quale si è partiti è la seguente: se la BCI dà in uscita una lettera anche quando l'utente non è attento, quest'ultima sarà data in maniera casuale e di conseguenza sarà sbagliata con probabilità 97,2% (nel nostro caso, infatti, la possibilità di indovinare la lettera corretta è solo 1 su 36).

Il problema della disattenzione diventa quello di riuscire a capire, dagli score in output dallo Speller, se la lettera è stata scelta perché vi era una forte evidenza che fosse la lettera corretta oppure se la scelta è stata fatta solo perché casualmente lo score era leggermente superiore alle altre.

Questa impostazione del problema permette di risolvere il problema asincrono e introduce anche una miglioria nel sistema Speller P300, ossia un meccanismo di correzione dell'errore (infatti, se gli score delle diverse lettere sono simili tra loro è molto probabile che si stia commettendo un errore).

La scelta progettuale, pertanto, è stata quella di addestrare il filtro a distinguere due classi: la classe delle lettere che vengono bloccate perché errate o date in modo casuale (classe NON-PASSA), e la classe delle lettere che possono, invece, essere mostrate a video (classe PASSA). La classe PASSA è considerata la classe dei positivi e la classe NON-PASSA dei negativi.

Il procedimento utilizzato per la creazione del filtro prevede alcuni passi da eseguire: l'estrazione delle feature, la selezione delle feature, la scelta del classificatore, l'addestramento del classificatore e infine la valutazione delle prestazioni.

3.3.1 Dataset

Come detto nei paragrafi precedenti, i dati utilizzati per la realizzazione del filtro provengono dall'output del test dello Speller. In particolare, quindi, sono degli score che si riferiscono a dati di 8 parole in cui il soggetto era attento e 8 in cui il soggetto non lo era.

Per essere certi che le performance trovate siano quanto più realistiche e veritiere possibili si è utilizzato il metodo dell'holdout, che consiste nel dividere il set di dati in due sottoinsiemi che chiameremo Train-set e Test-set, il primo dei quali è utilizzato per il processo di selezione e addestramento e il secondo per il processo di valutazione.

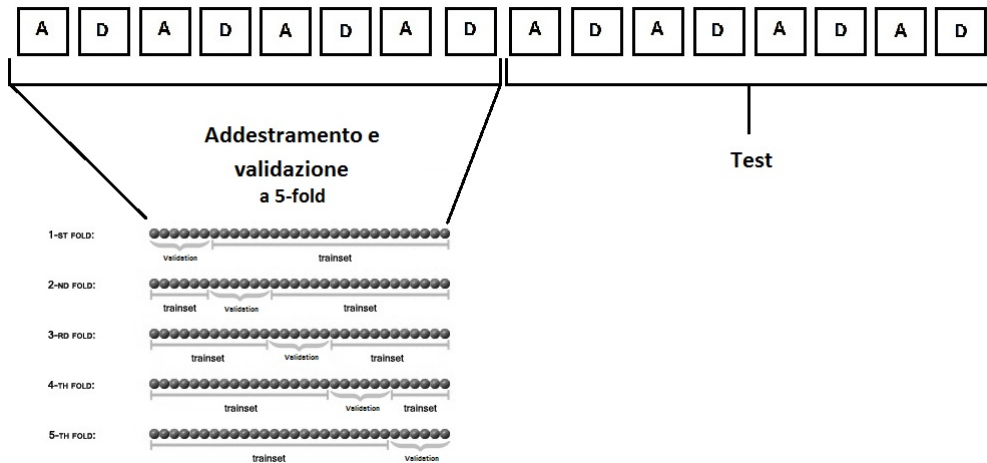


Figura 3.9: Dataset del filtro asincrono

Nel nostro caso si è scelto di dividere a metà i dati, in modo da avere 80 stimolazioni per il train-set, 40 attento e 40 disattento, e 80 stimolazioni per il test-set, anche in questo caso 40 attento e 40 disattento.

Per il processo di selezione delle feature e del miglior classificatore si sono utilizzati come detto i dati del Train-set, ma per evitare problemi di overfitting nella scelta (eccessivo adattamento ai dati), che potessero deteriorare le prestazioni su dati nuovi, si è scelto di usare la tecnica della cross-validazione.

La tecnica di cross-validazione a k -fold (nel nostro caso a 5-fold) consiste nel dividere il set di dati (Train-set) in k sottoinsiemi, $(Train-set)^{(1)}$, $(Train-set)^{(2)}$... $(Train-set)^{(k)}$; quindi, per k volte, a ogni iterazione i , il processo di addestramento è eseguito utilizzando i dati in $(Train-set)^{(1)}$, $(Train-set)^{(2)}$... $(Train-set)^{(i-1)}$, $(Train-set)^{(i+1)}$... $(Train-set)^{(k)}$, mentre il processo di valutazione è eseguito utilizzando i dati in $Train-set^{(i)}$. La performance finale è data dalla media delle performance calcolate a ogni iterazione (vedi Figura 3.9).

Una volta selezionate le feature e scelto il classificatore con la tecnica appena descritta, si utilizza l'intero Train-set per addestrare il classificatore definitivo. Infine si procede alla fase di test con i dati contenuti nel Test-set, che pertanto non erano mai stati utilizzati in nessuna fase precedente.

Shuffle dei dati

Nel corso del lavoro, ci si è resi conto che a un differente ordine di introduzione dei dati, corrisponde un cambiamento significativo delle performance del filtro. Il motivo di questo comportamento anomalo potrebbe risiedere nel fatto che data la durata delle acquisizioni, il comportamento dell'utente o meglio il suo livello di concentrazione tendeva ad abbassarsi verso la fine delle stesse e quindi i dati acquisiti

alla fine hanno performance peggiori. Si è pertanto deciso di mescolare i dati (in termini tecnici, effettuare uno *shuffle*) in maniera casuale in modo tale da eliminare questi trend temporali e permettere che i dati utilizzati sia per il train sia per il test fossero omogenei.

3.3.2 Estrazione delle feature

Una volta ottenuto l'output dall'algoritmo genetico, è necessario estrarre da questo delle caratteristiche utili a identificare l'attenzione prestata dall'utente all'interfaccia BCI. Questo procedimento prende il nome di calcolo delle feature (o *feature extraction*). Con il termine feature si intende il valore simbolico di un attributo o il valore numerico di una variabile che possa essere utilizzato per discriminare un input. Il fine principale del procedimento di calcolo delle feature è quello di ridurre la quantità di informazione acquisita; infatti, sebbene teoricamente sia possibile operare utilizzando l'intero set di informazioni senza nessuna operazione di feature extraction preventiva, questo risulta all'atto pratico molto oneroso, se non addirittura impossibile.

3.3.2.1 Pre-elaborazione dei dati

Innanzitutto si è proceduto a pre-elaborare i dati in due modi diversi, per ogni lettera si è creato:

Istogramma delle Ripetizioni si è costruito un istogramma utilizzando un sistema a punti, nel seguente modo: tutte le stimolazioni relative a una colonna partono da 0 punti e per ogni ripetizione ricevono un +1 se risultano avere lo score più alto delle altre; allo stesso modo si costruisce un istogramma per le stimolazioni relative alle colonne;

Somma degli Score per ciascuna stimolazione, si sono sommati gli score ottenuti nelle diverse ripetizioni, ottenendo così un solo score totale.

A prima vista i due metodi potrebbero apparire simili, ma nella sostanza producono dei risultati molto diversi: nel primo caso si va a sommare un +1 alla stimolazione con lo score maggiore, senza tener conto degli score delle altre stimolazioni; nel secondo caso la risultante è la somma di tutti gli score, quindi entrano in gioco tutte le componenti.

Da un punto di vista teorico, quello che ci aspettiamo di vedere per l'Istogramma delle Ripetizioni è:

- nel caso in cui un input faccia parte della classe PASSA, un punteggio alto corrispondente alla colonna esatta e uno corrispondente alla riga esatta (idealmente 10, come il numero delle ripetizioni), e i restanti valori molto bassi (idealmente 0) (vedi Figura 3.10);

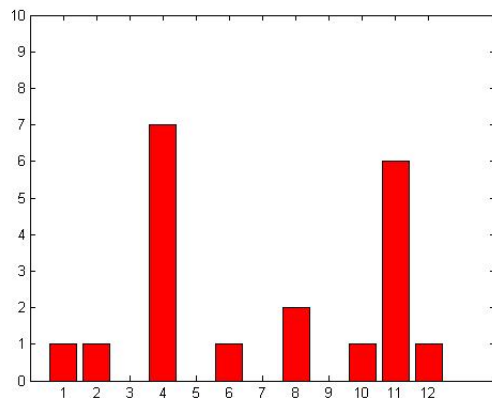


Figura 3.10: Istogramma delle ripetizioni. Esempio di elemento appartenente alla classe PASSA. Si può notare la netta differenza tra i valori alti e quelli bassi.

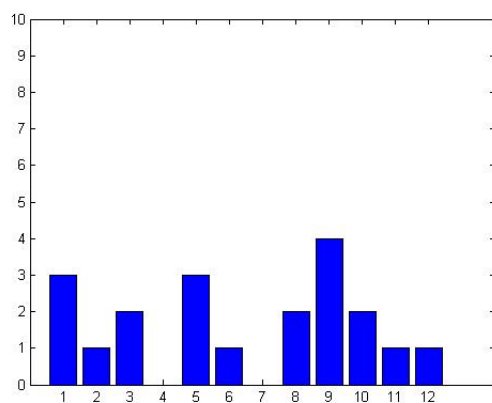


Figura 3.11: Istogramma delle ripetizioni. Esempio di elemento appartenente alla classe NON-PASSA. Si può notare che i valori sono molto simili tra di loro.

- nel caso in cui un input faccia parte della classe NON-PASSA, dei valori simili per tutte le stimolazioni o comunque l'assenza di un valore predominante (vedi Figura 3.11).

Per quanto riguarda la Somma degli Score, ci aspettiamo:

- per la classe PASSA, un valore alto che supera di molto gli altri per una riga e uno per una colonna (vedi Figura 3.12);
- per la classe NON-PASSA dei valori vicini fra di loro (vedi Figura 3.13).

8,94	3,26	2,34	3,65	3,35	3,63		4,70	7,86	4,67	3,71	2,96	3,85
-------------	------	------	------	------	------	--	------	-------------	------	------	------	------

Figura 3.12: Somma dei Punteggi. Esempio di elemento appartenente alla classe PASSA. Ci sono due valori nettamente più grandi degli altri

3,97	2,98	4,95	3,31	4,15	3,51		3,16	2,93	3,40	3,46	4,55	4,06
------	------	------	------	------	------	--	------	------	------	------	------	------

Figura 3.13: Somma dei Punteggi. Esempio di elemento appartenente alla classe NON-PASSA. Non ci sono valori che spiccano più degli altri

Partendo da queste considerazioni teoriche, dopo un attento studio ed una accurata fase di prova, si è giunti alla scelta di 15 feature che meglio riuscivano a separare le due classi.

Vogliamo ora presentare alcuni metodi usati per ottenere le feature.

- Con l'istogramma delle ripetizioni, uno dei metodi utilizzati è quello dell'**entropia**.

Nella teoria dell'informazione l'entropia misura la quantità di incertezza o informazione presente in un segnale aleatorio.

Da un punto di vista formale, l'entropia di una variabile aleatoria X è la media dell'autoinformazione $I(x_i)$ dei possibili valori della variabile stessa (x_1, x_2, \dots, x_n):

$$H(X) = E[I(x_i)] = \sum_{i=1:n} I(x_i) \cdot P(x_i) = \sum_{i=1:n} P(x_i) \cdot \ln \frac{1}{P(x_i)}$$

Da un punto di vista pratico, la variabile aleatoria per noi è un vettore di 12 elementi contenenti dei valori da 0 a 10 come descritto in precedenza; l'entropia sarà alta se i valori sono simili e molto vicini tra loro, mentre sarà bassa se ci sono valori alti insieme a valori bassi.

- Con la Somma dei Punteggi si sono costruite delle feature con dei metodi originali.

Per ogni lettera si sono presi in considerazione solo gli score delle stimolazioni riferite alle righe (o colonne) e sono stati ordinati in ordine decrescente.

Si è calcolata una retta che meglio approssimasse i punti così trovati. Il **coefficiente angolare** e l'**intercetta** della retta sono stati usati come feature (vedi Figura 3.14 e Figura 3.15).

Per quanto riguarda il coefficiente angolare, se il suo valore è molto negativo ci sarà evidenza della classe PASSA, mentre se il suo valore è vicino allo zero ci sarà evidenza della classe NON-PASSA.

Per quanto riguarda l'intercetta della retta, un valore alto apparterrà a un elemento della classe PASSA, un valore basso ad uno della classe NON-PASSA

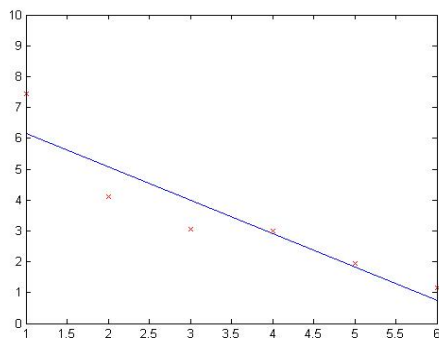


Figura 3.14: Feature retta approssimante. Esempio di elemento appartenente alla classe PASSA.

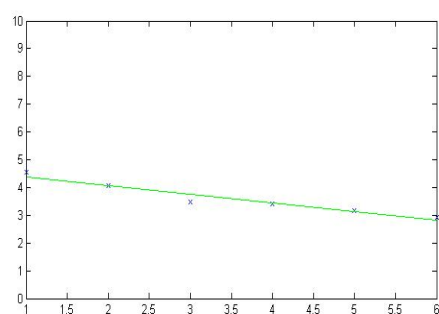


Figura 3.15: Feature retta approssimante. Esempio di elemento appartenente alla classe NON-PASSA.

Le altre feature sono varianti dei metodi esposti, alternando righe e colonne, piuttosto che utilizzando i dati dell'Istogramma delle Ripetizioni o della Somma dei Punteggi

3.3.3 Selezione delle feature

La quantità di dati contenuti nell'insieme di feature generato per mezzo di procedimenti di *feature extraction* è sicuramente inferiore alla quantità di dati contenuti nei segnali acquisiti, ma spesso ancora troppo elevata per poter essere utilizzata in modo efficiente. L'eccessiva quantità di dati può essere causa di diversi problemi tra cui:

Curse of dimensionality Una quantità di dati troppo grande può infatti rapidamente diventare ingestibile e rendere impossibile un uso pratico ed efficiente dei dati. Inoltre è noto che esiste un numero limite di dati o feature superato il quale le performance di un sistema non migliorano, ma addirittura peggiorano [5];

Costo computazionale la quantità di dati disponibili determina i requisiti di spazio e di tempo che un sistema deve soddisfare per gestire e analizzare questi dati. Un numero eccessivo di dati può rapidamente portare all'esaurimento della memoria del sistema o a tempi di elaborazione eccessivamente lunghi.

Per risolvere questi problemi, è opportuno ricorrere a un'ulteriore riduzione della quantità di informazione presente nell'insieme di feature calcolato. Si è quindi proceduto a una selezione delle feature (nota come *feature selection*).

Si noti, però, che una prima selezione delle feature è stata fatta durante le prove effettuate per l'estrazione delle stesse dai dati, pertanto i problemi di feature prive di potere predittivo e quelli di feature ridondanti sono stati già risolti.

Pertanto quello che si cerca di ottenere in questa fase sono in primo luogo dei miglioramenti nella performance della classificazione e in secondo luogo una ottimizzazione del tempo, dello spazio e delle risorse computazionali necessarie per l'elaborazione e l'archiviazione dei dati.

Si è scelto di utilizzare due metodi sequenziali: il sequential forward selection (SFS) e il sequential backward selection (SBS). I metodi sequenziali ricercano un sottoinsieme di feature utili aggiungendo o rimuovendo sequenzialmente feature a partire da un insieme di feature dato. Entrambe le procedure di sequential selection sono relativamente rapide e robuste all'overfitting [24] e offrono un buon trade-off tra il tempo e la qualità dei risultati; non garantiscono, però, il raggiungimento della soluzione ottima [30], infatti, pur partendo da uno stesso insieme iniziale di feature, i due algoritmi giungono a soluzioni differenti.

Sequential Forward Selection (SFS)

L'algoritmo di sequential forward selection parte da un sottoinsieme di feature vuoto e a ogni iterazione aggiunge la feature migliore al sottoinsieme finale di feature. Più rigorosamente l'algoritmo può essere riassunto come segue:

1. $F = \{f_1, f_2, \dots, f_n\}; G = \{\emptyset\};$
2. *select* $f_i \in F \mid \forall j, j \neq i, f_j \in F \text{ Perf}(G \cup f_i) > \text{Perf}(G) \wedge \text{Perf}(G \cup f_i) > \text{Perf}(G \cup f_j);$
3. $F = F \setminus f_i; G = G \cup f_i;$
4. *while* (*exist* f_i) *goto* 2;
5. *end.*

dove F è l'insieme di tutte le possibili feature, G il sottoinsieme finale di feature e $\text{Perf}()$ la funzione che restituisce il valore di performance del classificatore dato un insieme di feature. La funzione $\text{perf}()$ utilizzata tiene conto dell'indice AUC (vedi sottosezione 3.3.6.3).

Sequential Backward Selection (SBS)

All'opposto, l'algoritmo di sequential backward selection parte dall'intero insieme di possibili feature e a ogni iterazione rimuove la feature meno utile dal sottoinsieme finale di feature. Più rigorosamente l'algoritmo può essere riassunto come segue:

1. $G = \{f_1, f_2, \dots, f_n\}$;
2. *select* $f_i \in G \mid \forall j, j \neq i, f_j \in G \text{ Perf}(G \setminus f_i) > \text{Perf}(G) \wedge \text{Perf}(G \setminus f_i) > \text{Perf}(G \setminus f_j)$;
3. $G = G \setminus f_i$;
4. *while* (*exist* f_i) *goto* 2;
5. *end*.

dove G è il sottoinsieme finale di feature e Perf la funzione che restituisce il valore di performance del classificatore dato un insieme di feature. La funzione $\text{perf}()$ utilizzata tiene conto dell'indice AUC (vedi sottosottosezione 3.3.6.3).

3.3.4 Classificazione

Dopo aver generato un insieme di feature e averne opportunamente ridotto la dimensionalità, è possibile utilizzare queste informazioni per dedurre la funzione di classificazione $f(x)$. I dati prodotti possono essere rappresentati per mezzo della matrice delle feature, di dimensione $n \times m$, dove le righe denotano i singoli campioni acquisiti e le colonne le feature; e del vettore delle classi, di dimensione $n \times 1$, dove le righe denotano sempre i singoli campioni acquisiti e la colonna contiene il valore atteso (o classe):

$$\begin{array}{ccc}
 \textit{Feature} & & \textit{Valore atteso} \\
 \\
 \left[\begin{array}{cccc} x_{11} & x_{12} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nm} \end{array} \right] & & \left[\begin{array}{c} y_1 \\ y_2 \\ \vdots \\ y_n \end{array} \right]
 \end{array}$$

Il processo di classificazione consiste nell'uso di questi dati per il calcolo della funzione $f(x)$; la funzione $f(x)$ ha come dominio lo spazio m -dimensionale delle feature e come codominio lo spazio unidimensionale dei valori attesi. La funzione $f(x)$ deve essere in grado di predire quanto meglio possibile le classi; ovvero, dato un vettore $1 \times m$, che rappresenta un campione osservato di cui è ignoto il valore atteso, la funzione $f(x)$ deve restituire la classe del campione sulla base dei valori attesi osservati nei campioni precedenti.

Per il nostro lavoro sono stati scelti diversi tipi di classificatori di cui si sono testate le performance. Vediamo ora brevemente il funzionamento dei classificatori utilizzati.

Classificatore lineare Si consideri il problema della classificazione tra due classi, C1 e C2. Dato un vettore di feature $x = [x_1, x_2 \dots x_n]$, il classificatore lineare equivale a una semplice funzione discriminante del tipo:

$$y(x) = w^T x + w_0,$$

dove y è l'indice della classe, $w = [w_1 w_2 \dots w_n]^T$ è il vettore dei pesi e w_0 è il bias [49]; in altre parole x appartiene a C1 se $y > 0$, mentre appartiene a C2 se $y < 0$; il caso $y = 0$ può essere arbitrariamente assegnato a una delle due classi. L'equazione $y = 0$ corrisponde all'iperpiano H che separa le due classi; è possibile perciò distinguere due semispazi corrispondenti alle due regioni di decisione relative a C1 e a C2; l'orientamento dell'iperpiano è dato dal vettore normale w , mentre la locazione dell'iperpiano è determinata dal bias. Alternativamente, è possibile interpretare il modulo di $y(x)$ come la distanza tra x e l'iperpiano H , e il segno di $y(x)$ come la regione di decisione nella quale si trova x [15].

Classificatore SVM Un classificatore SVM (support vector machine) è un classificatore lineare che permette di classificare i dati di due o più classi che siano linearmente separabili o non-linearmente separabili [49]. Nel caso di dati linearmente separabili, il classificatore SVM cerca di massimizzare la distanza tra l'iperpiano di decisione e i vettori, paralleli al piano di decisione, passanti per i punti più vicini all'iperpiano e appartenenti a classi differenti; tali vettori sono chiamati vettori di supporto, da cui il nome di questo algoritmo di classificazione (vedi Figura 3.16).

Nel caso di dati non linearmente separabili, il classificatore SVM ricerca e calcola l'iperpiano che massimizza il margine e minimizza il limite superiore dell'errore di classificazione; il trade-off tra la massimizzazione del margine e la minimizzazione dell'errore di classificazione è determinato da una costante scelta a priori.

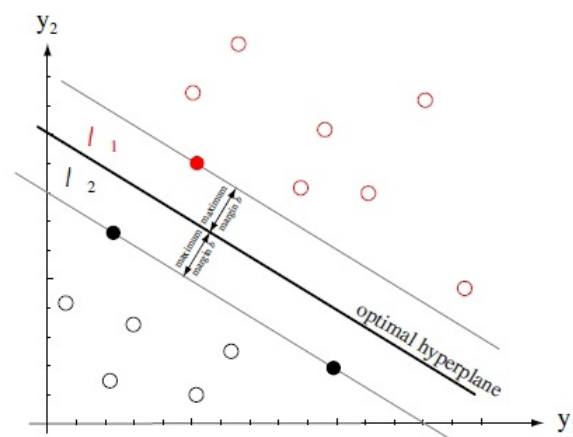


Figura 3.16: Classificatore SVM

Dato un vettore di feature $x = [x_1, x_2 \dots x_n]$, l'obiettivo è quello di trovare una semplice funzione discriminante del tipo:

$$y(x) = w^T x + w_0,$$

dove y è l'indice della classe, $w = [w_1, w_2 \dots w_n]^T$ è il vettore dei pesi e w_0 è il bias o threshold. Si noti che ogni possibile iperpiano è caratterizzato dalla sua direzione, data da w , e dalla sua esatta posizione nello spazio, determinata da w_0 ; una volta trovati i piani che massimizzano il margine per ogni possibile direzione, è necessario scegliere la direzione che garantisca il massimo margine possibile. Per trovare una soluzione è opportuno impostare un problema di ottimizzazione [49]:

$$\min J(w) = \frac{1}{2} \|w\|^2,$$

$$\text{vincolo } y_i(w^T x_i + w_0) \geq 1 \quad i = 1, 2 \dots n,$$

in cui si cerca di massimizzare il margine tra il piano e i vettori di supporto rispettando il vincolo che nessun punto sia contenuto nello spazio tra i vettori di supporto e l'iperpiano di decisione. Questo è un problema di ottimizzazione appartenente alla classe dei problemi di programmazione quadratica, il cui metodo di risoluzione standard è il metodo dei moltiplicatori di Lagrange. E' provato che tale problema ammette un'unica soluzione e che, conseguentemente, l'iperpiano di decisione selezionato dal classificatore SVM è unico [10]. Nel caso di dati non linearmente separabili la condizione di vincolo specificata precedentemente non può essere più rispettata; comunque si scelga l'iperpiano di decisione sarà possibile avere punti contenuti nello spazio delimitato dai vettori di supporto oppure punti contenuti nello spazio di un'altra classe. Per risolvere questo problema è possibile riscrivere il problema di ottimizzazione precedente nel seguente modo:

$$\min J(w, w_0, \xi_i) = \frac{1}{2} \|w\|^2 + C \sum_{i=1:n} \xi_i,$$

$$\text{vincolo } y_i(w^T x_i + w_0) \geq 1 - \xi_i \quad i = 1, 2 \dots n$$

dove è stata introdotta una nuova costante C e delle nuove variabili, ξ_i , dette *slack variable*; il nuovo obiettivo è dunque quello di massimizzare il margine del classificatore SVM e, nel contempo, mantenere il numero di punti con $\xi_i > 0$ il più basso possibile; il trade-off tra ampiezza del margine e numero di punti che non risultano linearmente separabili è determinato dal valore della costante C . Ancora una volta questo problema di ottimizzazione è un problema di programmazione quadratica risolvibile calcolando i moltiplicatori di Lagrange [49, 10]. Si noti, infine, che un classificatore SVM può essere opportunamente esteso per calcolare superfici di decisione non lineari invece di semplici iperpiani [30]; il metodo più utilizzato per determinare

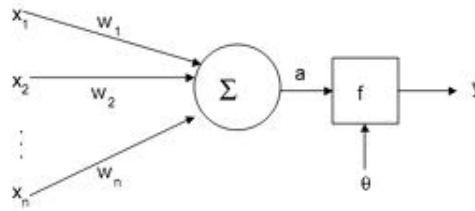


Figura 3.17: Struttura di un neurone artificiale

funzioni di decisione non-lineari è il metodo del kernel; tale metodo permette di proiettare le feature in uno spazio con più dimensioni e utilizzare quindi, in questo nuovo spazio, un classificatore SVM lineare [10].

Il classificatore SVM è implementato tramite un toolbox che permette la selezione automatica dei parametri [7].

Rete neurale Una rete neurale artificiale è un modello matematico che tenta di emulare il funzionamento dei neuroni biologici.

Alla base di una rete neurale c'è il neurone artificiale, la cui struttura è mostrata in Figura 3.17. Vi sono n canali di ingresso a ciascuno dei quali è associato un peso; l'uscita y di ogni neurone è calcolata applicando la funzione di attivazione (nel nostro caso la tangente sigmoideale) alla somma pesata degli ingressi.

La struttura della rete è formata da tre strati: lo strato di input, uno o più strati nascosti e uno strato di output (Figura 3.18).

La topologia della rete utilizzata presenta uno strato di input con un numero di neuroni pari al numero di feature utilizzate, uno strato nascosto di 20 neuroni ed uno strato di output di un solo neurone.

Nel nostro caso i segnali viaggiano dallo strato di ingresso verso lo strato di uscita, si parla pertanto di rete feedforward.

Durante la fase di addestramento l'input dato alla rete è propagato ai livelli successivi e viene calcolato l'errore commesso tramite il Mean Squared Error (errore quadratico medio). A questo punto l'errore fatto dalla rete è propagato all'indietro (Backpropagation) e i pesi sono aggiornati in maniera appropriata seguendo la funzione di apprendimento scelta (Levenberg-Marquardt).

3.3.5 Algoritmo di selezione automatica

E' stato messo a punto un algoritmo di selezione automatica di feature e classificatore.

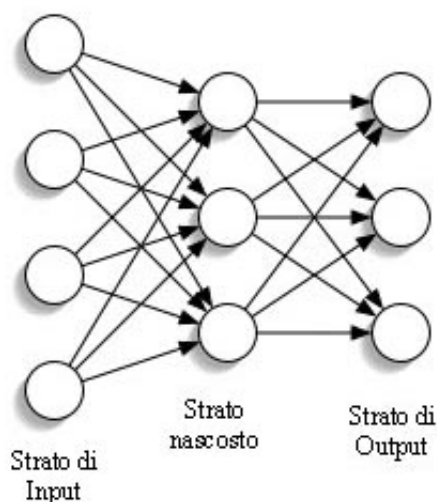


Figura 3.18: Struttura di una rete neurale

L'algoritmo funziona nel modo seguente:

1. estrazione delle feature dai dati di train;
2. selezione delle feature tramite validazione a k -fold (sottosezione 3.3.1) e metodo SFS (sezione 3.3.3) (la funzione $perf()$, in questo caso, è il max tra le $perf()$ di tutti i classificatori);
3. selezione del classificatore tramite validazione a k -fold con il sotto-insieme di feature trovato al punto 5;
4. train finale del classificatore trovato al punto 6, con il sotto-insieme di feature trovato al punto 5;

3.3.6 Valutazione del classificatore

3.3.6.1 Matrice di confusione, accuratezza, sensibilità e specificità

Un classificatore può essere descritto come una funzione che proietta gli elementi di un insieme in certe classi; nel nostro caso sono due che chiameremo convenzionalmente positivi P e negativi N. Gli esiti predetti dal classificatore li indicheremo con p ed n. Sono possibili quattro risultati:

- il classificatore produce il valore p partendo da un dato appartenente alla classe P. Si dice che il classificatore ha prodotto un vero positivo (TP);
- il classificatore produce il valore p partendo da un dato appartenente alla classe N. Si dice che il classificatore ha prodotto un falso positivo (FP);

- il classificatore produce il valore n partendo da un dato appartenente alla classe N . Si dice che il classificatore ha prodotto un vero negativo (TN);
- il classificatore produce il valore n partendo da un dato appartenente alla classe P . Si dice che il classificatore ha prodotto un falso negativo (FN).

I quattro valori identificati (TP, FP, TN, FN) possono essere rappresentati in una tabella a doppia entrata che conta il numero di unità classificate correttamente o meno per ciascuna delle due modalità possibili. Questa matrice è detta matrice di confusione.

		valori predetti	
		p	n
valori reali	P	veri positivi (TP)	falsi negativi (FN)
	N	falsi positivi (FP)	veri negativi (TN)

I numeri sulla diagonale della matrice di confusione rappresentano le unità statistiche correttamente classificate e gli altri sono gli errori. A partire da tale classificazione, si possono ottenere tre importanti indici sintetici della qualità della classificazione: l'accuratezza, la sensibilità e la specificità.

L'accuratezza è definita come

$$Accuratezza = \frac{TP+TN}{TP+FN+FP+TN},$$

ed esprime la proporzione tra le unità classificate correttamente e le unità totali.

La sensibilità è definita come

$$Sensibilità = Se = \frac{TP}{TP+FN},$$

ed esprime la proporzione di Veri Positivi rispetto al numero totale di positivi effettivi.

La specificità è definita come

$$Specificità = Sp = \frac{TN}{FP+TN},$$

ed esprime la proporzione di Veri Negativi rispetto al numero totale di negativi effettivi.

E' facile verificare che i valori di sensibilità e specificità sono fra loro inversamente correlati in rapporto alla scelta del valore di cut-off.

Per valore di cut-off si intende un valore di soglia che permette di discriminare i risultati in positivi e negativi, cioè quel valore assunto dalla variabile misurata nel test al di sopra del quale il soggetto viene dichiarato positivo e al di sotto del quale viene definito negativo.

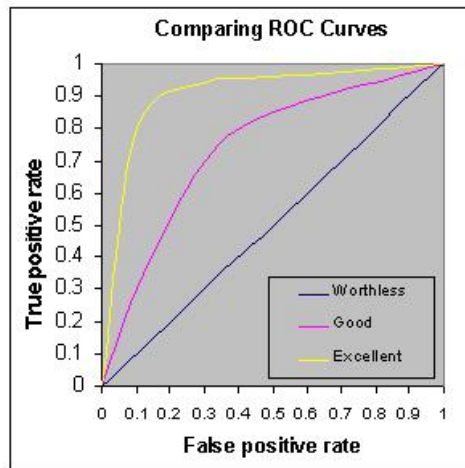


Figura 3.19: Esempi di curve ROC

3.3.6.2 La curva ROC

Il modello di classificazione sarebbe ottimale se massimizzasse contemporaneamente sia la sensibilità che la specificità. Questo tuttavia non è possibile: infatti elevando il valore della specificità, diminuisce il valore di falsi positivi, ma si aumentano i falsi negativi, il che comporta una diminuzione della sensibilità. Si può quindi osservare che esiste un trade-off tra i due indici. La relazione tra i suddetti parametri può essere rappresentata attraverso una linea che si ottiene riportando, in un sistema di assi cartesiani e per ogni possibile valore di cut-off, la proporzione di veri positivi in ordinata e la proporzione di falsi positivi in ascissa. Se il risultato del test è calcolato su scala continua, si possono calcolare i valori di Se e il complemento a uno della specificità, $1-Sp$. L'unione dei punti ottenuti riportando nel piano cartesiano ciascuna coppia di Se e $1-Sp$ genera una spezzata, la curva ROC (vedi Figura 3.19).

E' importante la pendenza locale della curva; ad esempio, una forte pendenza significa un buon incremento di sensibilità con piccola perdita di specificità. Un test perfetto dal punto di vista discriminatorio è rappresentato da una curva ROC che passa per l'angolo superiore sinistro degli assi cartesiani (massima specificità e sensibilità). Al contrario, la curva ROC per un test assolutamente privo di valore informativo è rappresentata dalla bisettrice.

Il valore di cut-off che si è scelto di utilizzare è il punto sulla curva ROC più vicino all'angolo superiore sinistro che rappresenta il miglior compromesso fra sensibilità e specificità.

Infatti, la distanza di ogni punto della curva ROC al punto $(0,1)$ è pari a

$$d = \sqrt{(1 - Se)^2 + (1 - Sp)^2}.$$

Per ottenere il punto di cut-off ottimale si calcola questa distanza per ogni combinazione di Se e Sp : il valore soglia ottimale sarà il punto con distanza minore [39].

3.3.6.3 Area sottesa alla curva ROC

La curva ROC può essere utilizzata per confrontare test diagnostici diversi che fanno riferimento ad uno stesso gruppo di unità statistiche. Uno degli indici più utilizzati per valutare la bontà della regola di classificazione è l'AUC (Area under the ROC curve).

L'AUC è la regione di spazio sottesa alla curva ROC. Il calcolo dell'AUC per una curva empirica (come nel nostro caso), può essere effettuato semplicemente connettendo i diversi punti della curva ROC all'asse delle ascisse con segmenti verticali e sommando le aree dei risultanti poligoni generati nella zona sottostante.

L'AUC è un ottimo metodo per la valutazione delle performance di un classificatore poiché permette di valutarne la prestazione indipendentemente dal punto di cut-off scelto. La validità del metodo è dimostrata in [6].

Durante la fase di selezione delle feature è stato necessario utilizzare una funzione (chiamata in precedenza Perf) che restituisse la performance del classificatore dato un determinato insieme di feature. La stessa funzione è stata utilizzata al momento di scegliere quale tra i classificatori provati avesse le migliori prestazioni.

Per determinare la prestazione di un classificatore questa funzione fa uso dell'indice AUC.

4 Risultati sperimentali

Sono state eseguite diverse prove sul filtro realizzato per determinare quali fossero le scelte progettuali migliori su cui lavorare. Alcune di queste prove (chiamate nel seguito *run*), che corrispondono a diverse combinazioni di test, sono riportate nel corso di questo capitolo in modo da offrire una visione abbastanza completa del lavoro svolto.

Si ricorda che il filtro viene addestrato e testato sull'output dello Speller P300 e in particolare sull'output prodotto dall'elaborazione dei migliori individui della popolazione (o meglio dei relativi cromosomi) selezionati dall'algoritmo genetico (vedi sottosezione 3.2.4).

Per la stesura dei risultati si è tenuta in considerazione la media delle performance su tutti gli individui testati.

Inoltre, come detto nella sezione 3.3.1, è stato introdotto un meccanismo di data shuffle casuale; pertanto per ogni combinazione di test sono stati eseguiti diversi run con differenti valori casuali per lo shuffle. I risultati mostrati nel corso del capitolo fanno riferimento alla media dei run.

Si è scelto di riportare nelle tabelle e nei grafici del presente capitolo tre misure di performance: l'accuratezza, la sensibilità e la specificità. Questa scelta deriva dal fatto che queste misure sono generalmente molto utilizzate e sono anche quelle di più immediata comprensione. L'accuratezza è una percentuale data dal rapporto tra i casi classificati correttamente e i casi totali. La sensibilità esprime la proporzione di Veri Positivi rispetto al numero totale di positivi effettivi. La specificità esprime, invece, la proporzione di Veri Negativi rispetto al numero totale di negativi effettivi. Per la definizione di questi concetti, si rimanda alla sottosezione 3.3.6.1. Ricordiamo brevemente che per Positivi si intende la classe PASSA, ossia la classe delle lettere corrette e stato di attenzione, mentre per negativi si intende la classe NON-PASSA, lettere sbagliate o stato di disattenzione.

Nel corso del capitolo, verranno dapprima introdotte le combinazioni di test utilizzate, poi saranno effettuati alcuni confronti per valutare le prestazioni dei diversi

classificatori e l'impatto delle tecniche di feature selection, nel seguito si mostreranno i risultati dell'uso di un algoritmo di scelta automatica del classificatore e delle feature e, infine, si cercherà di fare un confronto con i pochi studi nello stesso campo presenti in letteratura.

4.1 Combinazioni di test

Con i dati di ciascuno dei quattro soggetti che si sono sottoposti all'acquisizione in laboratorio, sono stati effettuati dei run con diverse combinazioni di test. In particolare sono state provate diverse combinazioni con i diversi classificatori implementati e con diverse tecniche di feature selection.

I valori nella tabelle, come descritto nel paragrafo precedente, sono delle medie e tra parentesi è possibile leggerne la deviazione standard.

Si riportano di seguito le configurazioni utilizzate:

1. **Metodo di feature selection:** nessuno
Classificazione: Classificatore Lineare

Questa configurazione è la più elementare in quanto le feature non vengono selezionate e la classificazione avviene per mezzo di un semplice classificatore lineare.

	SBJ1	SBJ2	SBJ3	SBJ4
Accuratezza	93,93% ($\pm 1,21$)	91,10% ($\pm 1,61$)	83,09% ($\pm 2,08$)	84,97% ($\pm 2,31$)
Sensibilità	93,48% ($\pm 2,77$)	88,22% ($\pm 4,50$)	77,67% ($\pm 5,67$)	79,66% ($\pm 5,14$)
Specificità	94,33% ($\pm 1,94$)	93,44% ($\pm 2,05$)	87,00% ($\pm 2,86$)	89,73% ($\pm 3,34$)

2. **Metodo di feature selection:** Sequential Forward Selection
Classificazione: Classificatore Lineare

Questa configurazione prevede la selezione delle feature per mezzo dell'algoritmo SFS e la classificazione avviene per mezzo di un classificatore lineare.

	SBJ1	SBJ2	SBJ3	SBJ4
Accuratezza	94,38% ($\pm 0,90$)	90,02% ($\pm 1,87$)	83,14% ($\pm 1,85$)	84,82% ($\pm 1,85$)
Sensibilità	94,40% ($\pm 2,57$)	86,35% ($\pm 4,29$)	78,98% ($\pm 4,48$)	78,65% ($\pm 5,22$)
Specificità	94,40% ($\pm 1,82$)	92,96% ($\pm 2,05$)	86,07% ($\pm 2,50$)	90,26% ($\pm 2,89$)

3. **Metodo di feature selection:** Sequential Backward Selection
Classificazione: Classificatore lineare

Questa configurazione prevede la selezione delle feature per mezzo dell'algoritmo SBS e la classificazione avviene per mezzo di un classificatore lineare.

	SBJ1	SBJ2	SBJ3	SBJ4
Accuratezza	94,53% ($\pm 0,94$)	90,34% ($\pm 1,91$)	83,22% ($\pm 1,73$)	85,12% ($\pm 2,11$)
Sensibilità	94,80% ($\pm 2,28$)	87,03% ($\pm 4,31$)	78,76% ($\pm 4,91$)	79,29% ($\pm 5,35$)
Specificità	94,35% ($\pm 1,80$)	92,98% ($\pm 1,72$)	86,43% ($\pm 2,47$)	90,35% ($\pm 2,73$)

4. **Metodo di feature selection:** nessuno
Classificazione: Support Vector Machine

In questa configurazione le feature non vengono selezionate e la classificazione avviene per mezzo di un classificatore SVM.

	SBJ1	SBJ2	SBJ3	SBJ4
Accuratezza	96,41% ($\pm 0,83$)	92,61% ($\pm 1,70$)	86,81% ($\pm 1,59$)	89,45% ($\pm 1,56$)
Sensibilità	97,34% ($\pm 1,18$)	92,22% ($\pm 2,17$)	88,36% ($\pm 3,15$)	88,50% ($\pm 2,94$)
Specificità	95,68% ($\pm 0,97$)	92,86% ($\pm 1,98$)	80,79% ($\pm 1,89$)	90,26% ($\pm 1,53$)

5. **Metodo di feature selection:** Sequential Forward Selection
Classificazione: Support Vector Machine

Questa configurazione prevede la selezione delle feature per mezzo dell'algoritmo SFS e la classificazione avviene per mezzo di un classificatore SVM.

	SBJ1	SBJ2	SBJ3	SBJ4
Accuratezza	96,15% ($\pm 0,93$)	91,71% ($\pm 1,88$)	86,86% ($\pm 1,38$)	88,64% ($\pm 1,63$)
Sensibilità	97,32% ($\pm 0,99$)	91,53% ($\pm 2,53$)	88,34% ($\pm 2,46$)	87,39% ($\pm 3,08$)
Specificità	95,25% ($\pm 1,19$)	91,89% ($\pm 1,87$)	85,96% ($\pm 1,50$)	89,68% ($\pm 1,38$)

6. **Metodo di feature selection:** Sequential Backward Selection
Classificazione: Support Vector Machine

Questa configurazione prevede la selezione delle feature per mezzo dell'algoritmo SBS e la classificazione avviene per mezzo di un classificatore SVM.

	SBJ1	SBJ2	SBJ3	SBJ4
Accuratezza	96,20% ($\pm 0,85$)	91,69% ($\pm 1,95$)	86,64% ($\pm 1,42$)	88,64% ($\pm 1,38$)
Sensibilità	97,10% ($\pm 1,16$)	91,54% ($\pm 2,39$)	88,89% ($\pm 2,30$)	87,70% ($\pm 2,56$)
Specificità	95,50% ($\pm 1,07$)	91,79% ($\pm 2,04$)	85,22% ($\pm 1,91$)	89,45% ($\pm 0,90$)

7. **Metodo di feature selection:** nessuno**Classificazione:** Rete Neurale

In questa configurazione le feature non vengono selezionate e la classificazione avviene per mezzo di una rete neurale.

	SBJ1	SBJ2	SBJ3	SBJ4
Accuratezza	94,72% ($\pm 0,94$)	87,51% ($\pm 2,15$)	80,21% ($\pm 1,68$)	83,33% ($\pm 2,86$)
Sensibilità	95,27% ($\pm 1,57$)	86,29% ($\pm 2,27$)	79,76% ($\pm 2,25$)	81,79% ($\pm 3,92$)
Specificità	94,30% ($\pm 0,96$)	88,56% ($\pm 2,57$)	80,50% ($\pm 1,46$)	84,65% ($\pm 2,58$)

8. **Metodo di feature selection:** Sequential Forward Selection**Classificazione:** Rete Neurale

Questa configurazione prevede la selezione delle feature per mezzo dell'algoritmo SFS e la classificazione avviene per mezzo di una rete neurale.

	SBJ1	SBJ2	SBJ3	SBJ4
Accuratezza	95,25% ($\pm 0,85$)	87,24% ($\pm 1,46$)	80,74% ($\pm 1,95$)	85,78% ($\pm 2,66$)
Sensibilità	96,49% ($\pm 1,26$)	85,29% ($\pm 2,67$)	78,14% ($\pm 1,72$)	84,53% ($\pm 2,84$)
Specificità	94,30% ($\pm 1,46$)	88,87% ($\pm 1,77$)	82,53% ($\pm 2,37$)	86,94% ($\pm 2,95$)

9. **Metodo di feature selection:** Sequential Backward Selection**Classificazione:** Rete Neurale

Questa configurazione prevede la selezione delle feature per mezzo dell'algoritmo SBS e la classificazione avviene per mezzo di una rete neurale.

	SBJ1	SBJ2	SBJ3	SBJ4
Accuratezza	94,35% ($\pm 0,52$)	88,05% ($\pm 0,93$)	79,46% ($\pm 2,49$)	83,56% ($\pm 3,40$)
Sensibilità	94,87% ($\pm 1,19$)	87,37% ($\pm 1,47$)	78,33% ($\pm 2,33$)	82,30% ($\pm 4,63$)
Specificità	93,91% ($\pm 0,60$)	88,67% ($\pm 1,39$)	80,31% ($\pm 2,94$)	84,67% ($\pm 3,04$)

4.2 Confronto classificatori

Una volta effettuati tutti i run, la prima cosa che si è pensato di effettuare è un confronto tra i diversi classificatori testati.

A tal fine si sono creati tre grafici, uno per ogni metodo di feature selection, che riportano la prestazione dei tre tipi di classificatore, calcolata sulla media dei 4 soggetti (Figura 4.1).

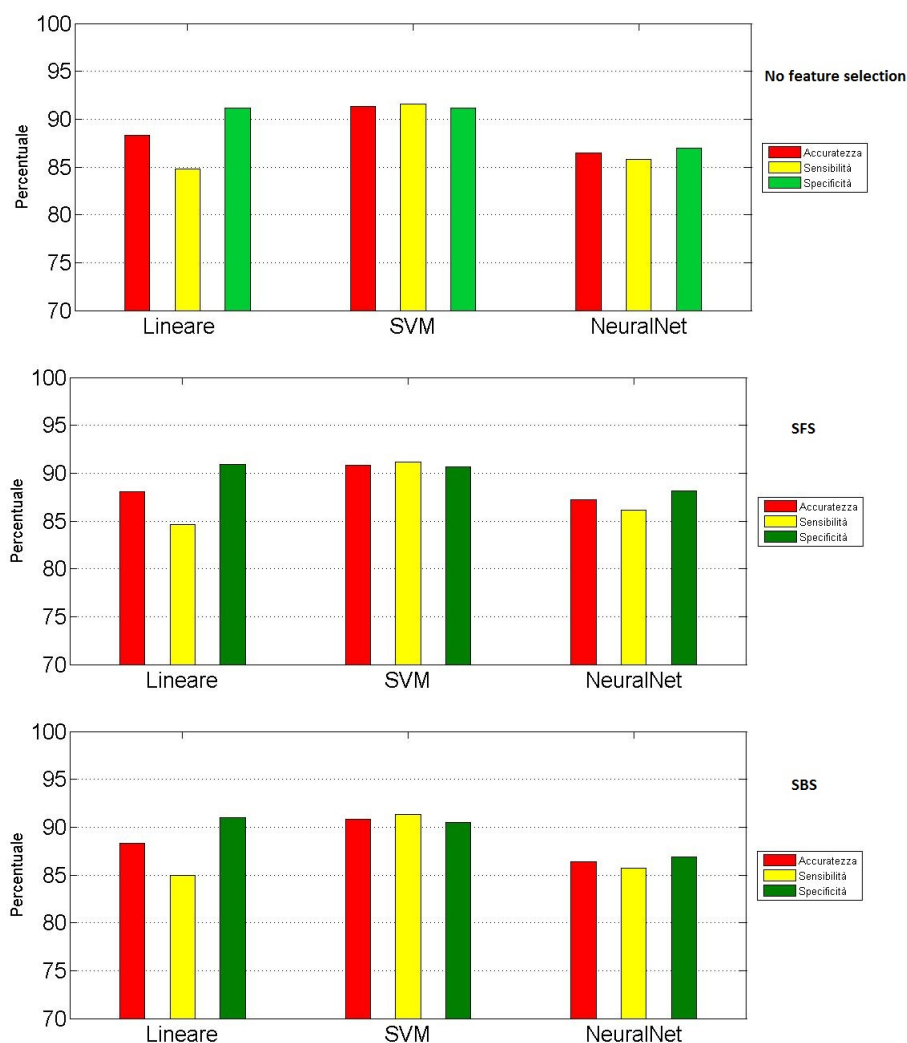


Figura 4.1: Confronto classificatori. Nessun metodo di feature selection in alto, Sequential Forward Selection al centro, Sequential Backward Selection in basso

Come si può notare dall'immagine i tre grafici sono molto simili, pertanto è possibile fare per tutti le medesime considerazioni. Il classificatore SVM risulta avere delle prestazioni migliori rispetto al classificatore lineare e alla rete neurale: raggiunge delle performance più elevate e, inoltre, dimostra di essere il più bilanciato dato che ha una differenza minima tra gli indici di accuratezza, sensibilità e specificità.

Al secondo posto, per accuratezza, troviamo il classificatore lineare. Il suo difetto, però, è quello di classificare troppi elementi come negativi: questo fa crescere il valore della specificità, ma abbassa di tanto il valore della sensibilità del test. Questo comportamento, dovuto probabilmente alla estrema semplicità del classificatore, porterebbe il filtro a rigettare molte lettere con conseguente stress da parte dell'utente del sistema.

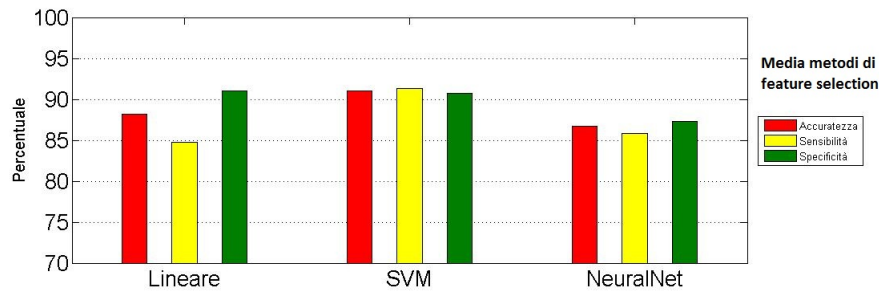


Figura 4.2: Confronto classificatori. Media dei tre metodi di feature selection

La rete neurale ha prestazioni medie inferiori agli altri tipi di classificatore, anche se anch'esso appare un classificatore bilanciato.

Per dare dei valori numerici, che possano rendere meglio l'idea dei concetti espressi, possiamo prendere in considerazione la media dei tre grafici precedenti, mostrata in Figura 4.2.

Il classificatore SVM raggiunge un'accuratezza media del 90,98% con una sensibilità del 91,35% e una specificità del 90,78%.

Il classificatore lineare invece registra una accuratezza dell'88,22%, però, a fronte di una specificità del 91,02%, ha una sensibilità molto più bassa, 84,77%.

I valori della rete neurale sono rispettivamente 86,68%, 85,87% e 87,35%.

Si può concludere che il classificatore migliore è risultato la Support Vector Machine; peraltro oltre a essere il migliore sulla media dei quattro soggetti, si può notare dalle tabelle della sezione 4.1 che risulta essere il migliore per ciascuno dei soggetti.

4.3 Confronto metodi feature selection

Analizzando i dati da una diversa prospettiva, si è potuto confrontare anche i diversi metodi di feature selection: per la precisione si sono confrontate le prestazioni ottenute dai diversi classificatori con il Sequential Forward Selection, il Sequential Backward Selection e l'assenza della procedura di feature selection (Figura 4.3). Anche in questo caso i valori si riferiscono alla media dei 4 soggetti.

Utilizzando il classificatore lineare, si nota che l'impatto di un algoritmo di feature selection è basso: le performance sono lievemente calate con l'algoritmo di SFS, mentre sono migliorate seppur in maniera poco significativa con l'utilizzo del SBS.

Considerando il caso dell'utilizzo di una Support Vector Machine, si nota subito che le prestazioni migliori si ottengono senza nessun algoritmo di feature selection. Infatti sia utilizzando il SFS sia utilizzando il SBS, le performance medie tendono a peggiorare di mezzo punto percentuale.

4.3 Confronto metodi feature selection

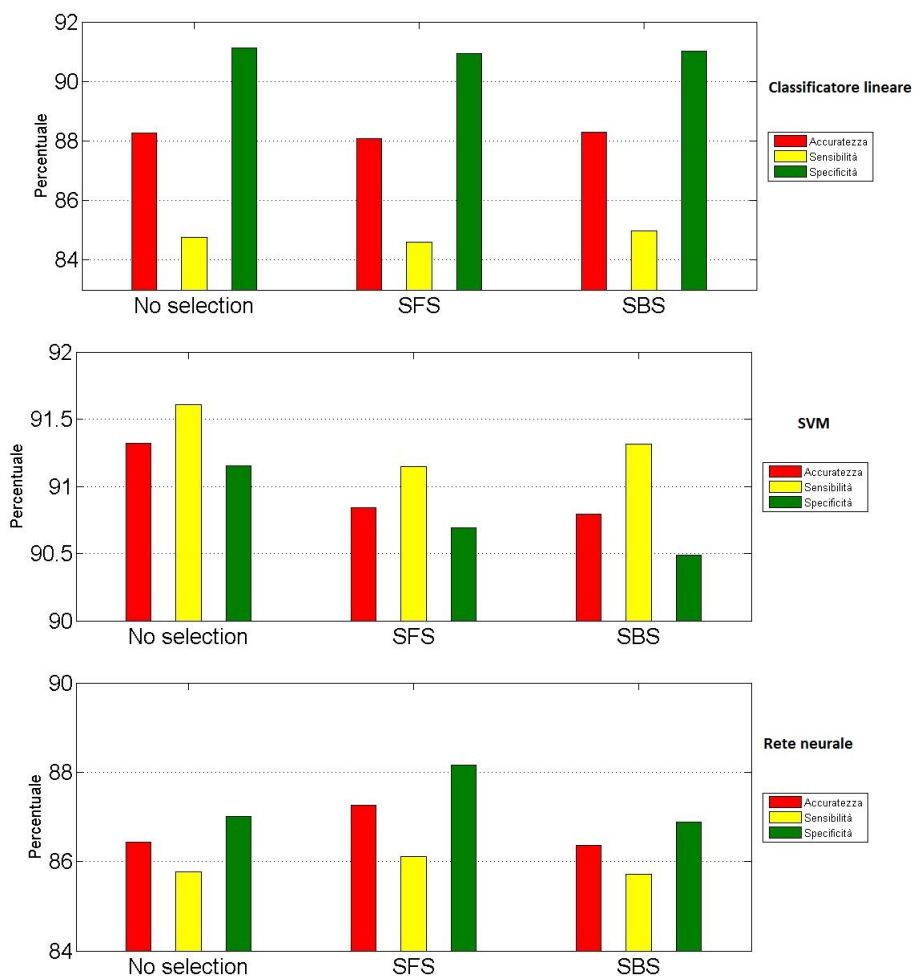


Figura 4.3: Confronto metodi di feature selection. Classificatore lineare in alto, SVM al centro, Rete Neurale in basso

La rete neurale, invece, risponde positivamente all'utilizzo dell'algoritmo di Sequential Forward Selection, con un miglioramento delle prestazioni che si aggira attorno all'1%, mentre resta indifferente all'algoritmo SBS.

Dalla lettura della media dei classificatori (Figura 4.4), possiamo avere una visione d'insieme dell'impatto degli algoritmi di feature selection.

L'utilizzo della Sequential Forward Selection permette un incremento medio dello 0,1% della accuratezza, un abbassamento della sensibilità dello 0,1% e un aumento della specificità dello 0,2%. Il Sequential Backward Selection peggiora, seppur di poco le prestazioni.

Il risultato del testing degli algoritmi di feature selection dimostra la bontà delle scelte effettuate in fase di feature extraction: non ci sono feature con scarso potere predittivo o feature ridondanti.

Dall'altra parte, ci si chiede se è davvero utile avvelersi dell'utilizzo di questi algoritmi-

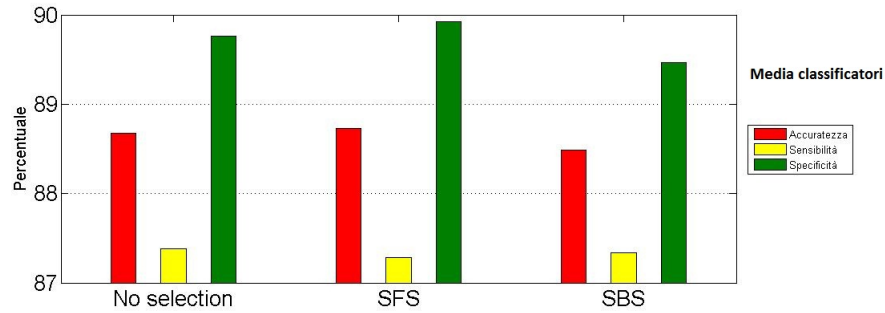


Figura 4.4: Confronto metodi di feature selection. Media dei tre classificatori

mi. La risposta non è immediata: bisogna valutare il trade-off tra il lieve incremento di prestazioni, unito al risparmio di memoria dovuto al minor numero di feature, e il tempo necessario per effettuare l'addestramento del sistema, che nel caso di utilizzo degli algoritmi di feature extraction aumenta notevolmente.

4.4 Confronto soggetti

Si è effettuato quindi un confronto per ognuno dei diversi soggetti. Per far ciò, si sono presi in considerazione tutti i run effettuati da ciascun soggetto e si è proceduto ad estrarre la media delle prestazioni per ognuno di loro. Il risultato è mostrato in Figura 4.5.

E' evidente la differenza di performance tra i vari soggetti:

- il soggetto 1 risulta il migliore del test con performance comprese tra 94,6% e 95,6%;
- il soggetto 2 ha prestazioni che si aggirano tra l'88,5% e il 91,3%;

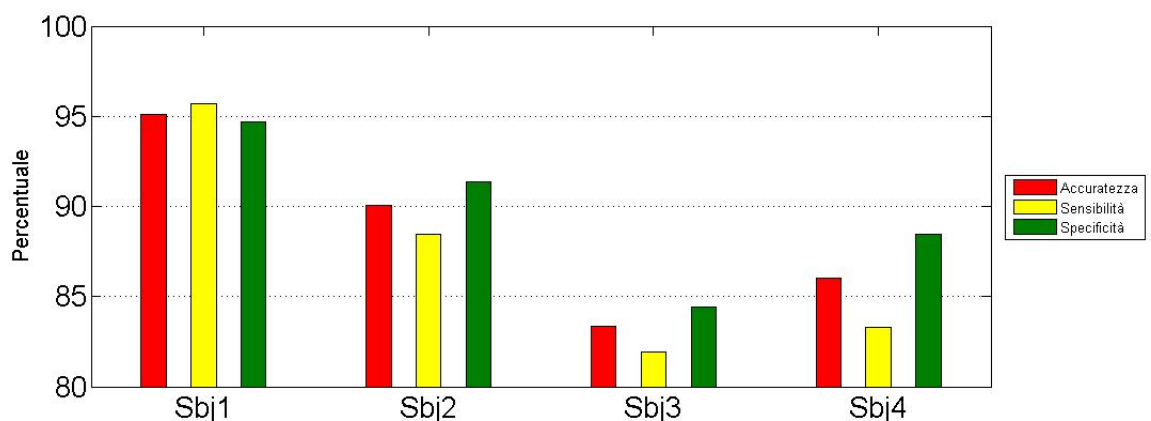


Figura 4.5: Confronto tra i soggetti

- il soggetto 3, con le performance più basse, ha una accuratezza dell'83% con una sensibilità dell'81,9% ed una specificità dell'84,4%;
- il soggetto 4 registra performance intorno all'86% (sensibilità 83,3%, specificità 88,4%).

La causa di questo comportamento, così differente tra i vari soggetti, probabilmente risiede nella differente capacità di concentrazione nel corso delle acquisizioni in laboratorio. Va detto, infatti, che entrambi i soggetti, 3 e 4, hanno riferito di aver avuto difficoltà a mantenere alta l'attenzione per tutta la durata degli esperimenti.

4.5 Selezione automatica di feature e classificatore

Si è messo a punto un algoritmo che permette la selezione automatica delle feature e del classificatore più adatti per un determinato set di dati (sottosezione 3.3.5). Si noti che l'algoritmo può selezionare classificatori diversi per ogni cromosoma testato e quindi la performance finale, che, si ricorda, è data dalla media di tutti i cromosomi, può essere una soluzione ibrida che sfrutta le potenzialità di tutti i classificatori.

Nel seguito si verifica il risultato prodotto da questo algoritmo confrontandolo con i risultati ottenuti in tutti i run precedenti; in particolare si è comparato il risultato con la media dei run precedenti e col il run che aveva ottenuto la performance massima.

L'esito di tale confronto è visibile in Figura 4.6.

Per il primo soggetto, le performance dell'algoritmo di scelta automatica sono ottime: migliorano quasi del 2% la performance della media dei run e ottengono un risultato maggiore, di mezzo punto percentuale, anche del massimo dei run. Per gli altri 3 soggetti l'algoritmo riesce a essere costantemente migliore della media dei run anche se non raggiunge la performance del run migliore.

4.6 Confronto con altri sistemi presenti in letteratura

Cercheremo ora di fare un confronto con i due sistemi presenti in letteratura e introdotti nella sottosezione 2.5.3.

Innanzitutto partiamo dalle prestazioni complessive del nostro sistema: considerando la media su tutti i soggetti dell'algoritmo di scelta automatica esposto nel paragrafo precedente, abbiamo una accuratezza del 90,77%, una sensibilità del 90,32% ed una specificità del 91,17%.

Introduciamo, poi, una nuova misura di prestazione molto presente in letteratura: l'*information transfer rate* (ITR), che indica il numero di bit che un utente è in grado di comunicare tramite l'interfaccia [53]. L'*information transfer rate* è definito nel modo seguente:

$$ITR = n_e \left\{ \log_2 N_s + P \log_2 P + (1 - P) \log_2 \left[\frac{(1-P)}{(N_s-1)} \right] \right\},$$

dove n_e è il numero di lettere elaborate in un minuto, N_s è il numero di simboli sulla griglia, ossia il numero di possibili selezioni equiprobabili, e P è l'accuratezza del sistema.

Nel nostro caso n_e è 2 eventi/min ($250ms \times 12 \times 10$) (sottosezione 3.1.2), N_s è 36 simboli e l'accuratezza è 90,77%, pertanto l'ITR del nostro sistema è 8,5 bit/min.

Il tasso di falsi positivi (FPR) può essere calcolato come $n_e(1 - Sp)$, dove Sp è la specificità; nel nostro caso, quindi, $FPR = 0,18$ eventi/min.

Il sistema sviluppato da Zhang, Guan e Wang [54], riporta un ITR di 20 bit/min con un tasso di falsi positivi di 1 eventi/min.

Va osservato come non sia possibile fare un confronto diretto con queste prestazioni, in quanto si sono seguiti due approcci al problema molto diversi. Nelle nostre elaborazioni abbiamo cercato di ottenere un FPR molto basso e in generale una accuratezza elevata della discriminazione tra classi (da qui la scelta di far effettuare 10 ripetizioni per ogni lettera), a discapito, però, della velocità di inserimento delle lettere. Mentre il sistema con cui ci stiamo confrontando è più veloce (20 bit/min contro il nostro 8,5 bit/min) ma ammette un tasso di errore molto più elevato (1 eventi/min contro 0,18 eventi/min). Inoltre la formula utilizzata per il calcolo dell'ITR nella pubblicazione in questione è leggermente diversa da quella originaria e precedentemente introdotta; nel caso utilizzassimo questa formulazione l'ITR del nostro approccio sarebbe 7,13 bit/min.

Per poter fare un confronto migliore, bisognerebbe, però, effettuare delle nuove acquisizioni con un numero di ripetizioni inferiore in modo da aumentare il valore di n_e ed avere così un ITR confrontabile.

Per quanto riguarda il lavoro di Panicker, Puthusserypady e Ying Sun [38], le considerazioni da effettuare sono analoghe in quanto viene riportata un'accuratezza della discriminazione tra le classi di circa l'88%, con un ITR di 19.05 bits/min, e non viene fornita indicazione del FPR.

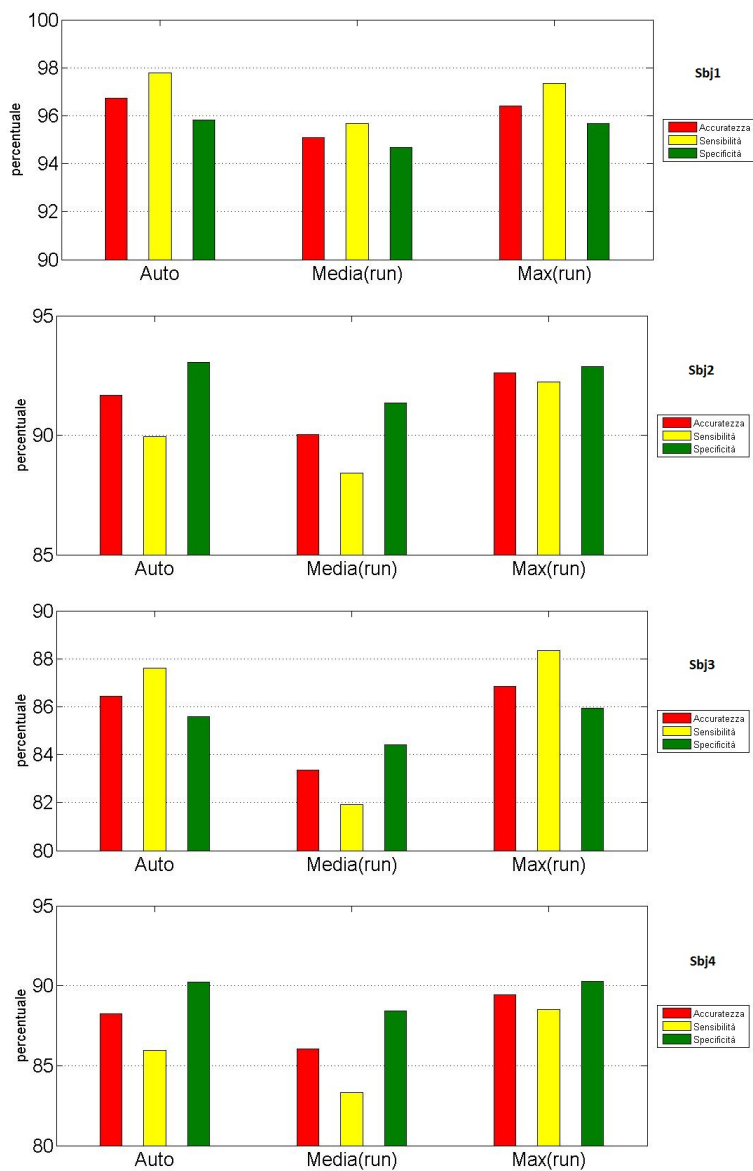


Figura 4.6: Confronto tra algoritmo automatico e run singoli, un istogramma per ogni soggetto

5 Conclusioni e sviluppi futuri

Nel corso del lavoro di tesi si è investigato sulla possibilità di effettuare un controllo asincrono di BCI basate su potenziali evento correlati. Come caso di studio si è preso lo Speller P300 sviluppato al Politecnico di Milano.

Sono stati raccolti in laboratorio dati da quattro soggetti sani in atteggiamento di attenzione e in atteggiamento di disattenzione e si sono elaborati questi dati al fine di creare un filtro che possa essere usato per rendere asincrono lo Speller P300.

La considerazione dalla quale si è partiti è che lo Speller P300 dà in uscita una lettera anche quando l'utente non è attento, pertanto quest'ultima sarà data in maniera casuale.

Pertanto si è cercato di trovare un metodo, che, partendo da alcuni valori calcolati dallo Speller, potesse capire se questa lettera era stata scelta perché vi era una forte evidenza che fosse la lettera corretta oppure se la scelta era stata fatta solo casualmente.

Si è addestrato il filtro a distinguere due classi: la classe delle lettere che vengono bloccate perché errate o date in modo casuale (classe NON-PASSA), e la classe delle lettere che possono, invece, essere mostrate a video (classe PASSA).

Il procedimento utilizzato per la creazione del filtro prevede alcuni passi da eseguire: l'estrazione delle feature, la selezione delle feature, la scelta del classificatore, e l'addestramento del classificatore.

Al fine di determinare quali fossero le scelte progettuali migliori su cui lavorare, sono state eseguite diverse prove sul filtro asincrono realizzato.

Da queste prove è emerso che vi è una forte differenza tra le performance di soggetti diversi, causata probabilmente dalla differente capacità di tenere alta la concentrazione per tutta la durata delle acquisizioni.

Dalla comparazione dei metodi di feature selection, il metodo del Sequential Forward Selection è risultato migliore del Sequential Backward Selection. Però, le performance ottenute con la SFS sono solo minimamente superiori alle prestazioni ottenute senza alcun algoritmo di feature selection; questo, da una parte, dimostra la bontà

delle scelte effettuate in fase di feature extraction, dall'altra, evidenzia che l'utilizzo di questi algoritmi, in questa specifica applicazione, non è indispensabile.

Tra i classificatori, quello che è risultato migliore degli altri, sia in termini performance più elevate sia in termini di bilanciamento tra gli indici di accuratezza, sensibilità e specificità, è stata la Support Vector Machine.

L'utilizzo di un algoritmo di scelta automatica del classificatore, ha però dimostrato che, in alcuni casi, un utilizzo di un sistema ibrido di più classificatori può portare ad ottenere performance superiori.

Visti i risultati mediamente buoni, si conclude che l'utilizzo asincrono della BCI basata su P300 è possibile e la via intrapresa porta a delle performance incoraggianti.

Gli sviluppi futuri, in questo contesto, sono molteplici in quanto solo di recente la ricerca ha cominciato a interessarsi al problema delle BCI self-paced basate su ERP; pertanto, ci limitiamo a proporre alcuni sviluppi che potrebbero essere apportati al sistema da noi elaborato.

Da un punto di vista del miglioramento delle performance del filtro, si rendono necessarie nuove acquisizioni di dati per validare ulteriormente le conclusioni raggiunte e permettere una migliore classificazione grazie alla possibilità di sfruttare un train-set più ampio.

Nell'immediato sarebbe auspicabile l'integrazione del filtro nel sistema Speller P300, con conseguente possibilità di effettuare delle prove in tempo reale sull'applicazione.

D'altra parte, va detto che il campo degli algoritmi di elaborazione del segnale è talmente vasto che gli algoritmi considerati in questa tesi sono solo un piccolo sottoinsieme di tutte le possibili tecniche sviluppate; pertanto un'interessante futura direzione di ricerca è costituita dallo sviluppo e dalla sperimentazione di nuovi e diversi algoritmi in grado di garantire performance migliori.

Si potrebbe poi pensare di integrare il filtro creato con altre tecnologie, come quelle legate agli SSVEP, per avere ulteriori informazioni da poter elaborare e di conseguenza arrivare a delle prestazioni ancora più elevate.

Ringraziamenti

Bibliografia

- [1] AA.VV. The clinical neurophysiology primer. 2007.
- [2] J.D. Bayliss and D.H. Ballard. A virtual reality testbed for brain-computer interface research. *Rehabilitation Engineering, IEEE Transactions on*, 8(2):188–190, jun 2000.
- [3] N. Birbaumer, A. Kubler, N. Ghanayim, T. Hinterberger, J. Perelmueter, J. Kaiser, I. Iversen, B. Kotchoubey, N. Neumann, and H. Flor. The thought translation device (ttd) for completely paralyzed patients. *Rehabilitation Engineering, IEEE Transactions on*, 8(2):190–193, jun 2000.
- [4] Niels Birbaumer. Breaking the silence: Brain-computer interfaces (bci) for communication and motor control. *Psychophysiology*, 43(6):517–532, 2006.
- [5] Christopher M. Bishop and Nasser M. Nasrabadi. Pattern recognition and machine learning. *J. Electronic Imaging*, 16(4), 2007.
- [6] A. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms, 1997.
- [7] C. C. Chang and C. J. Lin. *LIBSVM: a library for support vector machines*. Online, 2001.
- [8] F. Cincotti, D. Mattia, F. Aloise, S. Bufalari, M.G. Marciani, G. Schalk, G. Oriolo, A. Cherubini, F. Davide, and F. Babiloni. Non-invasive brain-computer interface system to operate assistive devices. In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, pages 2532–2535, aug. 2007.
- [9] S. Costagliola, B. Dal Seno, and M. Matteucci. Recognition and classification of p300s in eeg signals by means of feature extraction using wavelet decomposition. *Neural Networks, IEEE - INNS - ENNS International Joint Conference on*, pages 597–603, 2009.
- [10] Joaquim P. Marques de Sa. *Pattern Recognition*. Springer, 2001.
- [11] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Evolutionary Computation, IEEE Transactions on*, 6(2):182–197, apr 2002.
- [12] E. Donchin, K.M. Spencer, and R. Wijesinghe. The mental prosthesis: assessing the speed of a p300-based brain-computer interface. *Rehabilitation Engineering, IEEE Transactions on*, 8(2):174–179, jun 2000.

-
- [13] Emanuel Donchin. Surprise!... surprise? *Psychophysiology*, 18(5):493–513, 1981.
- [14] John P. Donoghue. Connecting cortex to machines: Recent advances in brain interfaces. *Nat Neurosci*, 2002.
- [15] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2 edition, 10 2000.
- [16] Erich E. and Sutter. The brain response interface: communication through visually-induced electrical brain responses. *Journal of Microcomputer Applications*, 15(1):31 – 45, 1992.
- [17] Chris Eliasmith and Charles H. Anderson. *Neural engineering*. 2003.
- [18] Beverina F. et al. User adaptive bcis: Ssvep and p300 based interfaces. *Psychology*, 2003.
- [19] L.A. Farwell and E. Donchin. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and Clinical Neurophysiology*, 70(6):510 – 523, 1988.
- [20] The Society for Neuroscience. *Brain Facts: A Primer on the Brain and Nervous System*. The Society For Neuroscience, fourth edition edition, 2002.
- [21] James Perkins John T. Hansen Frank H. Netter, John A. Craig and Bruce M. Koepfen. *Atlas of neuroanatomy and neurophysiology*. 2002.
- [22] David E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional, 1 edition, January 1989.
- [23] B. Greenstein and A. Greenstein. *Color Atlas of Neuroscience*. Thieme, 2000.
- [24] Isabelle Guyon, Steve Gunn, Masoud Nikravesh, and Lofti Zadeh. *Feature Extraction: Foundations and Applications*. Springer Verlag, 2006.
- [25] CHAPIN J. K. Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex. *Nature neuroscience*, 2(7):664–670, 1999.
- [26] J Kamiya. Biofeedback and self-control: An aldine reader on the regulation of bodily processes and consciousness. *Chicago: Aldine*, 1971.
- [27] Emanuele Lauricella. *Dizionario Medico*. USES Edizioni Scientifiche, Firenze, 1976.
- [28] Mikhail A. Lebedev and Miguel A.L. Nicolelis. Brain-machine interfaces: past, present and future. *Trends in Neurosciences*, 29(9):536 – 546, 2006.
- [29] Yee Leung. *Knowledge discovery in spatial data*. 2010.
- [30] Huan Liu and Hiroshi Motoda, editors. *Computational Methods of Feature Selection*. Chapman and Hall/CRC, 10 2007.
- [31] Werner Lutzenberger, Larry E. Roberts, and Niels Birbaumer. Memory performance and area-specific self-regulation of slow cortical potentials: Dual-task interference. *International Journal of Psychophysiology*, 15(3):217 – 226, 1993.

- [32] CARMENA J. M. Learning to control a brain-machine interface for reaching and grasping by primates. *PLoS BIOLOGY*, 1(2):1–16, 2003.
- [33] S.G. Mason and G.E. Birch. A brain-controlled switch for asynchronous control applications. *Biomedical Engineering, IEEE Transactions on*, 47(10):1297 – 1307, oct 2000.
- [34] N. E. Miller. Learning of Visceral and Glandular Responses. *Science*, 163:434–445, January 1969.
- [35] K.D. Nielsen, A.F. Cabrera, and Omar Feix do Nascimento. Eeg based bci-towards a better control. brain-computer interface research at aalborg university. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 14(2):202 –204, june 2006.
- [36] Micheal O’Shea. The brain: a very short introduction. 2005.
- [37] Micheal O’Shea. A longstaff, neuroscience, taylor et francis, 2005. 2005.
- [38] R.C. Panicker, S. Puthusserypady, and Ying Sun. An asynchronous p300 bci with ssvep-based control state detection. *Biomedical Engineering, IEEE Transactions on*, 58(6):1781 –1788, june 2011.
- [39] Neil J. Perkins and Enrique F. Schisterman. The inconsistency of optimal cut-points obtained using two criteria based on the receiver operating characteristic curve. *American Journal of Epidemiology*, 163(7):670–675, 1 April 2006.
- [40] G. Pfurtscheller and C. Neuper. Motor imagery and direct brain-computer communication. *Proceedings of the IEEE*, 89(7):1123 –1134, jul 2001.
- [41] John Polich. P300 clinical utility and control of variability. *Journal of Clinical Neurophysiology*, 15:14–33, 1998.
- [42] Dale Purves. *Neuroscience, Fourth Edition*. Sinauer Associates, Inc., 4th edition, 7 2008.
- [43] J.Huggins M. Fatourechi S. Mason, J. Kronegg. Evaluating the performance of self-paced brain-computer interface technology. 2006.
- [44] Gerwin Schalk, Jonathan R Wolpaw, Dennis J McFarland, and Gert Pfurtscheller. Eeg-based communication: presence of an error potential. *Clinical Neurophysiology*, 111(12):2138 – 2144, 2000.
- [45] Reinhold Scherer, Alois Schloegl, Felix Lee, Horst Bischof, Janez Janša, and Gert Pfurtscheller. The self-paced graz brain-computer interface: methods and applications. *Intell. Neuroscience*, 2007:9–9, January 2007.
- [46] Eric W. Sellers and Emanuel Donchin. A p300-based brain-computer interface: Initial tests by als patients. *Clinical Neurophysiology*, 117(3):538 – 548, 2006.
- [47] Bernardo Dal Seno. Toward an integrated p300- and errp-based brain-computer interface. *PhD thesis, Politecnico di Milano*, 2009.

-
- [48] S. Sutton, M. Braren, J. Zubin, and E. R. John. Evoked-Potential Correlates of Stimulus Uncertainty. *Science*, 150:1187–1188, November 1965.
- [49] Sergios Theodoridis and Konstantinos Koutroumbas. *Pattern Recognition, Second Edition*. Academic Press, 2 edition, 4 2003.
- [50] AA. VV. *Enciclopedia medica italiana*. Sansoni, Firenze, Italy, 1955.
- [51] Jonathan R Wolpaw, Niels Birbaumer, Dennis J McFarland, Gert Pfurtscheller, and Theresa M Vaughan. Brain-computer interfaces for communication and control. *Clinical Neurophysiology*, 113(6):767 – 791, 2002.
- [52] Jonathan R. Wolpaw, Dennis J. McFarland, Gregory W. Neat, and Catherine A. Forneris. An eeg-based brain-computer interface for cursor control. *Electroencephalography and Clinical Neurophysiology*, 78(3):252 – 259, 1991.
- [53] J.R. Wolpaw, N. Birbaumer, W.J. Heetderks, D.J. McFarland, P.H. Peckham, G. Schalk, E. Donchin, L.A. Quatrano, C.J. Robinson, and T.M. Vaughan. Brain-computer interface technology: a review of the first international meeting. *Rehabilitation Engineering, IEEE Transactions on*, 8(2):164 –173, jun 2000.
- [54] Haihong Zhang, Cuntai Guan, and Chuanchu Wang. Asynchronous p300-based brain-computer interfaces: A computational approach with statistical models. *Biomedical Engineering, IEEE Transactions on*, 55(6):1754 –1763, june 2008.