

UNIVERSITÀ DEGLI STUDI DI PADOVA
FACOLTÀ DI INGEGNERIA

—
DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

—
TESI DI LAUREA MAGISTRALE IN BIOINGEGNERIA

SVILUPPO E TEST DI UN
SOFTWARE USER-FRIENDLY PER
STIMA E ANALISI DI RISPOSTE
EVOcate SINGLE-TRIAL

RELATORE: PROF. GIOVANNI SPARACINO

CORRELATORE: ING. ANAHITA GOLJAHANI

LAUREANDO: DIMITRI BONUTTO

ANNO ACCADEMICO 2012-2013

“La palla che lanciavi giocando nel parco non è ancora scesa al suolo.”

DYLAN THOMAS

Indice

Sommario	XI
Introduzione	XIII
1 I potenziali evocati evento-relati	1
1.1 Breve introduzione al segnale EEG	1
1.2 Potenziali evento-relati e P300	3
2 Due recenti metodi bayesiani di stima single-trial: B2S e MTL	15
2.1 Richiami sulla misura degli ERPs	16
2.2 L’Averaging convenzionale per la stima degli ERP medi	18
2.2.1 Tecnica dell’averaging	18
2.2.2 Limiti dell’averaging	20
2.2.3 Oltre l’averaging: tecniche single-trial	21
2.3 Un approccio two-stage per la stima degli ERPs: B2S	23
2.3.1 Primo step: stima del potenziale medio	23
2.3.2 Secondo step: stima single-trial	31
2.4 Un approccio one-stage per la stima degli ERPs: MTL	33
2.4.1 Stima della curva media e degli shift individuali	37
2.5 Note sui metodi bayesiani	39
2.6 Scopo della tesi	41
3 Metodi di peak-detection per l’analisi di risposte single-trial	43
3.1 Visual inspection	44
3.2 Peak-picking	44
3.3 Woody’s method	45

3.4	Curve fitting	48
3.5	Fractional area	50
3.6	ML method	51
3.7	EML method	52
3.8	Discussioni e conclusioni	52
4	Progettazione della GUI e integrazione in EEGlab	55
4.1	Costruire una GUI in MATLAB®	55
4.2	Fase di Mock-up	56
4.3	Il tool EEGlab	58
4.4	Realizzare un plug-in per EEGlab	60
4.4.1	Creare una Text Input Window	60
4.4.2	Aggiungere un plug-in	64
4.5	Componenti grafiche inseribili in una GUI	65
5	GUI dell’algoritmo MTL per la stima single-trial	69
5.1	Opzioni di scorrimento	70
5.2	Pannelli principali	71
5.2.1	Pannello relativo al pre-stimolo	71
5.2.2	Pannelli informativi	74
5.2.3	Pannello contenente i vari comandi	75
5.3	Plot Options	76
5.3.1	Variabilità	77
5.3.2	Visualizzazione della curva media	81
5.4	Altre opzioni	82
5.5	Aggiornamento delle stime	83
6	GUI per l’analisi delle latenze	85
7	Altri aspetti implementativi	91
7.1	Artifact rejection	91
7.2	Approccio bayesiano	93
7.3	Post-filtraggio e rimozione della media	93
7.4	Aspetti computazionali	95

7.5 Utilizzo su diversi sistemi operativi	96
Conclusioni	97
A Stima single-trial mediante modelli ARX	99
A.1 Un'alternativa all'approccio bayesiano	99
A.2 Aspetti implementativi	102
B Singular Value Decomposition	105
C Scheletro della GUI	109
Bibliografia	118
Bibliografia	119

Sommario

Questo lavoro di tesi si prefigge lo scopo di sviluppare uno strumento automatico e flessibile in grado di rilevare i parametri ERP di interesse (ampiezza e latenza della componente P300) mediante un'analisi single-trial condotta nel contesto di stima bayesiano attraverso due differenti approcci algoritmici (Bayesian Two Steps e Multi Task Learning), seguita da una fase di peak-detection. La potenziale utilità di tale software in ambito clinico ha orientato la fase implementativa verso la realizzazione di un tool profondamente integrato con EEGLAB, pacchetto open-source di MATLAB® ormai ampiamente diffuso nel contesto di ricerca.

La validazione della metodologia è effettuata su dati reali, e in particolare tracciati EEG registrati durante un compito odd-ball acustico di tipo classico.

Introduzione

Lo studio dei potenziali evento-relati (ERPs) è uno dei temi classici della ricerca neuro-scientifica. Infatti, gli ERPs misurati in risposta a stimoli sensoriali, cognitivi e motori sono cruciali nella comprensione di molti aspetti della neuro-fisiologia e, poiché la loro acquisizione è relativamente semplice e non invasiva, essi hanno anche diverse applicazioni cliniche. D'altro canto, l'estrazione di questi segnali è spesso molto difficile in quanto essi sono completamente immersi nell'attività elettroencefalografica spontanea (EEG) che ha una maggiore ampiezza ma comune contenuto spettrale. L'approccio tradizionalmente usato per estrarre gli ERPs, la cosiddetta media convenzionale (CA), ha delle riconosciute limitazioni. Alla luce di questo, sono state prese in considerazione delle metodologie più sofisticate aventi come scopo il miglioramento della stima media fornita dalla tecnica CA e l'estrazione single-trial.

In questa tesi viene utilizzato un approccio di stima bayesiano; in particolare sono presentati due metodi in grado di fornire sia una stima dell'ERP medio che degli ERPs single-trials. Il primo metodo è basato su una procedura a due passi [1]. Nel primo passo viene calcolato l'ERP medio, determinato come media pesata delle epoche a disposizione, una volta che queste ultime siano state individualmente filtrate sfruttando delle informazioni note a priori sulla statistica del secondo ordine dei segnali coinvolti. Nel secondo passo, gli ERPs delle singole epoche vengono stimati nello stesso contesto, usando l'ERP medio stimato al passo precedente come valore atteso a priori (segnale di riferimento).

Il secondo metodo è basato su una procedura multi-task learning ad un solo passo [2]. Diversamente dalla maggior parte degli approcci proposti in letteratura,

il metodo fornisce delle stime della risposta media e degli ERPs single-trial considerando simultaneamente tutte le sweeps a disposizione. I segnali ERPs incogniti sono trattati come “individui” di una popolazione omogenea e l’informazione disponibile per un’epoca è considerata utile per la stima di tutte le altre. Il metodo assume che la generica sweep possa essere modellata come la somma di tre componenti stocastiche indipendenti: una curva media di popolazione che è comune a tutte le epoche, uno shift individuale che differenzia ogni epoca dalle altre, una componente di rumore EEG di fondo che varia da un’epoca ad un’altra.

I due approcci appena descritti, uniti ad una fase successiva di peak-detection, vengono utilizzati in questo progetto di tesi con il preciso scopo di sviluppare uno strumento automatico e flessibile in grado di rilevare i parametri ERP di interesse (ampiezza e latenza della componente P300). L’interfaccia grafica sviluppata offrirà in tal modo enormi vantaggi all’utente, consentendogli di interagire con il computer attraverso una manipolazione grafica degli oggetti (visualizzazione delle stime single-trial e dei picchi d’interesse) in maniera molto user-friendly. Inoltre, la potenziale utilità di tale software in molti aspetti della neurofisiologia e, soprattutto, in diverse applicazioni cliniche ha orientato la fase implementativa verso la realizzazione di un tool profondamente integrato con EEGLab, pacchetto open-source di MATLAB® ampiamente diffuso nel contesto di ricerca. Per offrire un’idea della portata del software EEGLab basti pensare che l’articolo di riferimento [3] è stato citato più di 2600 volte e avvengono di media 40 download giornalieri del toolbox che ha ormai raggiunto ben 88 domini nazionali.

Capitolo 1

I potenziali evocati evento-relati

1.1 Breve introduzione al segnale EEG

Il segnale registrato dall'elettroencefalografo, detto elettroencefalogramma, rileva l'attività spontanea del cervello, la quale consta di componenti a diversa frequenza. Esse sono definite in letteratura come “ritmi EEG”, ognuno dei quali corrisponde in realtà ad una determinata banda spettrale del segnale (figura 1.1). Le prime oscillazioni ad essere introdotte furono quelle di frequenza compresa tra 8 e 13 Hz. Esse furono denominate da Berger “onde alfa”, proprio in riferimento alla prima lettera dell'alfabeto greco, e sono rilevate soprattutto nelle registrazioni effettuate sulle regioni posteriori del cervello di un adulto a riposo con gli occhi chiusi. Coerentemente, Berger coniò il termine “onde beta” per riferirsi al range di frequenze 12-30 Hz. Fu poi la volta di Adrian, che battezzò le oscillazioni a 40 Hz come “onde gamma”, sulla scia di quanto fatto dal suo predecessore. Le oscillazioni nella banda da 1 a 4 Hz, prendono il nome di “onde delta” svincolandosi dall'ordinamento sequenziale poiché la loro scoperta è più recente, così come quella delle “onde theta” (ritmo da 4 a 8 Hz) la cui lettera è riferita alla loro presumibile regione di provenienza: il talamo.

I segnali EEG sono caratterizzati da un'ampiezza che è normalmente compresa tra poche unità di μV e $500 \mu V$ (suddivisa in bassa se $< 30 \mu V$, media se compresa tra 30 e $70 \mu V$ e alta se $> 70 \mu V$), e una frequenza variabile nel range 0.5-100 Hz. Facendo riferimento alla denominazione sopracitata, le onde aventi

rilevanza clinica (ovvero le alfa, beta, gamma, delta e theta) presentano frequenza ed ampiezza indicate in figura 1.2.

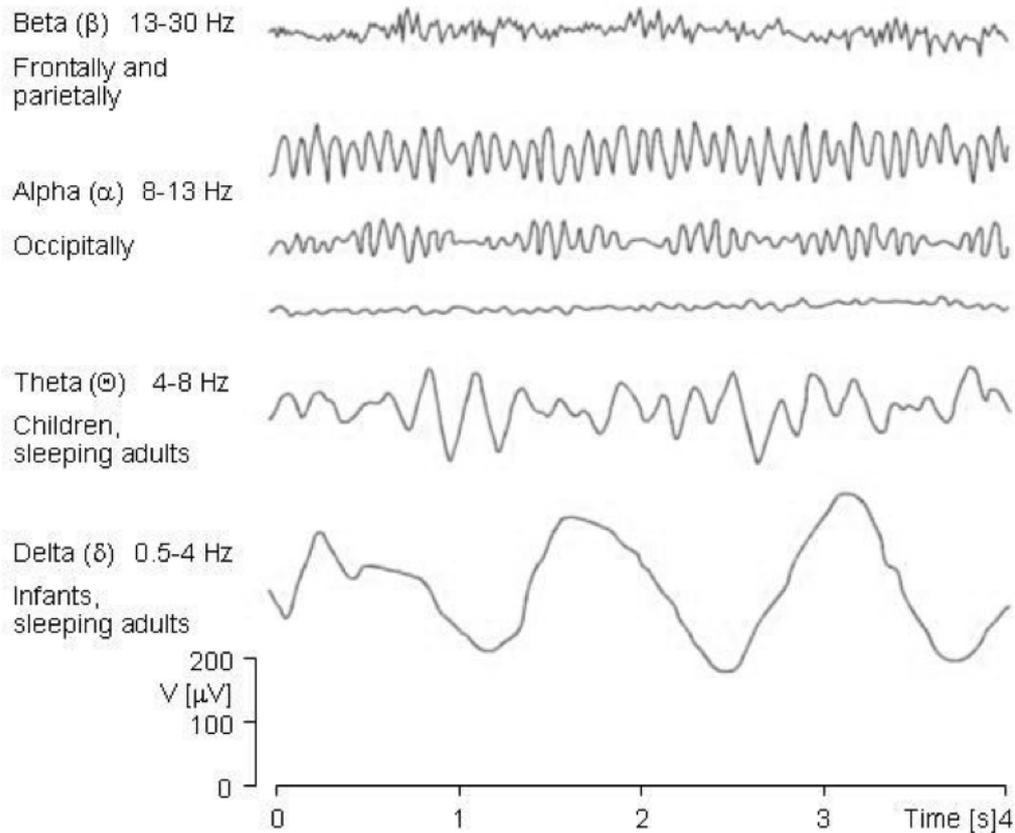


Figura 1.1: Esempi di onde EEG: i diversi ritmi presentano range di frequenze diversi.

L'EEG costituisce un esame fondamentale in ambito neurologico poiché consente di integrare la diagnosi di tumori, ischemie, epilessie ed alterazioni delle capacità cognitive ed attentive che si ripercuotono sull'attività elettrica del cervello alterandola in modo rilevante. Già attraverso l'ispezione visiva è infatti possibile comprendere modificazioni significative del tracciato EEG e inferire conclusioni coerenti sullo stato neurologico e clinico del soggetto. Oltre ad indagare l'attività neurale spontanea, risulta interessante studiare le peculiarità dell'EEG in conseguenza di un particolare stimolo esterno. La specifica risposta così suscitata, chiamata "potenziale evocato" (evoked potential), è fortemente dipendente dalla

Ritmo	Frequenza	Ampiezza	Caratteristiche
Gamma (γ)	più di 30 Hz	molto ridotta	Forte concentrazione, risoluzione di più problemi, meditazione.
Beta (β)	13-30 Hz	8-30 μV	Veglia, attenzione, cognizione, risoluzione di problemi, sonno onirico (REM).
Alpha (α)	8-13 Hz	15-45 μV	Rilassamento, tranquillità, riposo mentale, meditazione, allerta priva di ansia.
Theta (θ)	4-8 Hz	75 μV	Emozioni, rilassamento profondo, dormiveglia, creatività, utilizzo memoria.
Delta (δ)	0.5-4 Hz	150 μV	Sonno profondo privo di sogni, perdita della consapevolezza del corpo, accesso all'inconscio.

Figura 1.2: *Ritmi EEG: caratteristiche.*

stimolazione sensoriale esterna ed è caratterizzata da latenza, ampiezza e polarità. Si possono distinguere due tipologie di potenziali evocati [4]:

- potenziali evocati stimolo-correlati (o esogeni): dipendono in latenza ed ampiezza dalle caratteristiche fisiche dello stimolo, rappresentano la risposta obbligata di pool neuronali ad uno stimolo sottoposto mediante la via sensoriale cui essi fanno riferimento; la loro latenza varia nel range temporale della percezione sensoriale;
- potenziali evocati evento-relati, ERPs (o endogeni): vengono elicitati¹ in dipendenza del contenuto informativo dell'evento stimolante e del contesto psicologico in atto (dipendono dall'attenzione prestata allo stimolo e dal significato che il soggetto attribuisce ad esso); la loro latenza assume valori entro l'ambito temporale dei fenomeni attentivi e cognitivi.

1.2 Potenziali evento-relati e P300

Gli ERPs sono oggetto di studi e approfondimenti in quanto rappresentano la delicata interfaccia che trasduce le attività cerebrali e le funzioni cognitive nelle

¹Dal latino "elicere" (ex, fuori + lacere, attrarre), significa tirar fuori, trarre, far uscire. In psicologia, riferito a comportamenti o condotte, stimolarli, ottenerli mediante domande o altri stimoli.

relative manifestazioni comportamentali ed esperienze soggettive. Essi risultano relativamente poco esposti alla soggettività di medico e paziente, e nella maggior parte dei casi possono essere acquisiti in modo non invasivo e a prescindere dalla condiscendenza del soggetto. I potenziali evento-relati sono il risultato del cambiamento sincronizzato di potenziali post-sinaptici lenti, che avviene entro un elevato numero di neuroni corticali piramidali similmente orientati in una certa area della corteccia [5][4]. Si manifestano come brevi deflessioni monofasiche incorporate nell'EEG di fondo in preparazione o in risposta ad eventi discreti, siano essi interni o esterni al soggetto [4]. Dalla prospettiva dell'analisi in frequenza, gli ERPs sono dati dall'interazione di onde a frequenza di oscillazione diversa, ognuna delle quali possiede un particolare significato [6]. Facendo riferimento alla sopracitata classificazione, lo stato dell'arte rivela che:

- le onde delta e theta sembrano correlate con le funzioni di working memory, ovvero la memoria che ha come fine ultimo l'adempimento di un determinato compito e che è localizzata topograficamente a livello della corteccia prefrontale;
- le onde beta risultano praticamente nulle durante un atto motorio ed enfatizzate circa 1 secondo dopo l'azione stessa. Hanno sede nella regione sensorimotoria corrispondente alla zona del corpo coinvolta nell'atto. E' interessante menzionare che le onde beta sono osservabili anche nel caso in cui il movimento sia solo immaginato o durante altri processi cognitivi e possono essere indotte attraverso la stimolazione del nervo mediano;
- le onde gamma, ad alta frequenza, sono infine ipotizzate essere correlate con i processi cognitivi più elevati.

E' dunque chiaro come gli ERPs siano uno dei riflessi neurofisiologici più esplicativi del flusso di informazione cerebrale, in quanto capaci di veicolare i dettagli riguardanti la ricezione e il processamento a basso livello degli stimoli sensoriali, così come le elaborazioni che si pongono ad un livello più alto, ad esempio quelle riguardanti l'attenzione selettiva, l'aggiornamento della memoria, la comprensione semantica e altri tipi di attività cognitive [7].

Entrando ancora più nel dettaglio, gli ERPs sono forme d'onda caratterizzate da una sequenza di deflessioni positive o negative attivate in serie che vengono chiamate componenti [8] (rappresentate schematicamente in figura 1.3). Ciascuna componente è riconducibile a potenziali post sinaptici (eccitatori o inibitori) che risultano sincroni in un insieme di neuroni corticali sufficientemente ampio da poter essere rilevati in superficie. Accostando la conoscenza topografica di una componente e il suo significato, è possibile identificare la macro-area corticale attiva e l'istante temporale successivo al particolare stimolo. Le componenti degli ERPs sono quindi la manifestazione in tempo reale dell'attività elettrica cerebrale di un dato modulo neuro-anatomico associata ai diversi stadi di trasmissione ed elaborazione dell'informazione da parte del SNC. E' importante sottolineare che la forma d'onda visibile negli ERPs non è il risultato della sommazione lineare di processi indipendenti, bensì dell'interazione di meccanismi tra loro collegati e sinergici [8][6][4].

Esiste ad ogni modo il problema di localizzazione precisa della fonte neuronale del potenziale registrato: tale problema inverso² dà luogo infatti ad un numero pressoché infinito di soluzioni che devono essere ridotte ad una mediante l'uso sinergico di più tecniche di indagine, imaging, modelli neuro-anatomici, e attraverso la formulazione di supposizioni ed ipotesi addizionali circa il funzionamento del sistema nervoso [9]. L'analisi delle componenti degli ERPs è basata sui seguenti parametri:

- latenza: distanza temporale tra il momento di somministrazione dello stimolo e l'istante di comparsa della componente stessa;
- ampiezza: entità della deflessione della componente in esame; può essere riferita alla baseline oppure calcolata picco-picco (rispettivamente rispetto all'asse delle ascisse o rispetto al picco di un'altra componente);
- polarità: segno della deflessione (positivo o negativo);

²Un problema inverso è un contesto di indagine generico in cui vengono ricercate informazioni su una grandezza fisica, o più in generale su di un sistema, a partire da misurazioni o informazioni di tipo indiretto.

- topografia: posizione della superficie cranica dove è localizzata la massima ampiezza della componente.

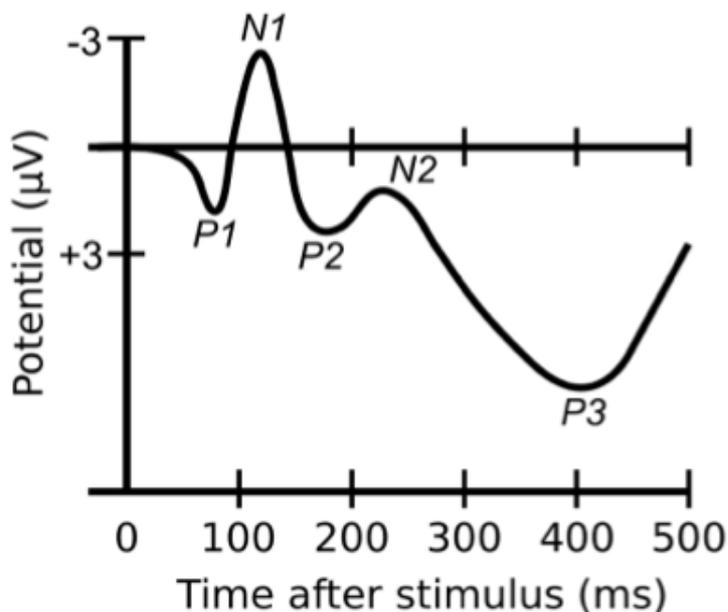


Figura 1.3: Forma d'onda che illustra diverse componenti caratteristiche degli ERPs. È interessante notare come l'asse delle ordinate risulti invertito, in accordo con la convenzione tipicamente diffusa nella pratica clinica.

Luck e collaboratori hanno stilato una lista di “dieci semplici regole” da seguire nello studio degli ERPs [10][6]:

- distinguere i semplici “picchi” dalle “componenti”: non è detto che ci sia nulla di fisiologicamente rilevante nel punto in cui il potenziale raggiunge un massimo o minimo locale;
- non limitarsi ad una singola onda per caratterizzare in latenza un picco o una componente in quanto possono esserci contributi, sovrapposizioni e coincidenze non ripetibili;
- ricordare che è estremamente pericoloso confrontare un effetto sperimentale con le forme d'onda del tracciato ERP grezzo;

- essere consapevoli che picchi di diversa ampiezza non significano necessariamente componenti di dimensione diversa, così come latenze diverse non corrispondono necessariamente a tempistiche diverse dell'intero complesso che identifica la componente;
- separare il concetto di variazione in ampiezza o latenza del picco da quello di variazione in ampiezza o latenza dell'ERP, soprattutto quando si introducono distorsioni dovute all'averaging;
- limitare quanto più possibile stimoli fisici che possano creare ambiguità nel tracciato o renderlo confuso; cercare di rendere omogenei gli stimoli;
- nel caso in cui ciò non fosse possibile, eseguire un esperimento di controllo che funga da riferimento;
- prestare attenzione al numero di trial che vengono confrontati (numero che incide fortemente sull'average finale e sul suo smoothing);
- essere cauti nell'analisi quando la risposta motoria varia tra le condizioni considerate;
- quando possibile, variare le condizioni sperimentali all'interno dei singoli blocchi di task piuttosto che tra un blocco e l'altro.

Approfondendo di un altro step il raggio d'analisi non resta che descrivere in dettaglio le singole componenti degli ERPs [8][4][7][11]. Nonostante sia indubbio che picchi e deflessioni costituiscono dei landmark importanti per orientare l'identificazione delle componenti ERPs, è inopportuno adottare la semplicistica teoria secondo cui picco (o valle) e componente sono sinonimi. La convenzione nell'assegnazione dei rispettivi nomi prevede che la prima lettera ('P' o 'N') identifichi la polarità della componente, mentre il numero che la segue sia rappresentativo dei millisecondi in cui tale componente si manifesta.

- P100: è rappresentata da un picco positivo, non sempre facilmente identificabile, posticipato di 50 ms o 100 ms circa rispetto allo stimolo a seconda che esso sia uditivo o visivo. La componente P100 è interpretata come un

indicatore neurofisiologico dell'attenzione allo stimolo sensoriale proposto, e dà informazioni sull'integrità del canale sensoriale utilizzato per la stimolazione. Mentre nel caso di stimolo uditivo questa componente compare prima dei 100 ms e si propaga dalla zona posteriore dello scalpo raggiungendo la sua massima ampiezza nelle regioni centrale e frontale, nella condizione di stimolo visivo la P100 si discosta dalla precedente in termini di meccanismi neurofisiologici, neurocognitivi, latenza del picco, distribuzione sullo scalpo e sorgenti neurali. In quest'ultimo caso infatti tale componente viene rilevata in corrispondenza del lobo occipitale e la sua ampiezza varia solitamente in base al grado di attenzione dedicato all'evento.

- N100: il suo picco – negativo – si presenta tipicamente dopo 100 ms rispetto alla presentazione dello stimolo, raggiungendo la sua massima ampiezza a livello delle aree fronto-centrali. Funzionalmente è correlata con le proprietà sensoriali e fisiche dello stimolo. Per quanto riguarda stimoli di tipo uditivo, l'ampiezza della N100 tende ad aumentare proporzionalmente all'attenzione. La maggior ampiezza della N100 visiva si manifesta nella regione occipitale, riducendosi nel caso in cui gli stimoli siano presentati a breve distanza nella sequenza temporale.
- P200: similmente alla P100 e alla N100, anche la P200 è considerata una componente obbligata poiché presenta una bassa variabilità inter-individuo e un'elevata replicabilità. Il picco è caratterizzato da polarità positiva e latenza di circa 200 ms. Questa componente viene elicitata in diversi task cognitivi, compresi quelli di attenzione selettiva e memoria a breve termine. Per quanto riguarda la P200 uditiva, essa si presenta spesso conglobata nella N100, condividendone anche molte delle caratteristiche. La sua localizzazione sullo scalpo è meno circoscritta e l'ampiezza massima è registrata in prossimità delle regioni frontali. La tipologia di stimolo uditivo (suono ad alta o bassa frequenza) influenza la morfologia di questa componente, così come l'incremento di intensità dello stimolo stesso ne determina l'aumento in ampiezza. Le sorgenti vere e proprie della P200 sono situate in corrispondenza delle cortecce uditive primaria e secondaria. Relativamente

alla P200 provocata da uno stimolo visivo, essa si manifesta con ampiezza maggiore in corrispondenza della zona frontale con uno shift della latenza tra 150 e 275 ms. La sua ampiezza cresce all'aumentare della complessità dello stimolo visivo o del task da portare a termine.

- N200: al contrario della precedente, la N200 è caratterizzata da una bassa ripetibilità poiché varia in modo sostanziale da individuo a individuo. La sua interpretazione psicologica verte essenzialmente su fattori quali la discriminazione dello stimolo, la selezione del target e l'orientamento della risposta. La sua durata si accorcia e la sua ampiezza diminuisce se gli intervalli inter-stimolo sono più brevi. La N200 si manifesta con maggiore ampiezza in corrispondenza della regione fronto-centrale, mentre la sorgente della componente pare essere localizzata nella corteccia uditiva sovratemporale. Incluso nel complesso N200 si trova anche il picco N170, massimo nell'area temporo-occipitale, il quale testimonia l'attività di processamento dei volti umani.
- MMN: mismatch negativity, deflessione negativa avente latenza piuttosto variabile (tra 100 e 250 ms) e massima ampiezza (tra 0.5 e 5 μV) nelle regioni frontale e centrale. Fu descritta per la prima volta nel 1978 da Naatanen, Gaillard e Mantysalo, che la individuarono in corrispondenza di paradigmi con una sequenza di stimoli variabili, diversi tra loro [7]. In altre parole la MMN si manifesta quando viene individuata automaticamente (anche da un soggetto distratto) una differenza tra lo stimolo precedente, di cui la memoria sensoriale ha tenuto traccia, e quello corrente, dal punto di vista della durata, dell'intensità, del tono della frequenza o di altre caratteristiche rilevabili. Com'è possibile dedurre dalle particolarità elencate, la componente MMN può essere impiegata per effettuare accertamenti sull'integrità della capacità di discriminazione, e risulta difficilmente elicetabile attraverso stimoli visivi.
- P300: ampio e vasto picco positivo che si presenta a circa 300 ms di latenza e ha ampiezza dipendente dal livello di attenzione e concentrazione del soggetto nei confronti di uno stimolo raro. La sua ampiezza massima

è localizzata in zona centro-parietale a livello mediano e la sua latenza è piuttosto variabile [12]. Individuata da Sutton e collaboratori nel 1965, costituisce senza dubbio la componente ad oggi più studiata [7]. Negli anni successivi, il fenomeno è stato scomposto da unitario ad un'interconnessione di diverse parti che riflettono una cascata di information processing la quale coinvolge meccanismi attentivi e di memoria. In particolare sono stati individuati due picchi che concorrono a caratterizzare la P300, denominati P3a e P3b [11][7][13], i quali sono collegati all'attenzione involontaria e al fenomeno di inibizione. Un evento irrilevante dal punto di vista del task proposto ma di per sé raro, nuovo o saliente, elicitava infatti una componente ERP positiva, la quale è stata etichettata come P3a. Essa si presenta come picco leggermente anticipato rispetto alla classica P300 (latenza variabile tra 250 e 300 ms) e avente ampiezza massima in corrispondenza delle aree mediane della parte fronto-centrale dello scalpo. Per quanto riguarda la P3b, essa è caratterizzata da una distribuzione per lo più concentrata in zona centro-parietale e da una latenza maggiore rispetto a quella della P3a. Il suo significato è correlato all'elaborazione della memoria, all'aspetto reattivo, e all'intenzione di risposta. Le sorgenti della P300 risultano ancora non del tutto chiare, ma sono state formulate diverse ipotesi basate su registrazioni intracraniche: fonti di questa componente potrebbero essere le strutture collocate nel lobo medio temporale, quali l'ippocampo, l'amigdala e il talamo [7]. Nel complesso, è ritenuta plausibile l'azione simultanea di più sorgenti, indipendenti o governate da un sistema integrato centrale, a generare una componente che risulta così diffusa nell'intero cervello. Come menzionato, la P300 si manifesta in risposta a degli stimoli rari denominati target, proposti in una sequenza randomizzata che li vede alternati a stimoli più frequenti, detti non target (paradigma odd-ball). Tale componente viene effettivamente elicitata solo dopo che lo stimolo è stato valutato e classificato, e mentre la sua ampiezza è direttamente proporzionale alla rarità dello stimolo, all'irregolarità del pattern di stimoli presentato, al tempo inter-stimolo e alla distanza tra uno stimolo target e il precedente, la sua latenza risulta in relazione diretta con la complessità di discernimento de-

gli stimoli [11] (si veda figura 1.4). Concludendo con una riflessione tratta da Donchin et al. [14], si può definire la P300 come “misura del livello di attività svolta da un processore corticale general-purpose”.

CONTEXT UPDATING THEORY OF P300

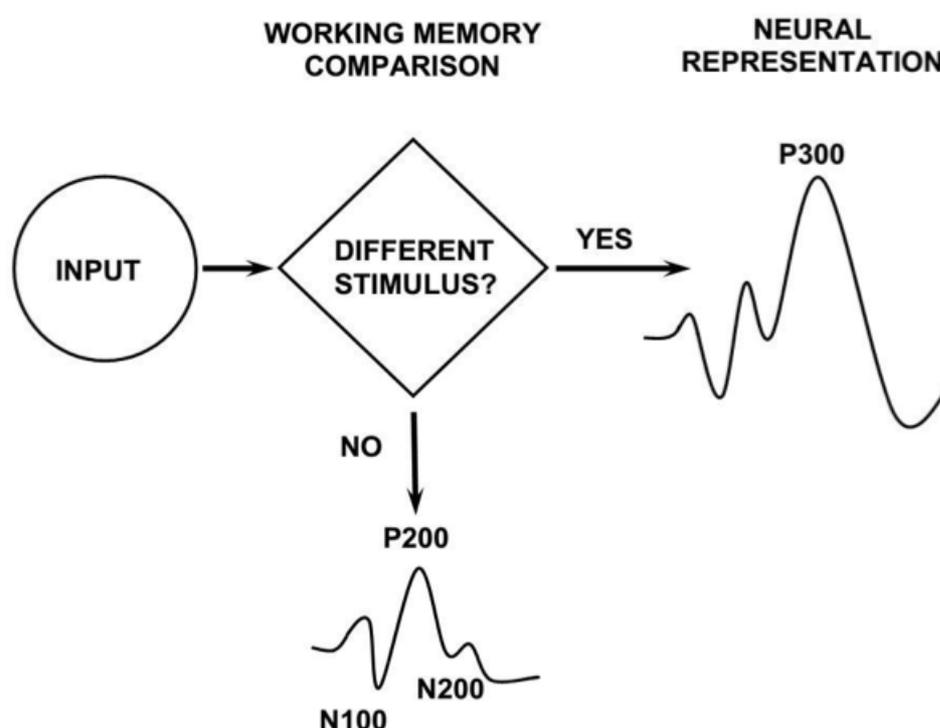


Figura 1.4: “Context-updating model” proposto da Polich et al.: lo stimolo entra nel sistema di elaborazione in cui un processo di confronto di memoria lo compara allo stimolo precedente determinandone l’uguaglianza o meno. Nel caso in cui gli stimoli non siano coincidenti, il soggetto alloca risorse attentive nei confronti dell’evento, e ciò determina l’elicitazione della P300 (P3b) in aggiunta alle componenti N100, P200 ed N200.

- N400: deflessione negativa che si presenta a 400 ms dopo l’inizio dello stimolo nel caso in cui vi sia un’incongruenza nel tipo di evento proposto. Più nel dettaglio la N400 fu identificata da Kutas e Hillyard nel 1980 durante un paradigma semantico nel quale le parole di una frase vengono presentate sotto forma di testo, una ad una, ad intervalli regolari [15]. Essi notarono che

inserendo come parole finali della frase due termini non congrui dal punto di vista del significato (e non della sintassi) rispetto al resto, al posto di una forma d'onda positiva tra 200 e 600 ms si manifestava una significativa componente di ampiezza negativa. La N400 è al centro di numerosi studi atti ad individuare l'esatto processo cognitivo che ne sta alla base e le sorgenti neurali che ne sono responsabili.

- P600: componente positiva avente l'onset a circa 500 ms e che raggiunge il suo picco massimo intorno ai 600 ms. Interpreta due fenomeni diversi tra loro: i processi di memoria e i processi associati al linguaggio. Può essere elicitata mediante task sia visivi che uditivi, somministrando al soggetto tipologie di stimoli che presentano errori grammaticali o sintattici, anomalie semantiche o musicali. Le più diffuse interpretazioni collegano questa componente a: processi di revisione (correzione della frase che deve essere interpretata e processata), processi di rianalisi (riarrangiamento coerente della struttura della frase), tempo e sforzo cognitivo necessari per portare a termine l'esercizio. Esiste anche l'ipotesi che la P600 sia in un certo senso simile alla P300, ovvero in relazione non tanto con i processi linguistici in senso stretto, quanto piuttosto con stimoli improbabili o inaspettati (dal momento che errori come quelli elencati sono rari in natura).

Modificazioni alle caratteristiche di ampiezza e latenza tipiche delle suddette componenti, possono non solo significare deficit cerebrali, ma orientarne l'identificazione e la diagnosi [7] integrando il quadro clinico delineato dai test di tipo neuro-psicologico. Tra le citate, la componente P300 riveste il ruolo di protagonista in questo lavoro di tesi, incentrato sul suo studio approfondito e sulla sua soddisfacente individuazione sia in termini di ERP medio che nel contesto delle singole forme d'onda relative agli stimoli presentati. Tale caratterizzazione mira a dissociare, tramite il ricorso ad un modello di stima bayesiano, la reale componente P300 dal vasto panorama di attività elettrica corticale che fa da sfondo ai processi cognitivi responsabili dell'elicitazione stessa della componente in analisi. Considerando come rumore l'attività elettrica di base (EEG di background), non correlata con i processi cognitivi in atto durante il task, è possibile ottenere una

rappresentazione specifica ed informativa della P300. Ciò può essere conseguito a fronte di un certo numero di ipotesi approfondite nel prossimo capitolo.

L'intento del prossimo capitolo è offrire una visione d'insieme circa le tecniche di analisi dei potenziali evento-relati che, pur occupando da tempo un ruolo di fondamentale rilievo in ambito clinico, presentano ampie potenzialità di miglioramento. Vengono illustrati in tal senso due approcci di stima single-trial nel contesto bayesiano.

Capitolo 2

Due recenti metodi bayesiani di stima single-trial: B2S e MTL

In questo capitolo viene presentato l'approccio Bayesiano (descrizione non funzionale del potenziale single-trial) per la stima single-trial dei potenziali evento correlati. In particolare, in questo lavoro di tesi, vengono implementati due metodi differenti:

- B2S [1]: basato su una procedura two-stage dove nel primo passo ogni raw-sweep viene filtrata sfruttando delle informazioni note a priori sulla statistica del secondo ordine dei segnali coinvolti (EEG di fondo e ERP da stimare), in questo modo viene stimato un average-ERP come media pesata delle N sweep filtrate, il quale viene poi utilizzato nel secondo passo di stima come risposta attesa a priori;
- MTL [2]: basato su una procedura multi task learning one-stage dove è possibile ottenere la stima della media e le risposte single-trial processando solo una volta e simultaneamente le N sweep. La componente ERP viene quindi stimata dall'analisi congiunta di tutte le tracce acquisite da uno stesso soggetto che costituiscono in tal modo la popolazione omogenea.

2.1 Richiami sulla misura degli ERPs

La misurazione del segnale EEG deve tenere in considerazione problematiche legate alla presenza di un fattore inficiante: il rumore. Nonostante la strumentazione sia concepita per registrare l'attività cerebrale, il tracciato registrato risulta infatti costituito non solo dal segnale desiderato, ma anche da componenti indesiderate chiamate artefatti, i quali possono avere origine fisiologica o non fisiologica [16]. Al primo gruppo appartengono tutti quei segnali indesiderati che hanno origine dal paziente e che, di norma, sono più complicati da individuare e da eliminare. I più diffusi artefatti fisiologici sono:

- artefatti oculari: generati a motivo del fatto che il bulbo oculare è assimilabile ad un dipolo avente polo positivo a livello della cornea (e quindi in posizione anteriore). Nel caso in cui l'occhio ruoti si ingenera di conseguenza un potenziale (di tipo corneoretinale o elettroretinale) che viene inevitabilmente rilevato dagli elettrodi nelle vicinanze, e specialmente da quelli in posizione frontale. Gli artefatti da movimento oculare risultano facilmente riconoscibili poiché caratterizzati da ampiezza superiore a quella del segnale EEG e minore frequenza. Le tecniche di rimozione degli artefatti oculari si basano sull'analisi incrociata dell'EEG con l'EOG (elettrooculogramma), il quale viene nella maggior parte dei casi co-registrato. Tali tecniche intervengono mediante sogliatura, rimuovendo cioè le tracce che presentano ampiezza superiore ad una certa soglia, mediante tecniche di cross-correlazione (vengono quindi scartate le tracce EEG che dimostrano un comportamento influenzato da quello dell'EOG) oppure mediante altri approcci correttivi (e quindi in grado di preservare la traccia, ripulendola invece di escluderla dal dataset) efficientemente riassunti da Fatourechi e collaboratori [16]. Esiste anche un altro tipo di artefatto oculare, dovuto al movimento muscolare necessario per muovere l'occhio e le palpebre (il quale, in quanto muscolare, rientra nel prossimo gruppo in elenco). Esempio classico è il cosiddetto blink, ovvero la rapida chiusura e riapertura delle palpebre;
- artefatti muscolari: tipologia di disturbo molto comune, dovuta ai potenziali d'azione che attraversano le fibre muscolari per indurne la contrazione

e dare atto al movimento. Nel caso della registrazione EEG, i principali muscoli della testa a causare interferenze sono quelli frontali e temporali. I potenziali mioelettrici presentano una durata più breve ed una banda di frequenze più elevata rispetto a quelli cerebrali, caratteristiche che candidano automaticamente l'applicazione di un filtro passa-basso come soluzione da adottare per ripulire il segnale di interesse. Per quanto riguarda una trattazione più approfondita circa questa tipologia di artefatti, si rimanda al dettagliato quadro presentato nello studio di Shackman e collaboratori [17];

- artefatti cardiaci: artefatti tipicamente ritmici dovuti alla variazione di potenziale cardiaco necessaria al battito, la quale si propaga fino allo scalpo. La sua ampiezza varia in dipendenza della localizzazione dell'elettrodo di riferimento utilizzato, e in particolare è considerevole nel caso in cui le reference siano i lobi auricolari. La caratteristica di regolarità e ripetitività rende questo tipo di artefatto facilmente individuabile, soprattutto per confronto tramite una co-registrazione ECG;
- artefatti da pelle: provocati dall'alterazione delle impedenze degli elettrodi a causa di processi fisiologici quali la sudorazione. Essi si manifestano come drift, ovvero lente derive del potenziale rilevato, e sono pertanto rimovibili mediante un filtraggio passa-alto con frequenza di taglio molto piccola (es. $0.01Hz$). Degna di nota è – in tutti i casi – la tecnica di rimozione basata su Independent Component Analysis (ICA) [18], che mira a dissociare il segnale originale multivariante nella combinazione lineare di più sorgenti indipendenti tra loro. La trattazione specifica di questa tecnica, chiamata anche “Blind Source Separation”, esula dagli scopi di questa tesi; per il suo approfondimento si rimanda dunque alla letteratura specifica [19].

Tra gli artefatti non fisiologici vi sono invece:

- artefatti da elettrodo: sono dovuti al contatto non adeguato tra un elettrodo e lo scalpo, a causa del posizionamento non corretto o del banale movimento fisico dell'elettrodo stesso. Tale artefatto si traduce in brusche variazioni

nel tracciato EEG che si verificano all'istante del distacco (discontinuità del valore di impedenza);

- artefatti da strumentazione: si tratta di disturbi ingenerati dalla strumentazione EEG. Possono derivare ad esempio dal cattivo contatto tra le boccole degli elettrodi e la testina di acquisizione. Nel caso in cui vi siano invece componenti indesiderate che si accoppiano al segnale utile, è possibile adottare provvedimenti di schermatura delle apparecchiature e appropriate tecniche di filtraggio [16]. Essendo tuttavia usuale che il segnale esterno e quello utile condividano almeno in parte la banda in frequenza, è necessario studiare accuratamente la frequenza di taglio e la banda passante del filtro, in modo tale da ripulire il segnale desiderato sacrificandone però meno contenuto informativo possibile;
- artefatti da rete: il rumore della rete elettrica di alimentazione si manifesta come disturbo additivo, tipicamente di modo comune, alla frequenza di 50Hz . Esso è causato da una non perfetta messa a terra dell'apparecchio (l'impedenza dell'amplificatore tra elettrodo e terra è troppo elevata), e può essere rimosso con un filtro notch centrato nella frequenza critica.

2.2 L'Averaging convenzionale per la stima degli ERP medi

2.2.1 Tecnica dell'averaging

Nel contesto dell'estrazione dei potenziali evento-correlati (ERPs), anche l'EEG di background è assimilato a rumore, in quanto segnale indesiderato sovrapposto alla componente di interesse. Il rapporto segnale/disturbo (SNR, Signal to Noise Ratio) tra ERP ed EEG risulta piuttosto ridotto in quanto l'EEG presenta ampiezza maggiore rispetto all'ERP (decine di μV contro μV) [20][21], e purtroppo ha contenuto spettrale in comune con quest'ultimo. Si rivela di conseguenza improponibile un filtraggio in frequenza ove l'obiettivo sia quello di isolare il segnale ERP dall'EEG di fondo. Ciò detto, si passa quindi dalla fase di elaborazione pre-

liminare alle metodiche in grado di estrapolare il segnale desiderato fornendone una stima attendibile e rendendolo dunque adeguato per il successivo stadio di analisi. Lo studio e l'individuazione dei potenziali evento-correlati e delle componenti di interesse presenti in essi possono essere condotti secondo una grande varietà di filosofie. L'approccio più semplice e classico è quello basato sull'averaging convenzionale (o media coerente), ovvero sul calcolo della media dei segmenti temporali di interesse dell'EEG, chiamati epoche o sweep. Essi coincidono con le parti di tracciato che seguono (ed eventualmente precedono) uno stimolo uguale nel medesimo task cognitivo (che prevede la ripetizione sequenziale e randomizzata di un certo numero N di stimoli). In pratica si ipotizza che contengano lo stesso tipo di potenziale evocato. La media $y_m(t)$ delle N sweep $y_i(t)$ acquisite risulta dunque:

$$y_m(t) = \frac{1}{N} \sum_{i=1}^N y_i(t) = \frac{1}{N} \sum_{i=1}^N (u_i(t) + v_i(t)) = u(t) + \frac{1}{N} \sum_{i=1}^N v_i(t) \quad (2.1)$$

Dove con $u_i(t)$ si intende l'ERP contenuto nell' i -esima sweep, con $v_i(t)$ il rumore EEG di background sovrapposto all'ERP di interesse (secondo un modello additivo), e con $u(t)$ l'ERP deterministico, assunto uguale in tutte le sweep. L'averaging si rende necessario poiché, essendo l'ampiezza dell'ERP ridotta rispetto al rumore di fondo, artefatti e altri "disturbi", esso consente, idealmente, di annullare l'effetto di questi ultimi, esaltando invece le componenti comuni.

La latenza (intervallo di tempo dallo stimolo) di un dato potenziale evocato tende infatti ad essere approssimativamente costante o con piccole variazioni di fase, mentre le fluttuazioni dell'EEG di background (che si ricorda essere annoverato come disturbo nel contesto dell'analisi ERP) risultano del tutto scorrelate e non sincronizzate. La risposta evocata media estratta è in questo modo ripulita dalle oscillazioni casuali e sfasate, che vengono quindi elise dal semplice processo di media rappresentato dall'ultimo addendo della relazione 2.1.

Il vettore y_m può essere concepito come una stima non polarizzata dell'ERP, poiché fornisce in media il potenziale evento-correlato di interesse. Il calcolo del suo valore atteso conduce infatti a:

$$E[y_m(t)] = E[u(t)] + \frac{1}{N}E\left[\sum_{i=1}^N v_i(t)\right] = u(t) \quad (2.2)$$

E' inoltre possibile caratterizzare la precisione della stima utilizzando come indice la varianza dell'errore di stima:

$$\text{var}[e(t)] = E\left[\left(\frac{1}{N}\sum_{i=1}^N v_i(t)\right)^2\right] = \frac{1}{N^2}E\left[\sum_{i=1}^N v_i^2(t)\right] = \frac{\sigma^2}{N} \quad (2.3)$$

Dal calcolo della varianza d'errore si può dedurre che la precisione dello stimatore basato su averaging convenzionale cresce all'aumentare del numero N di sweep mediate, diventando ipoteticamente pari a zero per $N \rightarrow \infty$. La medesima tendenza è dimostrata dal rapporto segnale/disturbo (SNR), il quale risulta direttamente proporzionale al numero N di sweep utilizzate nel processo di media.

2.2.2 Limiti dell'averaging

Illustrato l'approccio su cui si basa la tecnica di averaging, emergono alcuni punti critici tutt'altro che irrilevanti [22][1][5]:

- variabilità dell'ERP: per un funzionamento ottimale della tecnica, l'ERP elicitato dovrebbe non cambiare da una sweep all'altra. Tuttavia gli ERPs risultano lunghi dall'essere stazionari¹ e quindi perfettamente correlati allo stimolo che li provoca. Costano invece di componenti multiple le cui ampiezza e latenza possono variare indipendentemente l'una dall'altra da sweep a sweep, a causa sia delle dinamiche di processamento, sia del fenomeno di abitudine². In questo senso, se varia l'ampiezza il valor medio dell'ERP risulterà una stima non distorta, al più attenuata, della media della distribuzione delle singole sweep; se invece varia la latenza (ovvero se si verifica un rilevante jitter in latenza), la stima finale dell'ERP potrà risultare fortemente distorta oltre che di ampiezza ridotta. E' inoltre immediato

¹Si pensi ad esempio a come latenza e morfologia siano tutt'altro che invarianti da epoca a epoca, facendo cadere l'ipotesi di stazionarietà.

²Graduale diminuzione dell'attenzione e della risposta di un organismo a uno stimolo, a seguito del ripetersi dello stimolo stesso.

aggiungere che con questo approccio la variabilità intra-soggetto dell'ERP non può in alcun modo essere valutata;

- non stazionarietà dell'EEG: l'ipotesi che sta alla base dell'averaging è che l'EEG di background (trattato come rumore indesiderato poiché sovrapposto all'ERP) sia un segnale casuale, a media nulla e varianza σ^2 , non correlato e stazionario durante la registrazione delle N sweep. Tuttavia tale assunzione è molto lontana dall'essere vera: la stima così ottenuta è da considerarsi dunque sub-ottima. Il rumore EEG è assimilabile infatti ad un modello stocastico solo per brevi tratti;
- informazioni a priori: non viene sfruttata alcuna conoscenza a priori, nonostante sia possibile tenere in considerazione una serie di informazioni note su EEG ed ERP. Per questa ragione al fine di ottenere un ERP medio accettabile è necessario avere a disposizione un elevato numero di sweep prima di fare ricorso all'algoritmo (da qualche decina a qualche migliaio in dipendenza del tipo di ERP e del SNR per la singola epoca). Più in particolare Cohen e Polich [23] suggeriscono di mediare 20 sweep come numero ottimale per avere una stima stabile in ampiezza, mentre Duncan e collaboratori [7] hanno individuato come numero minimo 36 sweep, se nella pipeline è presente una fase preliminare di artifact-rejection o artifact-correction. Tuttavia per l'elicitazione di alcune particolari componenti di interesse (ad esempio la cosiddetta P300), è necessario che tale numero di stimoli target, sia accompagnato da un numero di stimoli non-target in proporzione 4:1. La quantità di stimoli cui il soggetto deve essere sottoposto affinché possano essere estratti dall'EEG un numero sufficiente di ERP, risulta di conseguenza proibitiva e, oltre a rappresentare un forte limite, è in disaccordo con la tendenza clinica che prevede di minimizzare alcuni fattori quali la durata e la pesantezza della prova, e soprattutto a limitare il fenomeno di abitudine.

2.2.3 Oltre l'averaging: tecniche single-trial

Appare ora probabilmente più chiaro come sia indispensabile sviluppare tecniche maggiormente sofisticate rispetto all'averaging convenzionale. Tecniche che

superino il limite di ipotesi ed assunzioni quasi mai rispettate in senso stretto. Tecniche che non richiedano la disponibilità di un elevato numero di sweep. Tecniche che siano in grado di preservare informazioni cliniche che si manifestano all'interno della singola epoca. In altre parole, la necessità è dunque proprio quella di avere a disposizione un'adeguata metodologia single-trial. Questa esigenza ha spinto dapprima molti gruppi di ricerca ad implementare nuovi approcci non single-trial anche molto diversi tra loro. Una suddivisione grossolana può essere fatta raggruppando le tecniche che lavorano nel dominio del tempo (es. modelli autoregressivi), dello spazio (es. Principal Component Analysis (PCA), Independent Component Analysis (ICA)), o dello spazio e del tempo contemporaneamente (es. wavelet, coerenza). Tecniche nel dominio della sola frequenza sono praticamente inesistenti in quanto incorrono in un problema decisivo: la componente spettrale del segnale utile risulta spesso almeno parzialmente sovrapposta a quella del rumore da rimuovere. Non è quindi possibile settare una frequenza di taglio "ottima" in grado di separare i due contributi. Di conseguenza, ad esempio, algoritmi di media pesata o basati sull'uso di informazioni a priori sono stati concepiti per migliorare l'averaging convenzionale dal punto di vista delle sweep richieste al fine di ottenere una stima stabile ed affidabile dell'ERP. Numerose altre tecniche hanno invece varcato la soglia dell'analisi su singola sweep, ma per una loro rassegna esaustiva si rimanda alle review di Blankertz e collaboratori [21], e Mouraux e collaboratori [5]. Nessuna di esse, tuttavia, è riuscita ad imporsi come standard in ambito clinico, dove l'approccio convenzionale rimane quello basato sull'averaging. In questo lavoro di tesi è presentata una tecnica di analisi single-sweep appartenente al contesto bayesiano, avente la potenzialità di impattare fortemente con gli attuali metodi standard ed apportare notevoli vantaggi in ambito clinico.

Un'alternativa a questo approccio è rappresentata da una tecnica parametrica altrettanto sofisticata e versatile, che prevede una descrizione non funzionale del potenziale single-trial: quella dei modelli ARX (per maggiori dettagli si rimanda all'Appendice A).

2.3 Un approccio two-stage per la stima degli ERPs: B2S

L'algoritmo B2S (Bayesian Two Steps) è basato, come si evince dal nome, su una procedura two-stage [1]. Nel primo passo tutte le sweeps disponibili vengono processate in modo da ottenere un ERP medio che funge da segnale di riferimento ed è quindi utilizzato come valore atteso a priori, nel secondo step, per la stima vera e propria.

2.3.1 Primo step: stima del potenziale medio

Modello dei dati

Viene utilizzato un modello additivo vettoriale

$$y = u + v \quad (2.4)$$

dove y è il vettore di dimensione n contenente i dati registrati, u è il vettore di dimensione n contenente i campioni del potenziale evento correlato e v è il vettore a media nulla di dimensione n contenente i campioni del rumore che corrompe il segnale utile, ovvero il segnale EEG di fondo sommato ad altre sorgenti di rumore e artifatti. I vettori u e v sono assunti indipendenti tra di loro, ipotesi ragionevole in quanto le risposte non vengono considerate generate da possibili mutamenti dell'attività di background. L'ipotesi fondamentale rimane tuttavia quella di considerare questi segnali come realizzazioni di processi stocastici in modo da permettere la stima degli ERPs in un contesto Bayesiano. La stima quindi risulta ottima e consente di minimizzare l'errore quadrato medio.

Modello del rumore EEG

Per interpretare adeguatamente il segnale EEG di fondo, che si ricorda essere parte preponderante dell'errore di misura totale, si ricorre ad un modello autoregressivo. Tale scelta, ampiamente accettata in letteratura, consente di formulare una descrizione quantitativa di media e varianza del segnale \mathbf{v} (vettore supposto a media nulla e correlato). L'ipotesi fondamentale è considerare il segnale EEG

un processo stazionario per al più 2 secondi (è ritenuta pertanto più che adeguata la registrazione di un pre-stimolo di durata pari a 500 ms) il che ci consente di descriverlo su brevi intervalli con un modello AR di ordine p .

Essendo

$$v_k = -a_1 v_{k-1} - a_2 v_{k-2} - \dots - a_p v_{k-p} + \varepsilon_k \quad (2.5)$$

con

$$\varepsilon_k \sim WN(0, \sigma^2) \quad (2.6)$$

passando alla forma vettoriale, si può descrivere la covarianza a priori del segnale EEG di background come

$$\Sigma_v = \sigma^2 (A^T A)^{-1} \quad (2.7)$$

con A matrice quadrata di Toeplitz di dimensione n (con n numero di campioni per sweep, nell'ipotesi di campionamento standardizzato), triangolare inferiore con prima colonna $[1, a_1, a_2 \dots a_p, 0 \dots 0]^T$.

Il protocollo di acquisizione prevede di sfruttare un intervallo di solo rumore precedente allo stimolo (il cosiddetto pre-stimolo) e identificare varianza e coefficienti del modello autoregressivo per ogni sweep i -esima ³.

La presente assunzione che il potenziale evento-correlato venga elicitato esclusivamente dopo l'istante di stimolazione risulta per certi versi critica, ma comunque accettabile.

L'ordine ottimo del modello viene stimato tramite l'Akaike Information Criterion (AIC) o il Final Prediction Error (FPE).

Ecco come appare in forma esplicita la matrice A :

³Per semplicità di trattazione le matrici non vengono descritte con l'apice i ma è di fondamentale importanza sapere che per ciascuna delle N sweep A , σ^2 e di conseguenza anche Σ_v sono diversi.

$$\begin{pmatrix} 1 & 0 & \dots & 0 & 0 & 0 \\ a_1 & 1 & 0 & \ddots & 0 & 0 \\ \vdots & a_1 & 1 & 0 & \ddots & 0 \\ a_p & \ddots & a_1 & 1 & 0 & \vdots \\ 0 & a_p & \ddots & a_1 & 1 & 0 \\ 0 & 0 & a_p & \dots & a_1 & 1 \end{pmatrix} \quad (2.8)$$

Modello dell'ERP

Come prima cosa è opportuno notare che non si conosce nulla di quantitativo sul potenziale incognito, è quindi necessario svolgere un'indagine di tipo qualitativo. Indipendentemente dai dati ci si aspetta che il segnale ERP sia regolare.

In questo modo non ci si sbilancia introducendo vincolanti, complicate e mai esaustive assunzioni sulla morfologia dell'ERP, la shape del quale costituisce in effetti l'obiettivo ultimo. Un modo semplice e versatile (ma cionondimeno riconosciuto valido in letteratura) di fornire una descrizione probabilistica a priori di un segnale "smooth" con regolarità nota è quello di assimilarlo ad un modello random walk semplice o integrato. Ciò significa che, più in particolare, il segnale ERP viene interpretato come realizzazione di un processo stocastico ottenuto come cascata di d integratori che intervengono su un processo di rumore bianco a media nulla e varianza λ^2 . Nello specifico caso di modello (descrizione non parametrica) random walk non integrato, l'equazione descrittiva per la sweep generica è:

$$u_k = u_{k-1} + w_k \quad (2.9)$$

per $k = 1, 2, \dots, n$ con u_k causale e

$$w_k \sim WGN(0, \lambda^2) \quad (2.10)$$

In altri termini ciò significa che se un processo aleatorio è regolare, risulta improbabile che la differenza tra due campioni consecutivi presenti variazioni

brusche. Viene invece considerata più verosimile la condizione in cui due campioni consecutivi assumono valori prossimi tra loro. Della variabile aleatoria w_k è possibile fornire una descrizione, come detto in precedenza, solamente qualitativa: la si caratterizza – nella fattispecie – come variabile aleatoria a media nulla, varianza finita (e incognita) pari a λ^2 .

Definendo la forma matriciale generale dell'equazione come $Fu = w$, dove $F = \Delta^d$ è la matrice delle differenze d -esime (con d numero di integratori; $d = 1$ nel caso di random walk semplice), è possibile generalizzare l'espressione della matrice di covarianza:

$$\Sigma_u = \lambda^2(F^T F)^{-1} \quad (2.11)$$

dove λ^2 è, come detto in precedenza, la varianza incognita ⁴ e Δ è la matrice quadrata di Toeplitz triangolare inferiore di dimensioni $n \times n$ (con n numero di campioni della sweep), avente come prima colonna il vettore $[1, -1, 0 \dots 0]^T$.

Ecco come appare in forma esplicita la matrice A:

$$\begin{pmatrix} 1 & 0 & \dots & 0 & 0 & 0 \\ -1 & 1 & 0 & \ddots & 0 & 0 \\ 0 & -1 & 1 & 0 & \ddots & 0 \\ \vdots & 0 & -1 & 1 & 0 & \vdots \\ 0 & \ddots & 0 & -1 & 1 & 0 \\ 0 & 0 & \dots & 0 & -1 & 1 \end{pmatrix} \quad (2.12)$$

Valori tipici per d sono compresi tra 1 e 6: per $d = 1$ l'ERP incognito viene descritto – come visto – da un modello random walk semplice, il quale nel contesto bayesiano sottintende che, dato u_{k-1} , il valore u_k si troverà con probabilità pari al 99.7% nel range $u_{k-1} \pm 3\lambda$ (sempre, chiaramente, in riferimento alla sweep i -esima). Ciò significa che più basso è il valore di λ calcolato per la sweep corrente, più il suo ERP u sarà stimato come segnale smooth. Valori relativamente elevati (ma comunque minori di 6) sono associabili alla modellizzazione di ERPs spiky,

⁴Anche in questo caso, per semplicità di trattazione le matrici non vengono descritte con l'apice i ma si utilizza una notazione più compatta, in ogni caso è di fondamentale importanza sapere che per ciascuna delle N sweep λ^2 e di conseguenza anche Σ_u sono diversi.

cioè rapidi e irregolari, mentre valori più bassi sono da preferire nella descrizione di profili ERPs lenti e smussati. Per una descrizione più approfondita si rimanda a [20]. Risulta comunque utile notare come, mentre la matrice A sia caratteristica della singola sweep in quanto calcolata sul particolare pre-stimolo, la matrice F dipenda unicamente dalla scelta preliminare del numero di integratori, e sia dunque la medesima per tutte le sweep.

Determinazione della stima media

Se il potenziale evento correlato e il segnale EEG di fondo sono assunti scorrelati, la stima lineare a minima varianza d'errore⁵ che si ottiene dal modello delle misure 2.4 è semplicemente

$$\hat{u} = (\Sigma_v^{-1} + \Sigma_u^{-1})^{-1} \Sigma_v^{-1} y \quad (2.13)$$

dove y è il vettore che contiene i vettori dei dati grezzi (anche in questo caso per semplicità di trattazione tralasciamo l'apice i che sta a indicare il numero della sweep) e Σ_v e Σ_u sono le matrici di covarianza a priori (indipendenti dai dati y) dei vettori v e u ⁶.

Se all'espressione generica sostituiamo le espressioni per le matrici di covarianza trovate in precedenza risulterà

$$\hat{u} = (A^T A + \gamma F^T F)^{-1} A^T A y \quad (2.14)$$

con $\gamma = \frac{\sigma^2}{\lambda^2}$ stimabile a posteriori con i criteri statistici di smoothing presentati nel paragrafo successivo. La diversità dei vari parametri e delle varie matrici per ciascuno degli N sweep permette inoltre di rilasciare l'ipotesi di stazionarietà dell'EEG e quella dell'invarianza del potenziale evocato (ipotesi cardine dell'averaging convenzionale).

⁵Per stima a minima varianza d'errore si intende che \hat{u} minimizza la varianza dell'errore di stima $E[||u - \hat{u}||^2]$.

⁶ $E[u]$ e $E[y]$ nella determinazione della stima media vengono considerati uguali a 0, il primo termine nel secondo step assumerà il valore stimato in questo primo step e di conseguenza risulterà $E[y] = E[u] + E[v] = E[u]$.

La stima media viene infine determinata come media pesata delle N sweep processate:

$$\bar{u} = \frac{1}{\sum_{i=1}^N w_i} \sum_{i=1}^N w_i \hat{u}^i \quad (2.15)$$

Per la scelta dei pesi si sfrutta la teoria della stima, infatti è ricavabile una misura della varianza dell'errore di stima:

$$\text{cov}(\tilde{u}^i) = \sigma_i^2 (A_i^T A_i + \gamma_i F^T F)^{-1} \quad (2.16)$$

Allora si può pesare ogni sweep per l'inverso della varianza dell'errore di stima:

$$\frac{1}{w_i} = E [||\hat{u}^i||^2] = \text{trace} [\text{cov}(\tilde{u}^i)] \quad (2.17)$$

In questo modo ogni sweep partecipa alla stima del segnale ERP medio proporzionalmente alla sua accuratezza di stima (più si discosta e più basso sarà il suo peso w_i).

Parametro di regolarizzazione γ

Il valore introdotto nella 2.14 viene chiamato parametro di regolarizzazione (o smoothing parameter), e può essere descritto come il rapporto tra la varianza del rumore d'ingresso al modello AR dell'EEG (σ^2 , nota) e la varianza incognita relativa al modello random walk dell'ERP (λ^2 , incognita), sempre in riferimento alla singola sweep. Considerando che \hat{u} nella 2.14 non è altro che la soluzione a

$$\text{argmin}_{\hat{u}} \{ (y - \hat{u})^T A^T A (y - \hat{u}) + \gamma \hat{u}^T F^T F \hat{u} \} \quad (2.18)$$

il nome che γ assume è esplicativo del suo ruolo. In tal senso esso costituisce un compromesso tra uno stimatore che privilegia l'aderenza alle misure e uno stimatore che invece dà maggior peso all'informazione a priori relativa all'irregolarità della stima. I casi limite sono infatti:

- $\gamma_i \rightarrow 0$: lo stimatore trascurava completamente l'informazione a priori considerando solo i dati misurati. Si ha in questo caso il fenomeno di overfitting poiché $\hat{u}^i \rightarrow y_i$;

- $\gamma_i \rightarrow \infty$: lo stimatore non prende affatto in considerazione le misure, ricorrendo solamente all'informazione a priori. Si ha in questo caso il fenomeno di oversmoothing poiché $\hat{u}^i \rightarrow 0$.

Criteri di regolarizzazione utilizzati

Essendo il parametro λ non noto, anche γ risulterà incognito, e dovrà essere stimato mediante un criterio di regolarizzazione (smoothing criterion). L'orientamento preso in questo lavoro di tesi è stato quello di adottare tre diverse metodologie, due delle quali a base statistica.

Il primo criterio utilizzato è quello di discrepanza introdotto da Twomey. Esso è un approccio di "buon senso" e non sfrutta alcuna proprietà statistica. Inizialmente prevede che, per ogni sweep, sia fissata la variabile aleatoria

$$WRSS(\gamma) = r(\gamma)^T A^T A r(\gamma) \quad (2.19)$$

dove $r(\gamma) = y - \hat{u}$ rappresenta il vettore dei residui della sweep considerata. Esso dipende evidentemente da γ in quanto è funzione di \hat{u} . $WRSS(\gamma)$ rappresenta di conseguenza la somma pesata dei residui al quadrato.

A questo punto il criterio suggerisce di scegliere il parametro di regolarizzazione iterativamente fino al soddisfacimento della seguente condizione:

$$WRSS(\gamma) \approx n\sigma^2 \quad (2.20)$$

con n numero di campioni in post-stimolo e σ^2 varianza del rumore d'ingresso al modello autoregressivo dell'EEG.

Se l'algoritmo di smoothing è tarato in modo adeguato ci si aspetta che (tralasciando per semplicità di trattazione la dipendenza da γ)

$$r^T r = E[v^T v] = \text{trace}(\Sigma_v) \quad (2.21)$$

Ovvero che

$$(y - \hat{u})^T B^{-1} (y - \hat{u}) = E[v^T B^{-1} v] = n\sigma^2 \quad (2.22)$$

dove è stato posto $B = (A^T A)^{-1}$.

Nel caso non parametrico il criterio si implementa sfruttando un algoritmo di bisezione particolarmente vantaggioso se si lavora su scala logaritmica⁷. Si scelgono γ_{min} e γ_{max} come valori limite (inferiore e superiore) “dentro” i quali si sa esserci il γ corretto e si itera finchè la norma del vettore dei residui normalizzati risulta circa uguale a $n\sigma^2$.

Infine, è di fondamentale importanza notare che la relazione 2.20 non ammette sempre soluzione. Nel caso in cui l’energia del segnale misurato risulti minore dell’energia del rumore, il primo termine della relazione non raggiungerà mai il secondo membro. Quando invece vale la condizione opposta, la soluzione esiste ed è unica. Un indice più oggettivo di γ circa il grado di regolarizzazione introdotto mediante la stima del segnale è fornito dal numero di gradi di libertà

$$q(\gamma) = trace \left\{ A (A^T A + \gamma F^T F)^{-1} A^T \right\} \quad (2.23)$$

Mentre infatti γ dipende da fattori quali la dimensione n del vettore stimato \hat{u} , il numero di gradi di libertà può essere normalizzato per il numero di campioni ($q(\gamma)/n$) ed assumere di conseguenza solo valori compresi tra 0 (massima regolarizzazione) e 1 (non viene effettuato smoothing).

Con il criterio di discrepanza però non si sfrutta appieno il vantaggio di lavorare in ambito stocastico. I vantaggi di operare in un contesto complesso come quello bayesiano sono innumerevoli, ad esempio:

- è possibile calcolare correttamente la matrice di covarianza dell’errore di stima;
- la statistica fornisce dei criteri per fissare γ ad un valore compatibile perchè sia il valore ottimo γ^0 .

Brevemente, introducendo l’errore di stima $\tilde{u} = u - \hat{u}$ e sfruttando il fatto che, in ambito bayesiano, tutti i vettori sono aleatori si può propagare la covarianza di u e v (nell’ipotesi che siano tra loro incorrelati) da cui si ottiene una formula non riportata per semplicità di trattazione. In funzione di γ , il valore minimo della traccia della matrice di covarianza dell’errore di stima si trova per il valore $\gamma^0 = \frac{\sigma^2}{\lambda^2}$ per il quale si ottiene l’equazione già espressa in 2.16.

⁷Ad ogni iterazione si pone $\gamma = 10^{((\log(\gamma_{min}) + \log(\gamma_{max}))/2)}$

Un valore generico di γ (ottenuto, ad esempio, con i criteri empirici/deterministici quali discrepanza) non consente di invocare propriamente l'espressione appena ottenuta per la matrice di covarianza dell'errore di stima.

Sfruttando ancora una volta l'ambito stocastico e la proprietà di consistenza degli stimatori si possono individuare due criteri di smoothing, detti appunto criteri di consistenza (nel caso Gaussiano il parametro di regolarizzazione trovato è anche di massima verosimiglianza, poichè massimizza la likelihood dei dati).

Il primo approccio utilizza la varianza del rumore EEG calcolata dal pre-stimolo e impone l'uguaglianza

$$WESS(\gamma) = \lambda^2 q(\gamma) \quad (2.24)$$

con $\lambda^2 = \frac{\sigma^2}{\gamma}$, mentre il secondo non adopera questo specifico valore e sfrutta la seguente relazione:

$$\frac{WRSS(\gamma)}{n - q(\gamma)} = \gamma \frac{WESS(\gamma)}{q(\gamma)} \quad (2.25)$$

A livello implementativo, l'approccio algoritmico utilizzato in entrambi i casi ricalca quello spiegato precedentemente per il criterio di Twomey.

2.3.2 Secondo step: stima single-trial

Nel secondo step, a differenza del primo, si dispone del vettore riferimento \bar{u} calcolato nella 2.15. Esso viene impiegato per portare a termine la stima single-trial vera e propria, denominata \check{u}^i per la sweep i -esima. I passaggi propri di questo secondo step ricalcano quelli effettuati nel primo, considerando \bar{u} come informazione a priori (o vettore di riferimento) circa l'ERP atteso, che costituisce a tutti gli effetti l'obiettivo finale della stima. Per chiarezza di notazione le stime ERPs parziali (a \bar{u} nullo) ottenute nel primo passo sono state indicate con \hat{u}^i , mentre quelle definitive che si andranno a calcolare nel presente second step sono indicate con \check{u}^i (per semplicità di notazione l'apice sarà successivamente tralasciato).

Vale dunque, per ogni sweep

$$\check{u} = \bar{u} + (A^T A + \xi L^T L)^{-1} A^T A (y - \bar{u}) \quad (2.26)$$

Come si può notare per confronto con la relazione 2.14, nella presente espressione la matrice A resta invariata (si ricorda che A è una matrice quadrata di Toeplitz, triangolare inferiore, di dimensione n , con n numero di campioni della sweep). Essa risulta uguale in entrambi i passi poiché la sottrazione del riferimento \bar{u} viene fatta solo sul post stimolo. Il pre-stimolo, rispetto a cui viene identificato il modello AR non risulta invece modificato.

La matrice F , al contrario, può risultare differente per la scelta di un numero diverso di integratori, ed è stata dunque sostituita da L . Si ricorda che il suo ruolo è quello di interpretare il modello adottato per descrivere la regolarità a priori del segnale ERP, assimilato ad un processo random walk a media nulla e varianza λ^2 pluri-integrato con numero di integratori che può variare da d (nel primo step) a q (nel secondo step). Mentre d indicava infatti lo smoothness dell'ERP di per sé, q determina lo smoothness della deviazione che presenta l'ERP single-trial dall'ERP medio \bar{u} . Il modello random walk del primo step presentato nell'equazione 2.9, assume nel secondo step la forma seguente:

$$u_k - \bar{u}_k = u_{k-1} - \bar{u}_{k-1} + w_k \quad (2.27)$$

Avendo cioè introdotto il vettore di riferimento, il modello non effettua più ipotesi circa la distanza di valore tra due campioni consecutivi del segnale ERP incognito, bensì pone a confronto la distanza calcolata tra il campione del segnale ERP incognito all'istante k -esimo e il campione del vettore di riferimento \bar{u} all'istante k -esimo, con la distanza calcolata tra il valore del segnale ERP incognito all'istante $(k-1)$ -esimo e il campione del vettore di riferimento all'istante $(k-1)$ -esimo, ipotizzando che tali distanze non siano poi così diverse tra loro. La matrice L risulterà pertanto pari a Δ^q . Concludendo, si può quindi affermare che, attorno a \bar{u} , il segnale u spazia secondo traiettorie descritte da modelli di rumore bianco plurintegrato.

Il parametro di regolarizzazione, infine, è stato riscritto come ξ e sarà calcolato ricorrendo ad un criterio di regolarizzazione come ad esempio quello di discrepanza. Il calcolo del valore ottimale del parametro di regolarizzazione prevede, ad ogni iterazione, un'operazione di inversione di matrice, notoriamente molto pesante dal punto di vista computazionale. Tuttavia, sfruttando la forma

di Toeplitz delle matrici in questione, ricorrendo ad alcuni artifici di algebra e adottando una strategia basata sulla Singular Value Decomposition [24] (si rimanda all'Appendice B per maggiori dettagli e eventuali chiarimenti), è possibile ridurre il numero di operazioni computazionalmente impegnative necessarie per giungere alla soluzione ottima.

2.4 Un approccio one-stage per la stima degli ERPs: MTL

Questo metodo, originariamente presentato in [25] e successivamente sviluppato in [26], è basato su una procedura multi-task learning ad un passo. In questo lavoro di tesi viene utilizzata una versione di questa tecnica riadattata e rivisitata in [27].

Diversamente dalla maggior parte degli approcci proposti in letteratura, il metodo fornisce delle stime della risposta media e degli ERPs single-trial considerando simultaneamente tutte le sweeps a disposizione. I segnali ERPs incogniti sono trattati come “individui” di una popolazione omogenea e l'informazione disponibile per un'epoca è considerata utile per la stima di tutte le altre. Il metodo assume che la generica sweep possa essere modellata come la somma di tre componenti stocastiche indipendenti: una curva media di popolazione che è comune a tutte le epoche, uno shift individuale che differenzia ogni epoca dalle altre, una componente di rumore EEG di fondo che varia da un'epoca ad un'altra.

Modello dei dati

L'ERP da stimare è dato dalla somma di due funzioni continue nel tempo: la componente media di popolazione $\bar{u}(t)$ e lo shift individuale $\tilde{u}(t)$. Il modello dei dati risulterà quindi

$$y_k^i = \bar{u}(t_k) + \tilde{u}^i(t_k) + v^i(t_k) \quad (2.28)$$

dove i rappresenta la sweep i -esima e k rappresenta l'istante temporale il cui la misura y_k^i è disponibile. Vengono mantenuti i vincoli impostati nel paragrafo precedente: il numero delle sweep è uguale a N mentre quello dei campioni a n .

I processi stocastici $\bar{u}(t)$, $\tilde{u}(t)$ e $v(t)$ sono inoltre assunti indipendenti tra loro.

Considerando simultaneamente tutti i dati contenuti nelle N epoche, il modello⁸ può essere descritto in forma vettoriale come

$$y = \bar{u} + \tilde{u} + v \quad (2.29)$$

dove, nel caso di campionamento standardizzato⁹

$$y := [y_1^1 \dots y_n^1, y_1^2 \dots y_n^2, \dots, y_1^N \dots y_n^N] \quad (2.30)$$

$$\bar{u} := [\bar{u}(t_1) \dots \bar{u}(t_n), \bar{u}(t_1) \dots \bar{u}(t_n), \dots, \bar{u}(t_1) \dots \bar{u}(t_n)] \quad (2.31)$$

$$\tilde{u} := [\tilde{u}^1(t_1) \dots \tilde{u}^1(t_n), \tilde{u}^2(t_1) \dots \tilde{u}^2(t_n), \dots, \tilde{u}^N(t_1) \dots \tilde{u}^N(t_n)] \quad (2.32)$$

$$v := [v^1(t_1) \dots v^1(t_n), v^2(t_1) \dots v^2(t_n), \dots, v^N(t_1) \dots v^N(t_n)] \quad (2.33)$$

Modello del rumore EEG

Il modello del rumore adottato corrisponde con quello già presentato nell'approccio two-stage.

Ricordando l'indipendenza tra le differenti epoche dei vettori contenenti il rumore EEG, si può derivare la seguente espressione per la matrice di covarianza:

$$\Sigma_v = \text{blockdiag} \{ \Sigma_v^1, \dots, \Sigma_v^N \} \quad (2.34)$$

⁸Il modello riscritto in forma compatta ha dimensione $n_{TOT} = N \times n$.

⁹Per campionamento standardizzato si intende che ogni trial contiene lo stesso numero di campioni.

Modello dell'ERP

In accordo con lo schema proposto in [25] vengono sviluppati dei processi di Wiener integrati in modo da evidenziare le caratteristiche di regolarità della curva media e dello shift individuale. In particolare un processo di questo tipo può essere descritto dal seguente modello con una rappresentazione in spazio di stato:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bw(t) \\ z(t) = Cx(t) \end{cases} \quad (2.35)$$

dove $x(0) \sim N(0, X_0)$ e $w(t) \sim WGN(0, \lambda^2)$ indipendente da $x(0)$.

Le matrici in gioco sono $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$; $B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$; $C = \begin{bmatrix} 1 & 0 \end{bmatrix}$.

Il parametro λ^2 guida il modello ed è direttamente associato alla regolarità delle realizzazioni. Valori piccoli corrispondono a segnali particolarmente "smooth". In particolare, si assume che gli elementi di $\bar{u}(t)$ siano estratti da un processo di Wiener integrato a media nulla, guidato da un processo di rumore bianco gaussiano con varianza $\bar{\lambda}^2$ non nota. I campioni dello shift individuale $\tilde{u}(t)$ sono modellati allo stesso modo con la differenza che la varianza a cui riferirsi, in questo caso, è $\tilde{\lambda}^2$.

Introduciamo le matrici

$$\bar{R}(t, \tau) := cov [\bar{u}(t), \bar{u}(\tau)] \quad (2.36)$$

$$\tilde{R}(t, \tau) := cov [\tilde{u}^i(t), \tilde{u}^i(\tau)] \quad (2.37)$$

Nel caso di segnali le cui condizioni iniziali possano essere considerate deterministicamente note¹⁰ le funzioni di autocovarianza dei processi assumono le seguenti espressioni:

¹⁰Nel caso particolare di stima bayesiana degli ERPs i segnali considerati sono assunti a valore nullo nell'istante temporale $t=0$.

$$\bar{R}(t, \tau) = \bar{\lambda}^2 \begin{cases} \frac{t^2}{2}(\tau - \frac{t}{3}), & t \leq \tau \\ \frac{\tau^2}{2}(t - \frac{\tau}{3}), & t > \tau \end{cases} \quad (2.38)$$

$$\tilde{R}(t, \tau) = \tilde{\lambda}^2 \begin{cases} \frac{t^2}{2}(\tau - \frac{t}{3}), & t \leq \tau \\ \frac{\tau^2}{2}(t - \frac{\tau}{3}), & t > \tau \end{cases} \quad (2.39)$$

A questo punto si possono ottenere le matrici di covarianza

$$\Sigma_{\bar{u}} = \begin{bmatrix} \bar{R} & \dots & \bar{R} \\ \vdots & \ddots & \vdots \\ \bar{R} & \dots & \bar{R} \end{bmatrix} \quad (2.40)$$

$$\Sigma_{\tilde{u}} = \text{blockdiag} \{ \tilde{R}, \dots, \tilde{R} \} \quad (2.41)$$

dove

$$\bar{R} = \begin{bmatrix} \bar{R}(t_1, t_1) & \dots & \bar{R}(t_1, t_n) \\ \vdots & \ddots & \vdots \\ \bar{R}(t_n, t_1) & \dots & \bar{R}(t_n, t_n) \end{bmatrix} \quad (2.42)$$

$$\tilde{R} = \begin{bmatrix} \tilde{R}(t_1, t_1) & \dots & \tilde{R}(t_1, t_n) \\ \vdots & \ddots & \vdots \\ \tilde{R}(t_n, t_1) & \dots & \tilde{R}(t_n, t_n) \end{bmatrix} \quad (2.43)$$

La matrice $\Sigma_{\tilde{u}}$ è block diagonal poiché gli scostamenti individuali sono assunti indipendenti tra loro. Il valore atteso di \tilde{u}^i può essere assunto nullo in quanto rappresenta lo shift rispetto alla curva media, la stessa ipotesi può essere fatta sul valore atteso di \bar{u} se i dati sono stati opportunamente scalati sottraendone la media.

Si noti che, modellando gli shift individuali tramite processi di Wiener integrati, la varianza tende ad infinito con t (espressioni 2.42 e 2.43). Per risolvere questo problema, si può aggiungere un'informazione a priori sul valore asintotico della

curva per t che tende ad infinito utilizzando un opportuno cambio di coordinate oppure trattando l'ultimo campione come deterministicamente noto.

2.4.1 Stima della curva media e degli shift individuali

Lo stimatore può essere espresso in forma chiusa se i processi coinvolti sono congiuntamente gaussiani. La stima bayesiana a minima varianza d'errore risulta

$$\hat{u} = E[\bar{u}|y] = \Sigma_{\bar{u}}(\Sigma_y)^{-1}y = \Sigma_{\bar{u}}(\Sigma_{\bar{u}} + \Sigma_{\tilde{u}} + \Sigma_v)^{-1}y \quad (2.44)$$

$$\hat{\tilde{u}} = E[\tilde{u}|y] = \Sigma_{\tilde{u}}(\Sigma_y)^{-1}y = \Sigma_{\tilde{u}}(\Sigma_{\tilde{u}} + \Sigma_{\bar{u}} + \Sigma_v)^{-1}y \quad (2.45)$$

Le matrici di covarianza coinvolte sono le stesse ottenute nella sezione precedente e dipendono unicamente da i parametri di regolarizzazione $\bar{\lambda}^2$ e $\tilde{\lambda}^2$ i quali vengono stimati tramite Maximum Likelihood.

Calcolo della funzione di verosimiglianza

Gli iper-parametri $\bar{\lambda}^2$ e $\tilde{\lambda}^2$ determinano, rispettivamente, quanto i segnali incogniti \bar{u} e \tilde{u}^i siano regolari. Per la stima di questi parametri, si applica il cosiddetto metodo "bayesiano empirico" basato sulla stima Maximum Likelihood. Si cercano, quindi, i valori del vettore parametrico $\theta = [\bar{\lambda}^2, \tilde{\lambda}^2]^T$ che massimizzano la funzione di verosimiglianza.

Ciò equivale a determinare

$$\operatorname{argmin}_{\theta} J(y, \theta) = \operatorname{argmin}_{\theta} \{y^T \Sigma_y^{-1}y + \log[\det(\Sigma_y)]\} \quad (2.46)$$

Si noti che il calcolo della funzione costo $J(y, \theta)$ richiede l'inversione e il computo del determinante di una matrice di dimensioni n_{TOT} . Dal momento che il minimo di $J(y, \theta)$ viene valutato iterando il calcolo della funzione per diversi valori di θ , la complessità diventa notevole nel caso in cui n_{TOT} sia molto grande (ad esempio, se si hanno a disposizione $N = 20$ sweeps, campionate ad 1 secondo con frequenza di campionamento $fs = 512Hz$, risulterà $n_{TOT} = 10240$). Per diminuire notevolmente l'onere computazionale si utilizza un algoritmo ad hoc,

particolarmente efficiente, e viene data all'utente la possibilità di impostare un fattore di downsampling, soprattutto nel caso di campionamento molto fitto.

Tornando al calcolo della funzione di verosimiglianza, per quanto riguarda il secondo termine della funzione costo, il determinante è calcolabile sfruttando il seguente algoritmo:

$$\det(\Sigma_y) = \left(\prod_{i=1}^m \det(C_{ii}) \right)^2 \quad (2.47)$$

dove le varie C_{ii} sono matrici quadrate di dimensione n costruite in modo che

$$C_{ii} = \text{chol}[A_{ii} - D_i] \quad (2.48)$$

$$C_{(i+1)i} = (\bar{R} - D_i)(C_{ii}^T)^{-1} \quad (2.49)$$

$$C_{ki} = C_{(i+1)i} \quad \text{per } k > i + 1 \quad (2.50)$$

$$D_{i+1} = D_i + C_{(i+1)i} C_{(i+1)i}^T \quad (2.51)$$

dove la formula *chol* denota la fattorizzazione inferiore di Choelsky, inoltre risulta

$$A_{ii} = \bar{R} + \tilde{R} + \Sigma_v^i \quad (2.52)$$

Le condizioni iniziali per l'algoritmo sono

$$\begin{cases} D_1 = 0 \\ C_{21} = \bar{R}(C_{11}^T)^{-1} \end{cases} \quad (2.53)$$

Si ricercano quindi i valori del vettore θ che rendono minima la funzione costo $J(y, \theta)$, essi verranno poi sostituiti in 2.38 e 2.39 in modo da ottenere le matrici di autocovarianza della curva media e degli scostamenti individuali.

Da un punto di vista implementativo, il problema di minimizzazione si risolve sfruttando la funzione *fminsearch.m* di MATLAB[®], che utilizza il Nelder–Mead simplex method [2].

Casi limite

Gli iper-parametri possono presentare una vasta gamma di valori. Esistono tuttavia dei casi limite particolarmente significativi:

- se $\tilde{\lambda}^2 \ll \bar{\lambda}^2$ tutti gli ERPs single-trial risultano “molto simili” tra loro, in particolare se $\tilde{\lambda}^2 \rightarrow 0$ tutte le stime collassano al valore dell’ERP medio stimato;
- se $\tilde{\lambda}^2 \gg \bar{\lambda}^2$ gli ERPs single-trial non mostrano componenti comuni tra loro, in particolare se $\bar{\lambda}^2 \rightarrow 0$ l’ERP di popolazione si riduce ad una curva piatta, costante nel tempo e identicamente uguale a 0.

Il secondo caso può talvolta presentarsi come punto critico dell’algoritmo soprattutto quando si utilizza un fattore di downsampling molto alto. Si può ovviare a questo problema operando una strategia di scelta del passo di sottocampionamento particolarmente intelligente: si ricerca il valore minimo che permetta di avere più dati possibile e di effettuare la stima in tempi abbastanza ridotti. D’altro canto, gli effetti indesiderati si verificano unicamente nella stima della curva media, infatti $\tilde{\lambda}^2$ si “adatterà” al valore basso di $\bar{\lambda}^2$ e le stime single-sweep risulteranno comunque accettabili. In ogni caso, per evitare questo risultato particolarmente problematico, si può procedere alla forzatura del parametro $\bar{\lambda}^2$.

2.5 Note sui metodi bayesiani

I due approcci bayesiani sono stati confrontati tra di loro in [2]. Ovviamente, si può ottenere una valutazione “quantitativa” delle performance di un metodo rispetto all’altro solo lavorando con dati simulati. L’indice adottato per quantificare l’accuratezza di un algoritmo di stima è il seguente¹¹:

$$E = 100 \times \frac{\|u_{true} - \bar{u}\|^2}{\|u_{true}\|^2} \quad (2.54)$$

E’ fin troppo evidente l’impossibilità di usare questa espressione in un contesto “real-data” poichè non si ha a disposizione u_{true} ed esso non è calcolabile.

¹¹L’indice espresso nella 2.54 si riferisce unicamente alla stima dell’ERP medio e non all’analisi single-sweep.

Se il numero dei trial è ridotto, la tecnica CA (Conventional Averaging) offre delle stime piuttosto rumorose. Al contrario, B2S (Bayesian Two Steps) e MTL (Multi Task Learning), con lo stesso numero di epoche, calcolano un \bar{u} molto simile a u_{true} (come si può osservare in figura 2.1), non mostrando delle differenze significative a livello di performance.

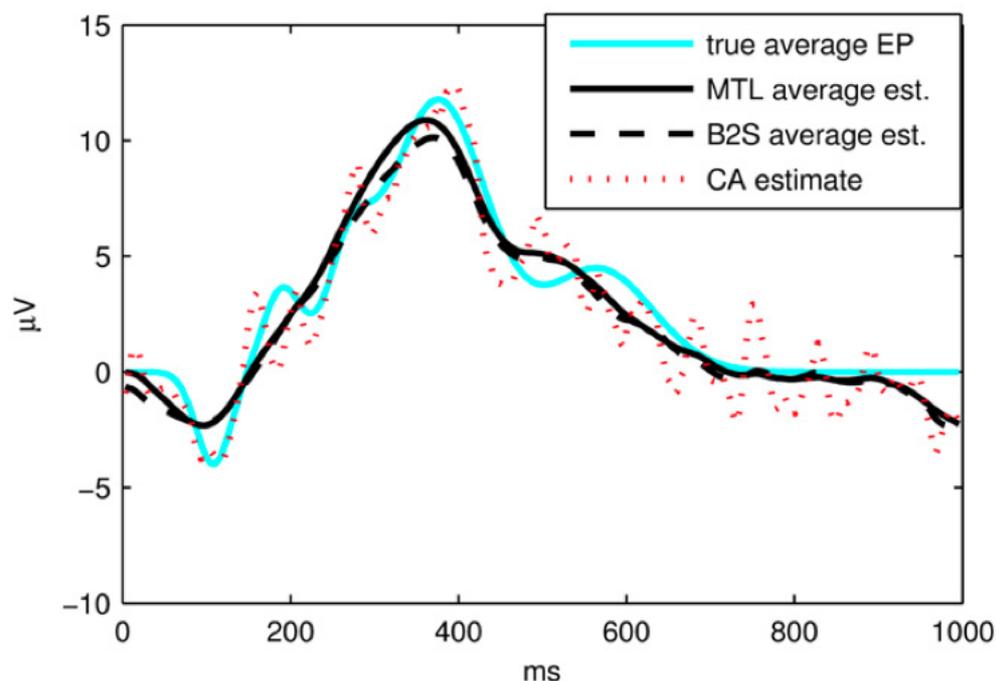


Figura 2.1: *Confronti tra stime medie.*

Tuttavia, non si può ripetere tale affermazione nel contesto di analisi single-trial:

- in molti casi i valori di latenza e ampiezza stimati dall’algoritmo B2S risultano “peggiori” rispetto a quelli stimati da MTL;
- se la raw-sweep è particolarmente rumorosa la stima a due passi tende a “seguire i dati” mentre quella ad un passo è decisamente più accurata (figura 2.2);
- gli indici di errore relativi al metodo di Multi Task Learning sono, in media, minori rispetto a quelli valutati con Bayesian Two Steps;

- l'analisi quantitativa espressa nel punto precedente viene ulteriormente rafforzata dal test di Wilcoxon non parametrico, il quale evidenzia delle divergenze statisticamente significative tra gli indici di errore relativi ai due approcci algoritmici.

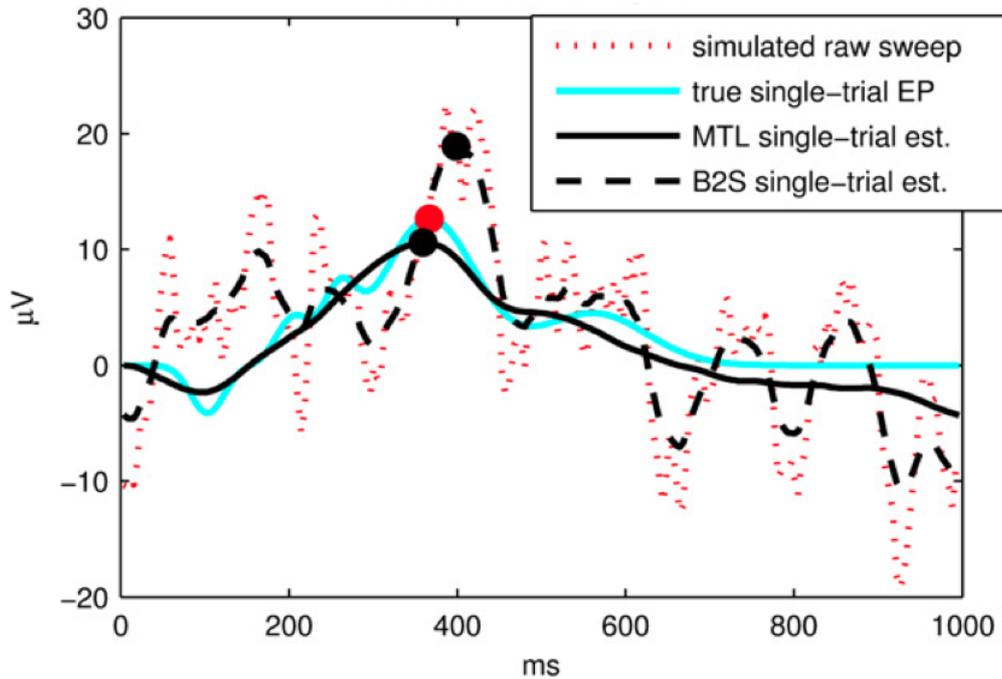


Figura 2.2: *Confronti tra stime single-trial.*

Come detto precedentemente, non si può eseguire un test in un contesto “real-data”, ci si aspetta tuttavia che le stime MTL riflettano maggiormente la variabilità intra-individuale¹² della componente P300 rispetto a quelle B2S.

2.6 Scopo della tesi

Gli obiettivi principali della tesi sono:

- sviluppare delle metodologie sofisticate per lo studio degli ERPs che prevedano l’implementazione degli algoritmi di stima bayesiana appena descritti seguiti da una fase di peak detection (capitolo successivo);

¹²Variabilità a carico dello stesso individuo. Definizione in contrasto con quella di variabilità inter-individuale (fra diversi individui).

422. DUE RECENTI METODI BAYESIANI DI STIMA SINGLE-TRIAL: B2S E MTL

- sviluppare un software user-friendly in grado di rilevare i parametri ERP di interesse (latenze e ampiezze della componente P300);
- integrare il tool sviluppato all'interno di un software pienamente riconosciuto in ambito clinico e neurofisiologico: l'ambiente EEGlab.

Capitolo 3

Metodi di peak-detection per l'analisi di risposte single-trial

L'ambito della peak-detection (ricerca del picco) impatta su un contesto di conoscenze piuttosto eterogeneo. Sono chiamati in causa metodi tra loro molto diversi: l'occhio clinico, il filtraggio in frequenza, le basi di template matching, la teoria della verosimiglianza e numerose altre tecniche di analisi. Lo scopo dei metodi di peak-detection è quello di rilevare una o più deflessioni di interesse a partire da una preparazione preliminare del segnale a disposizione [28]. La tecnica convenzionale di ricerca del picco nel campo dell'analisi ERPs prevede che, al fine di ottenere una misura rappresentativa, i parametri non siano misurati sulle singole sweep e poi mediati, bensì misurati una volta sola sulla sweep media ottenuta mediante averaging convenzionale. Questo per il fatto che la performance dei metodi proposti è strettamente dipendente dal SNR, il quale risulta critico considerando che gli ERPs sono immersi nel segnale EEG di fondo avente peraltro ampiezza superiore [29]. Tuttavia il passo di averaging comporta, come visto in precedenza, anche numerosi svantaggi. Si è adottato in tal senso il sopra descritto metodo di stima bayesiana, seguito da una fase di post-filtraggio passa-basso, a conclusione dei quali l'SNR si presenta radicalmente migliorato a livello delle singole sweep. Tale fattore incide profondamente sulla scelta del metodo ottimo, il quale andrà dunque ad essere applicato non su una traccia media, bensì sulle stime single-trial.

In questo capitolo vengono presentati i vari metodi di ricerca del picco da un

punto di vista puramente descrittivo. Per quanto riguarda gli aspetti implementativi si rimanda ad un lavoro di tesi contestuale e sinergico [30] ove tutte le scelte algoritmiche sono illustrate esaustivamente.

3.1 Visual inspection

Il nome esprime chiaramente l'approccio adottato in questo metodo: l'ispezione visiva. Già dal 1973 questo metodo è stato apostrofato da Donchin e collaboratori come inadeguato nel caso in cui costituisca l'unica via di ispezione condotta [8]. Il suo punto critico è infatti la mancanza di oggettività, che si aggiunge alla necessità di avere a disposizione un operatore clinico in grado di saper apprezzare semplicemente "a colpo occhio" la componente ERP di interesse estraendone i parametri. La scelta del picco si rivela talvolta ardua coinvolgendo da un lato l'esperienza del clinico e dall'altro la sua valutazione soggettiva.

3.2 Peak-picking

Il funzionamento del metodo di peak-picking [31] è chiaro e diretto: esso è basato sull'osservazione che, dopo un primo step di regolarizzazione del segnale, la componente P300 sia identificabile con immediatezza in un numero significativo di trial come picco di ampiezza evidentemente superiore entro un fissato intervallo di interesse. Più in dettaglio, il segnale grezzo viene inizialmente ripulito mediante un filtraggio passa-basso o per mezzo di un'altra tecnica (nel presente lavoro si è adottato il filtraggio bayesiano seguito da un'ulteriore step di LPF), dopodiché si ricerca la massima deflessione positiva all'interno di una predefinita finestra dove ci si aspetta che la P300 si manifesti. Un tipico valore del range di ricerca è di circa 250-700 ms [29][32]. La latenza è dunque identificata come intervallo temporale che intercorre tra l'istante zero (di presentazione dello stimolo) e la comparsa del punto di massimo del picco.

Una semplice e accurata identificazione è chiaramente possibile solo se la prima fase di preparazione del segnale risulta efficiente nell'isolare quanto più possibile l'ERP desiderato dall'attività EEG di background. Il presente metodo, andando

ad identificare quale picco della componente di interesse il valore massimo entro un intervallo fissato, risulta particolarmente sensibile alle irregolarità del segnale. Lo smoothing ottenuto mediante il filtraggio bayesiano gioca pertanto un ruolo fondamentale nel dare credibilità ai risultati ottenuti.

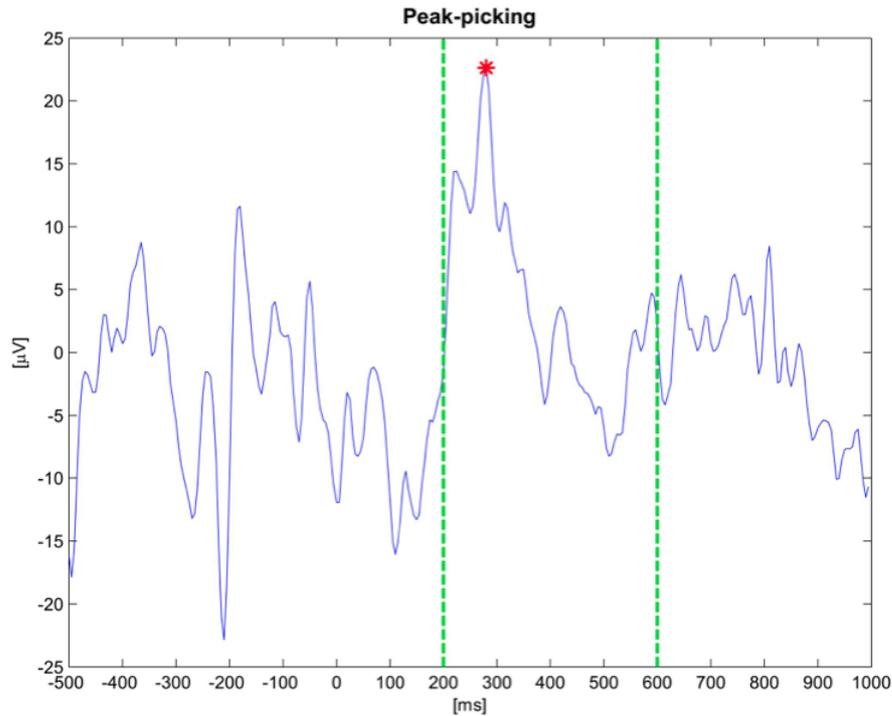


Figura 3.1: Tracciato EEG su cui è applicata la tecnica peak-picking entro la finestra di ricerca 200-600 ms. Il picco individuato dall'asterisco rosso coincide semplicemente con il massimo valore assunto dal segnale all'interno dell'intervallo temporale fissato.

3.3 Woody's method

Il metodo di Woody prevede il calcolo della cross-correlazione (detta anche correlazione incrociata), oppure della cross-covarianza (detta anche covarianza incrociata) tra un template che ricalca la forma della componente di interesse e la singola traccia registrata ripulita mediante una fase di pre-filtraggio [33]. Per template si intende la forma d'onda che funge da sagoma, da sonda, utile per effettuare la scansione della single-sweep al fine di localizzare profili simili a quello

di interesse (rappresentato dal template stesso) [34]. Esso è ricavato solitamente dall'average delle tracce disponibili, ma sono stati utilizzati in letteratura anche segnali standard quali metà di una sinusoida, un'onda triangolare o una porzione di parabola con la concavità concorde a quella della componente [29]. La latenza single-trial è definita come il lasso di tempo del quale deve essere shiftato il template per ottenere il massimo valore di cross-correlazione o cross-covarianza.

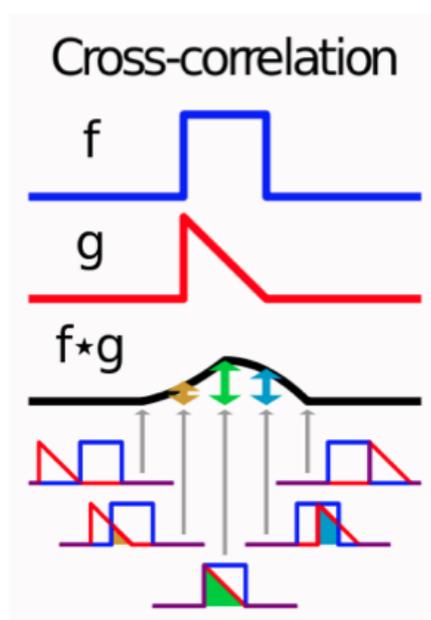


Figura 3.2: *Illustrazione esplicativa del concetto di cross-correlazione. L'onda quadrata f in blu può essere paragonata al segnale ERP su cui il template rosso g scorre a generare il segnale di correlazione $f*g$. Su quest'ultimo viene poi individuato il valore di massimo, che andrà riportato al segnale originale in modo da identificare latenza ed ampiezza del picco ricercato.*

Smulders e collaboratori hanno approfondito la differenza presente tra le due possibili operazioni di confronto template-sweep, concludendo che la cross-covarianza sia da preferire. Mentre infatti la cross-correlazione considera solo la fisionomia della forma d'onda, la cross-covarianza si basa non solo sulla ricerca della porzione di sweep avente la fisionomia più simile al template, ma tiene presente anche del fattore ampiezza [35].

Il modello del segnale è:

$$x_i(n) = s(n - d_i) + e_i(n) \quad (3.1)$$

dove $i = 1, \dots, N$ è il numero delle sweep, n tale che $0 < n < N$ è il campione temporale osservato, $x_i(n)$ è l'ampiezza della sweep i -esima all'istante n , $s(n - d_i)$ rappresenta una certa onda-template shiftata di d_i , $e_i(n)$ costituisce infine l'errore dovuto al rumore. L'obiettivo consiste nello stimare lo shift d_i tipico della singola sweep, in grado di allinearla al template massimizzandone la cross-covarianza (o cross correlazione). La relazione che permette di eseguire questa stima è:

$$\hat{d}_i = \operatorname{argmax}_{d_i} \frac{1}{N} \sum_{n=1}^N x_i(n) \hat{s}(n - d_i) \quad (3.2)$$

dove $\hat{s}(n)$ è una stima iniziale del template (ottenuta tramite average o assimilata ad un segnale standard), rispettivamente shiftata di d_i .

Talvolta il filtro viene reso "adattativo" [36] la ricerca del massimo della cross-covarianza (o in alternativa della cross-correlazione) viene ripetuta ricavando iterativamente un nuovo template dai single-trial che vengono progressivamente allineati in latenza. Questo passo si rende necessario poiché spesso il fenomeno di jitter in latenza rende l'average considerevolmente distorto e dunque inutilizzabile come template. Nella fattispecie il template iniziale consiste in un segnale standard rispetto al quale vengono calcolate le correlazioni di ogni singola epoca. Queste ultime vengono quindi allineate tra loro e con il template sulla base del massimo valore di cross-covarianza ottenuto. Si calcola dunque un nuovo template ottenuto come average tra le singole sweep shiftate. Si ripete il processo iterativamente finché il valore di cross-covarianza diviene stabile o l'incremento da uno step al successivo non è sensibile. Diversi studi supportano l'utilizzo del metodo di peak-picking mentre altri quello di Woody [29][35]. E' stato comunque dimostrato che in entrambi i casi il rapporto segnale-rumore riveste un ruolo cruciale, e che lo smoothing riduce le differenze di efficienza tra i due metodi.

3.4 Curve fitting

La tecnica di curve fitting, o fitting polinomiale, si basa su un approccio matematico che mira ad approssimare il segnale grezzo con una curva di grado n nel modo più aderente possibile [37]. Nella fattispecie la curva considerata è spesso una parabola, ed il polinomio in questione è dunque di secondo grado. I parametri di tale luogo geometrico vengono calcolati mediante un procedimento di stima in modo tale da ottimizzare l'approssimazione dei campioni nei dintorni del picco del quale si vogliono calcolare ampiezza e latenza. La stima dei coefficienti del polinomio è ottenuta con il metodo dei minimi quadrati, il quale mira a minimizzare la somma quadratica dei residui, dove per residuo si intende la differenza tra il valore osservato y e il valore stimato \hat{y} ad un certo istante i . Pertanto, dato che $r_i = y_i - \hat{y}_i$ il residuo può essere interpretato come l'errore di stima riferito al particolare istante i . La somma quadratica dei residui risulterà

$$RSS = \sum_{i=1}^n r_i^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (3.3)$$

con n pari al numero di campioni considerati nel processo di fitting, ed RSS obiettivo della minimizzazione. La curva viene in questo modo costruita in modo da ottenere il best-fit. Nel caso particolare di polinomio di grado due, i coefficienti da stimare saranno a , b e c della relazione $\hat{y}_i = ax_i^2 + bx_i + c$, con $i = 1, \dots, n$. In notazione matriciale si ottiene

$$\hat{y} = Xh \quad (3.4)$$

Avendo posto:

$$\hat{y} = \begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_n \end{pmatrix}; \quad X = \begin{pmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ x_n^2 & x_n & 1 \end{pmatrix}; \quad h = \begin{pmatrix} a \\ b \\ c \end{pmatrix} \quad (3.5)$$

La stima ai minimi quadrati del vettore h , ovvero dei coefficienti del polinomio di secondo grado è data da

$$\hat{h} = (X^T X)^{-1} X^T y \quad (3.6)$$

Ottenuta la parabola ottima, il problema di localizzazione del picco della componente si riduce alla ricerca del vertice della parabola stessa. Dalla geometria analitica si ottengono:

- ascissa del vertice=latenza del picco= $-\frac{\hat{b}}{2\hat{a}}$;
- ordinata del vertice=ampiezza del picco= $\hat{c} - \frac{\hat{b}^2}{4\hat{a}}$.

In altri termini lo scopo di questo metodo è quello di giungere ad una regolarizzazione della curva in modo tale da rendere banale l'identificazione di latenza e ampiezza del picco di interesse.

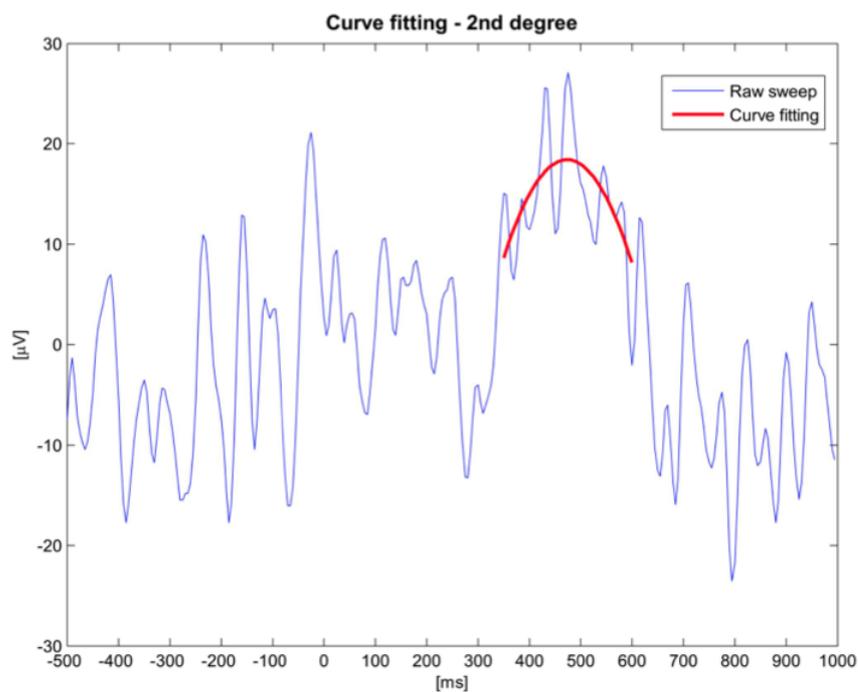


Figura 3.3: *Approssimazione del tracciato EEG mediante una parabola (fitting polinomiale di ordine 2).*

3.5 Fractional area

Introdotta da Hansen e collaboratori [38], questo metodo si fonda sullo studio dell'area sottesa alla componente di interesse (AUC, Area Under the Curve). In particolare la tecnica consiste nel definire come latenza della componente il primo istante temporale in cui una determinata percentuale dell'AUC è stata raggiunta. E' inoltre possibile introdurre un vincolo sull'ampiezza, come mostrato in figura 3.4: l'area sottesa alla componente negativa N1 è identificata come l'AUC sotto la soglia di $-2\mu V$ nell'intervallo temporale compreso tra 0 e 150 ms dalla presentazione dello stimolo. Definita l'area di interesse, la latenza di N1 viene determinata come punto temporale in cui tale area raggiunge il 50% del totale [39].

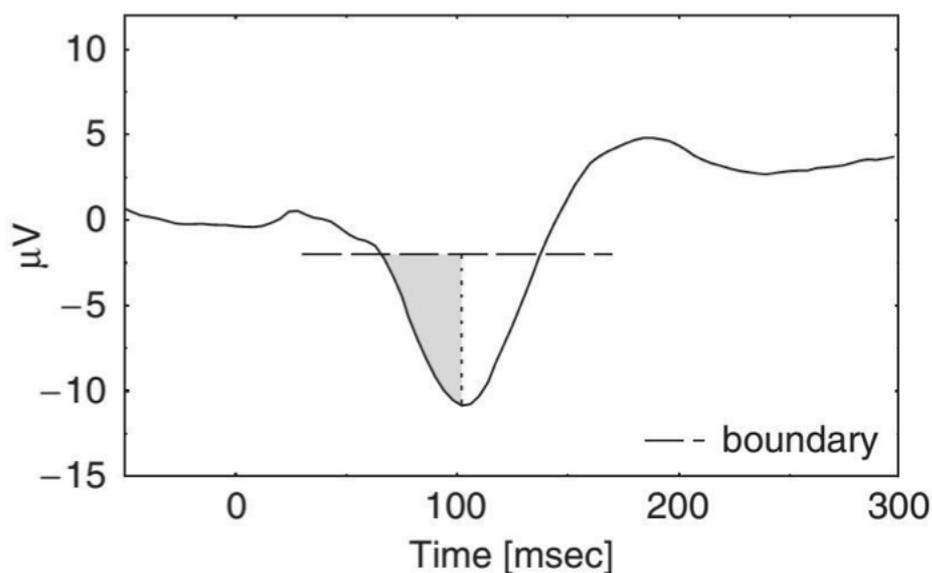


Figura 3.4: *Illustrazione della tecnica di fractional area nel particolare caso della misura di latenza della componente N1. Essa viene definita come istante temporale in cui viene raggiunto il 50% dell'area totale, in accordo con i limiti temporali e di ampiezza fissati.*

E' dunque necessario fissare due parametri decisivi per l'efficacia del metodo: la percentuale di area cui si fa corrispondere la misura di latenza, e i limiti temporali e di ampiezza entro cui effettuare il calcolo integrale dell'area stessa. La precisa caratterizzazione delle proporzioni di AUC di ogni componente ERP e

degli intervalli da fissare dipendono da numerosi fattori, e vanno peraltro caratterizzati empiricamente (ad esempio mediante l'uso di simulazioni) dato che in letteratura non esistono indicazioni riconosciute ed affermate in merito [40]. Nel caso della P300, Luck ha definito la latenza come l'istante temporale in cui il 50% dell'area sottesa alla componente viene raggiunto, avendo fissato come intervallo temporale di calcolo il range da 300 a 800 ms [39][32].

3.6 ML method

Il metodo Maximum Likelihood (ML) è stato introdotto da Pham e collaboratori [41]; esso si basa sulle assunzioni che l'attività registrata sia composta dal segnale desiderato e dal rumore di fondo, e che l'unico parametro variabile da sweep a sweep sia la latenza, mentre fisionomia ed ampiezza sono ipotizzate costanti. L'espressione matematica di tale modello è

$$r_i(t) = s(t + \tau_i) + e_i(t) \quad (3.7)$$

dove t , variabile da 1 a T , indica l'istante temporale della forma d'onda analizzata, $r_i(t)$ è il segnale registrato al tempo t , $s(t)$ è il segnale utile (potenziale evento correlato), $e_i(t)$ rappresenta il rumore, e τ_i indica lo shift in latenza tipico del trial i -esimo.

E' possibile notare con immediatezza che l'ERP di per sé non presenta pedice caratteristico della singola epoca: l'unico parametro caratterizzante è lo shift in latenza dato da τ_i . Il segnale ERP, secondo questa ipotesi, trasla esclusivamente nel tempo mantenendo invariata la sua forma trial dopo trial. Diversamente dal metodo di Woody precedentemente descritto, la tecnica ML riconosce esplicitamente la presenza del rumore. Il principale discriminante tra le due tecniche è costituito dalla differenza nell'approccio: mentre il Woody's method si pone come obiettivo quello di ricercare il massimo di somiglianza tra segnale e template ignorando il rumore di fondo, il metodo ML cerca invece di massimizzare la separazione tra segnale e rumore sfruttandone le differenze. Tale separazione viene effettuata nel dominio della frequenza, e dunque la citata equazione 3.7 viene trasformata secondo Fourier in

$$R_i(\omega) = S(\omega)\exp(-j\omega\tau_i) + E_i(\omega) \quad (3.8)$$

dove $\omega = 0, \frac{2\pi}{T}, \frac{4\pi}{T}, \dots$ è pari alle singole frequenze della forma d'onda analizzata, e le lettere maiuscole identificano le trasformate di Fourier delle grandezze precedentemente espresse nel dominio del tempo. Rispetto a tale modello può essere formulata una likelihood function (funzione di verosimiglianza), dalla quale è poi possibile ricavare i parametri desiderati mediante massimizzazione della derivata della log likelihood function per mezzo del calcolo del punteggio di Fisher iterativo. Il risultato della ricerca converge piuttosto raramente, ed è pertanto necessario fissare empiricamente un numero ottimo di iterazioni.

3.7 EML method

Il metodo Enhanced Maximum Likelihood (EML) consiste in una generalizzazione del metodo ML formulata da Jaskowski e Verleger [42]. La massimizzazione della funzione di log likelihood viene infatti condotta su un modello che contempla per l'ERP, oltre allo shift in frequenza, un ulteriore grado di libertà che va a caratterizzare ogni sweep: la variazione in ampiezza. La formulazione diventa quindi

$$r_i(t) = a_i s(t + \tau_i) + e_i(t) \quad (3.9)$$

dove a_i è pari all'ampiezza tipica della sweep i -esima.

La presente generalizzazione del modello precedente rende il metodo EML più aderente alla realtà: si pensi ad esempio a come l'ampiezza della componente P300 sia fortemente influenzata non solo dal tipo di stimolo, ma anche da variabili intra-task quali gli intervalli tra stimoli target successivi e il fattore attentivo [29].

3.8 Discussioni e conclusioni

Alla luce del particolare contesto di analisi, si è presentata la necessità di effettuare con cognizione di causa una scelta tra metodi sopra descritti.

Trattandosi di un tool automatico, l'opzione della visual inspection è stata esclusa a priori poiché, come detto, richiede non solo la supervisione ma il diretto intervento di un operatore clinico.

Per quanto riguarda invece i metodi di Woody, del fitting polinomiale, della fractional area ed EML, sono stati ritenuti non ottimi per la presente applicazione in quanto interpretano il problema da un punto di vista non appropriato (i primi tre sono stati comunque inseriti nel tool). Di fatto il Woody's method presuppone la disponibilità di un template adeguato ad interpretare l'operazione di correlazione; il curve fitting si candida come alternativa in grado di raggiungere una degna approssimazione nel caso di segnali rumorosi e frastagliati (tuttavia si dispone, in questo caso, di segnali regolari e smussati in output dalla stima bayesiana), ma richiede di impostare un intervallo quanto più possibile ristretto per ottimizzare il fit; l'approccio fractional area manca di proporzioni e direttive certi e testati in letteratura nel caso della P300; infine l'EML method non è stato preso in considerazione per la sua caratteristica di intervenire su segnali irregolari che subiscono un processo di smoothing preliminare contestualmente al metodo stesso.

Da un'analisi delle stime bayesiane effettuate sui tracciati EEG a disposizione si è potuto notare come, semplicemente fissando un intervallo temporale ottimo ed effettuando una ricerca del massimo al suo interno secondo l'approccio peak-picking, le stime di ampiezza e latenza della P300 fossero affidabili. A corroborare la validità della via intrapresa interviene la seguente conclusione di Jaskowski e Verleger [29]: i vantaggi del metodo EML sono tangibili in particolar modo quando il rapporto segnale rumore è elevato; nel caso in cui la sweep sia regolare i metodi citati perseguono quasi perfettamente il loro obiettivo.

Capitolo 4

Progettazione della GUI e integrazione in EEGlab

Un'interfaccia grafica consente all'utente di interagire con il computer manipolando graficamente degli oggetti in maniera molto user-friendly, al contrario di quanto avviene sulla cosiddetta riga di comando di una CLI (command line interface), in cui l'esecuzione del programma viene guidata da istruzioni o comandi impartiti dall'utente tramite tastiera. La Graphical User Interface (GUI) sviluppata in questa tesi ha il fine di rendere fruibile lo strumento di stima bayesiana e ricerca del picco anche da parte di destinatari la cui familiarità con l'ambiente MATLAB® è scarsa. Infatti le interfacce grafiche, come accennato precedentemente, nascono con il preciso scopo di eliminare le necessità di imparare un linguaggio di programmazione e di digitare quindi dei comandi (o sviluppare delle function) per eseguire un'applicazione.

4.1 Costruire una GUI in MATLAB®

Esistono due metodologie differenti, se vogliamo antitetiche, per costruire un'interfaccia grafica in ambiente MATLAB® [43]:

- sviluppo interattivo: sfruttando GUIDE (GUI development environment), che fornisce strumenti per la progettazione di interfacce utente per le applicazioni personalizzate. Utilizzando il GUIDE Layout Editor, è possibile

progettare graficamente l'interfaccia utente, la quale viene costruita a partire da codice MATLAB® generato in modo automatico. Essa può, inoltre, essere modificata in modo da stabilire il funzionamento dell'applicazione;

- sviluppo a livello di programmazione: per avere un maggior controllo sulla progettazione si può creare in modo diretto del codice MATLAB® che definisca tutte le proprietà e i comportamenti delle varie componenti. MATLAB® contiene funzionalità incorporate che consentono di creare l'interfaccia grafica per l'applicazione specifica. È inoltre possibile aggiungere delle finestre di dialogo, dei controlli (ad esempio pulsanti e cursori) e dei contenitori (come i pannelli e i gruppi di pulsanti). Si rimanda all'Appendice C per una spiegazione più dettagliata degli aspetti tecnici di questi componenti grafici.

In questo lavoro di tesi è stato adottato il secondo approccio. La complessità dell'interfaccia grafica e la necessità di aggiungere nuove componenti durante lo sviluppo hanno reso la procedura di completa programmazione una scelta quasi forzata. GUIDE diventa in questo caso dispendioso e poco gestibile, risulta più conveniente scrivere dei programmi che creino in maniera puntuale e riproducibile la GUI (la figura non necessita di essere salvata tra una sessione di lavoro ed un'altra proprio perchè ogni volta che viene lanciato il programma essa viene completamente ricreata).

4.2 Fase di Mock-up

Al fine di raggiungere un grado di usabilità ottimale, il design effettivo dell'interfaccia è stato preceduto da una fase dedicata alla realizzazione di mock-up. Per mock-up si intende il modello grafico preliminare dello strumento da concretizzare, modello che permette di valutare in anteprima aspetti positivi e negativi legati al solo aspetto estetico del tool, dando modo di procedere con modifiche e correzioni mirate. Sono illustrati nelle sezioni successive alcuni esempi emblematici del processo di design, i quali hanno subito una progressiva ottimizzazione fino a raggiungere il prototipo finale.

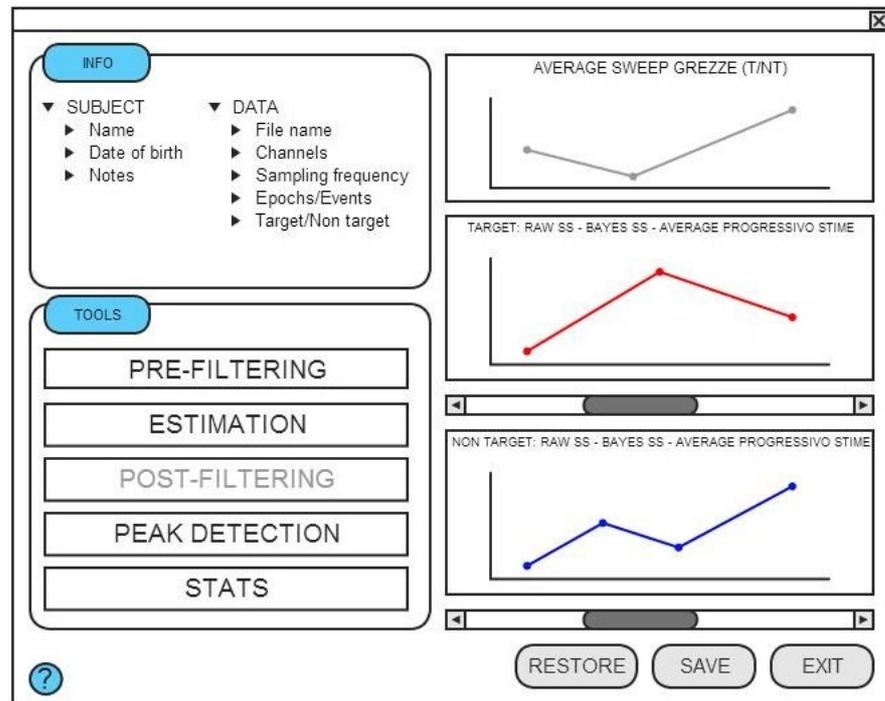


Figura 4.1: Fase di mock-up: finestra principale in grado di gestire le diverse operazioni ed organizzare le informazioni disponibili.

Si è deciso di implementare due diversi plug-in in modo tale da rendere autonome e chiare le due fasi di lavoro: sebbene sia stato concepito un loro utilizzo in cascata, il funzionamento della parte di filtraggio nel contesto bayesiano è stata resa autosufficiente e non vincolata al successivo passo di ricerca del picco.

Nella figura 4.1 è possibile notare come, oltre al riquadro dedicato al plot degli average, sono stati concepiti altri due box sottostanti entro cui accogliere i plot delle singole sweep (target e non target), esplorabili con una barra di scorrimento. Inoltre, i pulsanti disponibili nella sezione “TOOLS” richiamano altrettante finestre pop-up.

Una di queste è riferita alla fase di peak detection (figura 4.2): il menu a tendina consente di scegliere tra i diversi metodi di ricerca del picco implementati. I campi editabili predisposti permettono di settare i parametri tipici del singolo metodo.

La visualizzazione grafica delle singole epoche invece è stata concepita come uno strumento flessibile e allo stesso tempo immediato: è infatti lasciata all’u-

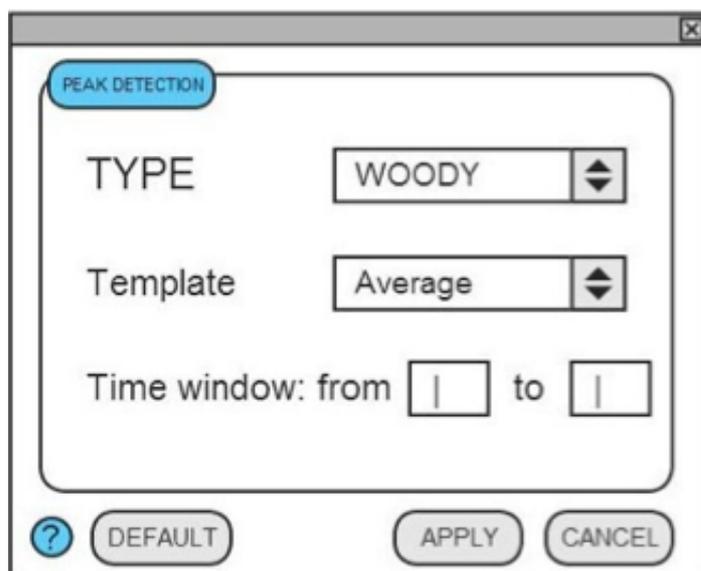


Figura 4.2: Fase di mock-up: finestra dedicata alla ricerca del picco.

tente la scelta dei tracciati da “plottare”, così come sono a disposizione diverse possibilità di esplorazione dei dati (scroll, zoom, data cursor).

4.3 Il tool EEGLab

Questo paragrafo vuole essere una sintetica introduzione al toolbox EEGLAB, al fine di mettere a fuoco il contesto entro cui si è scelto di integrare la Graphical User Interface del software implementato nel presente lavoro di tesi. Per informazioni più dettagliate si rimanda al sito ufficiale [44] e alle trattazioni di Delorme e collaboratori [3][45].

EEGLAB può essere definito come un toolbox interattivo di MATLAB®[®], dedicato all’elaborazione di dati e segnali elettrofisiologici. Esso consente di effettuare il filtraggio dei dati, di condurre delle analisi nei domini del tempo e della frequenza, di estrarre informazioni dai dati e di visualizzarne l’average o i singoli trial. Esso dispone inoltre di un’interfaccia grafica (GUI, Graphical User Interface) in grado di alleviarne la complessità e renderne accessibili le funzionalità: tutte le potenzialità del software risultano in questo modo alla portata di un uso clinico.

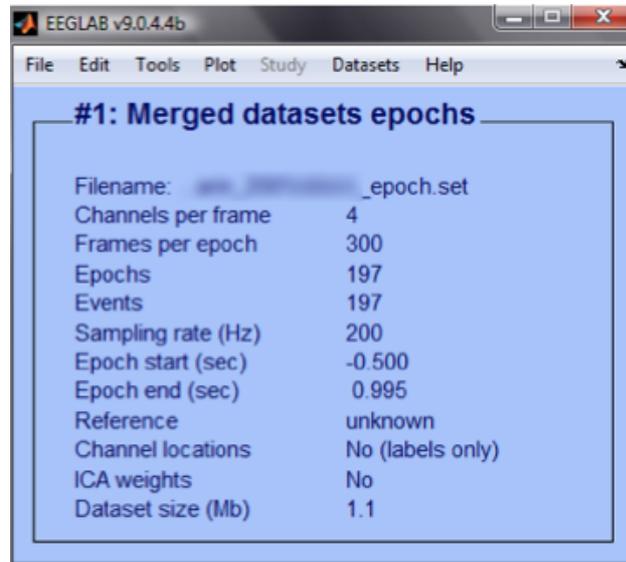


Figura 4.3: Schermata iniziale del toolbox EEGLAB.

Rimanendo tuttavia un toolbox sviluppato in ambiente MATLAB®¹, EEGLAB offre non solo la possibilità di accedere alle sue variabili d'appoggio (organizzate nel Workspace e manipolabili per mezzo di script o function esterne direttamente da riga di comando), ma anche di implementare plug-in personalizzati integrandoli nei menu della sua interfaccia grafica. In questo modo è come se EEGLAB costituisse a tutti gli effetti una piattaforma open-source¹ su cui poter basare lo sviluppo di nuove funzionalità e metodi “custom”, i quali divengono automaticamente portabili nella misura in cui si desidera condividere con altri utenti quanto implementato.

Per quanto riguarda la portata dell'impatto di questo toolbox, basti considerare che le 400 MATLAB® function di cui consta sono state scaricate da ricercatori di almeno 88 domini nazionali, che il paper ufficiale di presentazione è stato citato oltre 2600 volte, e che i download giornalieri del pacchetto principale si aggirano intorno ai 30-40. Il tool implementato nel presente lavoro di tesi – e descritto esaurientemente nei prossimi paragrafi – è stato concepito per essere integrato nella

¹Il toolbox EEGLAB è distribuito sotto la GNU General Public License, la quale assicura all'utente non solo la piena libertà di utilizzo del software, ma anche la sua copia, modifica e distribuzione. Licenza GNU GPL è dunque sinonimo di software libero, che rimane tale anche dopo eventuali modifiche, correzioni o ampliamenti (“persistenza” della GNU GPL, battezzata copyleft – in contrasto con il ben più noto concetto di copyright).

piattaforma EEGLAB. Fattori catalizzanti per tale scelta sono stati la flessibilità, la filosofia open-source e la possibilità di sfruttare le già presenti funzionalità del toolbox, i quali, accostati ad una sua comprovata diffusione in ambito clinico, si rivelano essenziali per offrire un potenziale respiro allo strumento realizzato.

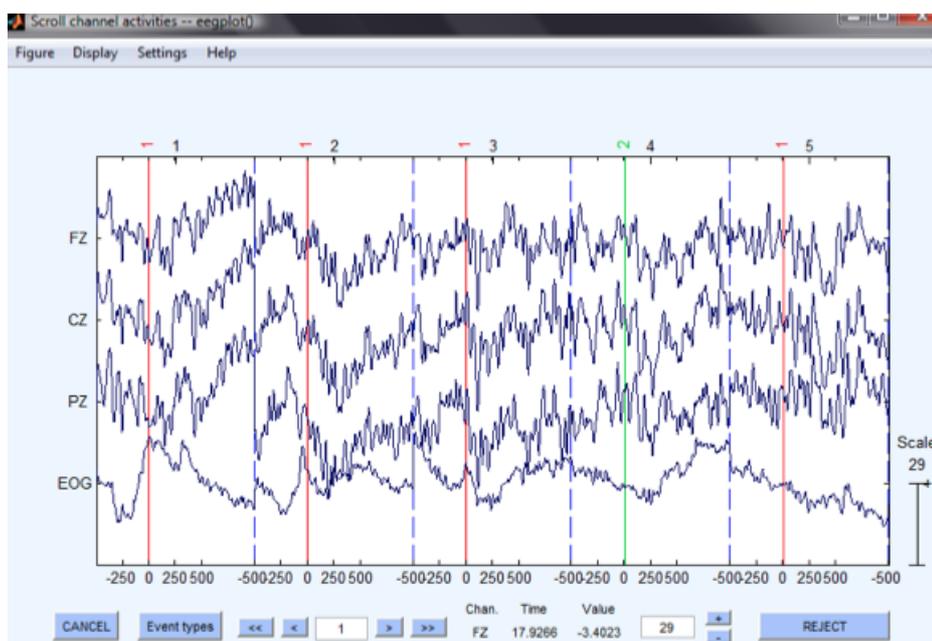


Figura 4.4: Visualizzazione di un file .set epocato tramite EEGLAB. E' possibile notare i marker relativi agli eventi non target (1, in rosso) e target (2, in verde), così come i delimitatori di ogni singola sweep.

4.4 Realizzare un plug-in per EEGLab

4.4.1 Creare una Text Input Window

Per aggiungere nuove funzionalità a EEGLAB sono necessarie un paio di funzioni: la prima elabora il segnale d'interesse (signal processing function) mentre la seconda funge da "accompagnamento" (pop_function) ed è strettamente connessa alla procedura principale.

La pop_function crea un'interfaccia grafica per la funzione principale² ma, al

²Il compito della pop_function è proprio quello di aprire una finestra di testo dove l'utente possa inserire gli opportuni argomenti di input relativi alla signal processing function.

contrario di essa, deve rispettare qualche vincolo³ di programmazione in più:

- il primo parametro in ingresso deve essere sempre la struttura dati 'EEG';
- il secondo argomento può specificare se la funzione di elaborazione del segnale sfrutterà l'algoritmo ICA;
- ulteriori parametri in input sono opzionali⁴;
- deve ritornare una stringa che contenga la chiamata alla funzione principale (o in rari casi alla `pop_function` stessa).

Un esempio pratico

In questo breve paragrafo viene presentato il codice MATLAB[®] utilizzato per creare la `pop_function` relativa all'algoritmo MTL.

Nella parte centrale si definiscono le componenti grafiche che si vuole facciano parte della GUI e se ne delinea la geometria, nella parte finale si introducono le variabili da "passare" alla signal processing function.

```
function [EEG,com] = pop_mtl_main_eeglabTBOX(EEG,sel_ch,
flag_EOG_rej, passo, ord_max, criterion);

chan_labels = [];
chan_labels = cellstr(strvcat(chan_labels, EEG.chanlocs(:).labels));
alltypes = {'AIC' 'FPE'};
step_down = round(EEG.pnts/150);

% Definizione componenti grafiche

uilist = {...
```

³La signal processing function ha come unico vincolo il primo argomento di input che deve necessariamente essere, come per la `pop_function`, la struttura dati 'EEG'.

⁴E' da tenere presente che, per quanto riguarda gli argomenti d'ingresso, la firma della funzione principale è la medesima della `pop_function` associata.

```

{'style' 'text' 'string' 'Channel to consider:' 'fontweight' 'bold'}...
{}...
{'style' 'popupmenu' 'string' strvcat(chan_labels{:})}...
{}...
{'style' 'text' 'string' 'Removing artifacts from EEG'
'fontweight' 'bold'}...
{}...
{}...
{'style' 'checkbox' 'value' 1 'string' 'EOG rejection'
'fontweight' 'bold'}...
{'style' 'text' 'string' 'Downsampling step' 'fontweight' 'bold'}...
{}...
{}...
{'style' 'edit' 'string' num2str(step_down)}...
{'style' 'text' 'string' 'AR Model Maximum Order'
'fontweight' 'bold'}...
{}...
{}...
{'style' 'edit' 'string' '10'}...
{}...
{'style' 'text' 'string' 'Criterion to identificate the best AR order:'
'fontweight' 'bold'}...
{}...
{'style' 'popupmenu' 'string' strvcat(alltypes{:})}...
};

% Struttura geometrica della Text Input Window

geometry = {[1 0.2 0.3] 0.2 [1 0.2 0.2 0.4] [1 0.2 0.2 0.4]
[1 0.2 0.2 0.4] 0.3 [1 0.2 0.3]};

result = inputgui('geometry',geometry,'uilib',uilib,'title','MTL',

```

```
'helpcom', 'pophelp(''pop_mtl_main_eeglabTBOX'''))';

if isempty(result), return; end;

% Variabili da passare alla funzione principale

sel_ch = chan_labels{result{1}};
flag_EOG_rej = result{2};
passo = eval(result{3});
ord_max = eval(result{4});
criterion = alltypes{result{5}};

com = 'Thank you for using MTL_toolbox';

%% nell'ultima riga viene richiamata la signal processing function
```

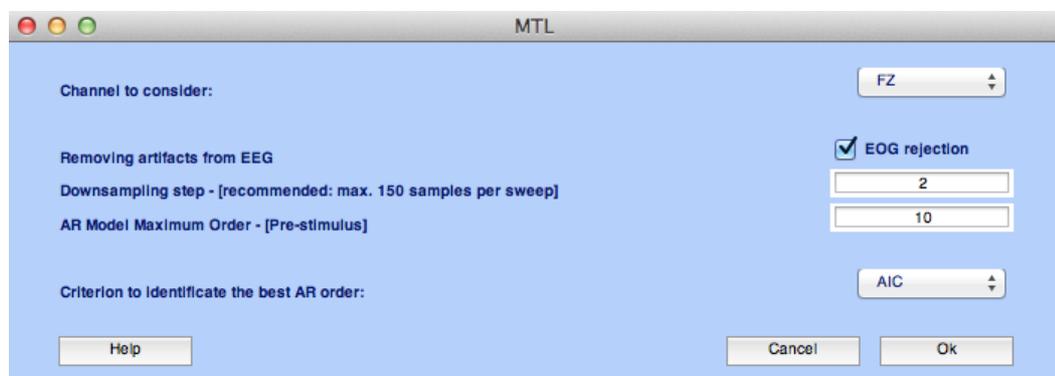


Figura 4.5: Schermata iniziale del plug-in di stima bayesiana MTL derivata dalla `pop_function` precedentemente descritta.

4.4.2 Aggiungere un plug-in

Una volta scritte le due funzioni presentate nel paragrafo precedente è necessario aggiungere un plug-in⁵ ad EEGLAB in modo da poter effettivamente visualizzare la schermata di figura 4.5.

Per creare un plug-in è sufficiente scrivere una procedura MATLAB® il cui nome inizi con *eegplugin_* ed inserirla nella subdirectory dedicata a questo tipo di funzioni o più semplicemente nella cartella principale di EEGLAB. Questa funzione deve necessariamente avere tre argomenti:

- il primo ('fig') è l'handle⁶ alla finestra principale di EEGLAB;
- gli altri due parametri sono strutture passate da EEGLAB che permettono alla plug-in function di controllare i vari parametri o di individuare dei potenziali errori.

Per creare infine un sub-menu è sufficiente aggiungere dei comandi specifici (presentati nel prossimo paragrafo con un esempio) alla plug-in function.

Un esempio pratico

In questo breve paragrafo viene presentato il codice MATLAB® della plug-in function finalizzato alla costruzione del sub-menu MTL. Il tag indica la posizione del sub-menu all'interno del toolbox EEGLAB (in questo caso sotto la sezione PLOT) mentre la label conferisce il nome effettivo al sub-menu stesso. Da notare infine la chiamata alla *pop_function* che permette di visualizzare la schermata di figura 4.5.

```
function eegplugin_MTL(fig,try_strings,catch_strings);
```

⁵Il plugin in campo informatico è un programma non autonomo che interagisce con un altro programma per ampliarne le funzioni.

⁶MATLAB® permette di assegnare alle variabili valori di tipo "funzione". Un valore di tipo funzione può essere assegnato a una variabile detta handle. L'handle può essere applicato a opportuni argomenti per ottenere una invocazione della funzione.

```
% Creazione del sub-menu

plotmenu = findobj(fig,'tag','plot');
uimenu(plotmenu, 'label', 'MTL',
'CallBack', ['[EEG,LASTCOM] = pop_mtl_main_eeglabTBX(EEG)']);
```

4.5 Componenti grafiche inseribili in una GUI

La maggior parte degli elementi di una interfaccia grafica utente, siano essi di controllo o puramente grafici, sono creati con lo scopo di perseguire un'azione, modificare delle impostazioni per azioni future o visualizzare risultati. Gli elementi di controllo utilizzati in questa tesi includono push button, toggle button, radio button, slider, edit text control, static text control, pop-up menu e check box. Altri componenti fondamentali sono axes, panel, table e button group. Di seguito se ne illustrano brevemente le caratteristiche principali:

- Axes: componente che permette alla GUI di visualizzare grafici, come immagini e plot;
- Button Group: pannello utilizzato per gestire la selezione di più radio button e toggle button ad azione, di solito, mutuamente esclusiva;
- Check Box: casella di spunta che genera un'azione se selezionata ed è utile per indicare lo stato di un'opzione o un attributo e nel caso in cui l'utente necessiti di esprimere un certo numero di scelte indipendenti e non mutuamente esclusive; presenta due stati possibili: on, in cui la casella quadrata contiene una spunta, e off, in corrispondenza del quale la casella si presenta vuota;
- Editable Text: casella di testo che permette all'utente di inserire un contenuto o modificarlo, utilizzata se si vuole visualizzare in input una stringa o una serie di numeri; è indicabile inizializzare la casella di testo con una

stringa o un valore di default, così da non rendere l'utente necessariamente obbligato a inserirne uno;

- **Panel:** componente che permette la suddivisione grafica della figura, conferendo maggior modularità e gestibilità anche alla codifica relativa; ciascun pannello può avere un titolo e oggetti propri al suo interno;
- **Pop-Up Menu:** menu a tendina che si apre verso il basso permettendo la scelta di una sola tra le opzioni in lista; l'opzione selezionata viene visualizzata nella casella che rappresenta il menu in forma chiusa; a differenza di un editable text, il pop-up menu è riconoscibile da un simbolo posto sull'estremo destro dell'oggetto in cui un triangolino rivolto in basso indica il verso di apertura del menu;
- **Push Button:** pulsante che genera lo stesso tipo di azione ogni volta che viene premuto; finché il pulsante è premuto esso appare nel suo stato di depressione, mentre riassume l'aspetto originario quando viene rilasciato. Un push button invoca un evento in maniera immediata ed è uno tra i componenti maggiormente utilizzati nelle interfacce grafiche; poiché questi oggetti rappresentano un'azione, spesso sono etichettati con un verbo o una parola che la richiami alla mente in maniera immediata e intuitiva;
- **Radio Button:** pulsante simile al check box che indica un'opzione da selezionare; è tipicamente mutuamente esclusivo all'interno di un gruppo di pulsanti dello stesso tipo e presenta due stati ad esso associati (selezione e deselegione) in corrispondenza dei quali il simbolo circolare dell'oggetto contiene o meno un pallino;
- **Slider:** componente che accetta input numerici, all'interno di un range pre-stabilito, per mezzo di una barra scorrevole e due frecce agli estremi; la posizione corrente della barra indica la localizzazione del valore corrispondente all'interno del range di valori;
- **Static Text:** componente che consente il controllo della visualizzazione di stringhe testuali all'interno dell'interfaccia ma non avvia nessun tipo di azio-

ne; è generalmente utilizzato come etichetta di altri componenti di controllo e, a differenza dell'editable text, non può essere modificato dall'utente;

- Table: oggetto tabella, formato da una serie di colonne e righe numerate ed etichettabili, che può contenere nelle proprie caselle stringhe di testo, numeri o indicatori di scelta; le dimensioni della tabella si adattano automaticamente alle dimensioni del contenuto oppure possono essere impostate nella codifica; è possibile rendere alcune colonne internamente modificabili;
- Toggle Button: componente binario, o bipolare, che genera un'azione se premuto e che presenta due stati, on e off; non presenta uno stato intermedio, perciò esso rimane nel suo stato selezionato o deselezionato finchè l'utente non clicca nuovamente su di esso.

Ciascun oggetto grafico, sia esso componente di controllo o oggetto di visualizzazione grafica, possiede un set di proprietà che possono essere manipolate dal programmatore fino al raggiungimento dell'aspetto desiderato. Tali proprietà, numerose e dettagliate dal punto di vista gestionale, non vengono in questa sede presentate. Si ritiene superfluo e dispendioso un approfondimento ulteriore a riguardo, essendo questa la presentazione di una GUI e non un manuale teorico su come realizzarne una.

Capitolo 5

GUI dell'algoritmo MTL per la stima single-trial

In questa sezione vengono presentate le varie scelte progettuali adottate durante lo sviluppo dell'interfaccia grafica relativa a MTL¹.

La parte di stima bayesiana è raggiungibile dal menu 'Tools' di EEGLAB dopo aver importato nel toolbox un dataset epocato valido mediante l'istruzione 'Load existing dataset' presente nel menu 'File'. Cliccando su 'MTL' si accede alla schermata presentata precedentemente. Il menu a tendina consente di selezionare – tra le derivazioni disponibili nel file di registrazione importato in EEGLAB – il canale da considerare per l'analisi ERPs. Sono inoltre lasciate all'utente le scelte riguardanti l'eventuale reiezione degli artefatti oculari (attivabile mediante il relativo flag), il fattore di downsampling per permettere un minor onere computazionale (l'opzione consigliata è di 150 campioni per sweep), l'ordine massimo del modello autoregressivo che descrive il pre-stimolo² e il criterio utilizzato per l'identificazione (AIC o FPE). Al fine di facilitare l'utilizzo del tool, sono stati

¹Le GUI relative a B2S e MTL sono identiche al di fuori del pannello 'Regularization' che ospita necessariamente parametri differenti. Viene quindi presentato solamente il software dell'algoritmo MTL, significativo per entrambi gli approcci di stima bayesiana.

²Viene impostato di default un valore relativamente basso in quanto il criterio di Akaike e il Final Prediction Error "trovano" spesso degli ordini ottimi "esagerati". Inoltre, l'utilizzo del fattore di downsampling per la stima, rende ancor più necessaria e quasi obbligata la scelta di ridurre i gradi di libertà parallelamente ai campioni.

preimpostati nei campi editabili dei valori di default interattivi, ovvero variabili in dipendenza delle opzioni scelte dai menu a tendina allo scopo di rappresentare sempre i setting ottimali. La pressione del tasto 'Help' conduce ai commenti introduttivi ed esplicativi della pop function che gestisce la chiamata a MTL; il tasto 'Cancel' riporta al main screen di EEGLAB dal quale si proviene; il tasto 'Ok' dà avvio al processo di stima.

5.1 Opzioni di scorrimento

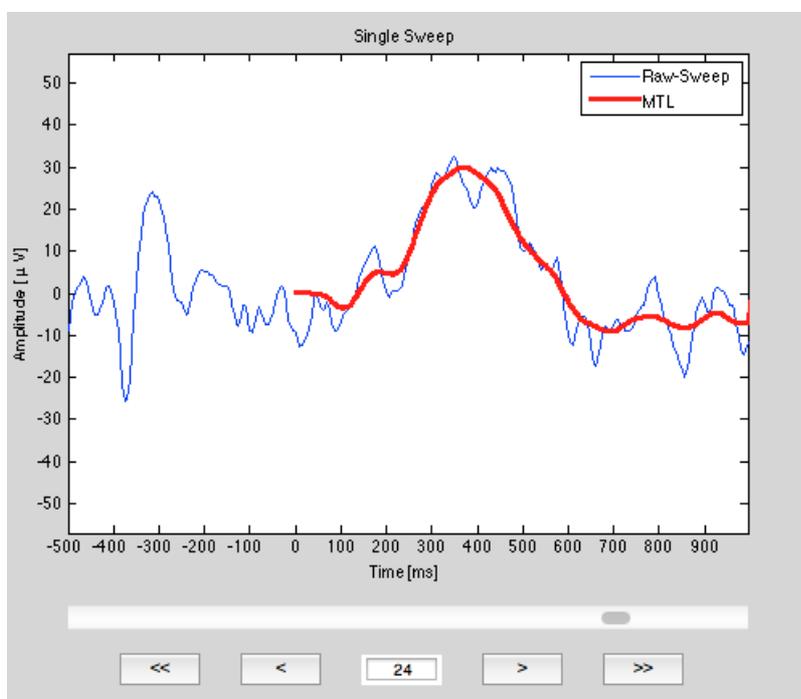


Figura 5.1: Visualizzazione della sweep i -esima tramite le opzioni di scorrimento sottostanti.

La prima informazione visualizzabile è aggiornata automaticamente in conseguenza dello scorrimento delle singole epoche, il quale è governato dagli strumenti posti in calce al grafico:

- freccia singola a destra o a sinistra: plot della sweep precedente o successiva;
- freccia doppia a destra o a sinistra: plot della sweep salvata cinque posizioni prima o dopo rispetto alla sweep corrente;

- barra di scorrimento: l'esplorazione delle sweep può avvenire mediante clic su una delle frecce o trascinamento del cursore mobile. L'uso tipico della scrollbar consente anche di utilizzare la funzionalità di scorrimento rapido tenendo premuta la freccia desiderata;
- campo numerico editabile: offre un accesso immediato ad una particolare posizione. Utile nel caso in cui si voglia esaminare il plot di una determinata sweep senza dover per forza scorrere le precedenti;
- frecce direzionali della tastiera: risultano attive dopo aver interagito con la scrollbar. Le frecce "sinistra" e "in alto" riportano alla sweep salvata nella posizione precedente rispetto a quella corrente, le frecce "destra" e "in basso" conducono invece a quella successiva.

5.2 Pannelli principali

5.2.1 Pannello relativo al pre-stimolo

Questa componente grafica non è inizialmente visibile. Cliccando sul pulsante '>>' presente a destra del grafico dove vengono visualizzate le varie epoche³, la figura si "ingrandisce" ospitando questo ulteriore riquadro. Infatti, un clinico che si occupa dell'analisi dei vari trial non è solitamente interessato agli aspetti puramente ingegneristici presenti nel pannello e può quindi completamente ignorarlo.

Come si può notare nella figura 5.2 la casella ospita un istogramma e due sotto-pannelli. Nel grafico sono visualizzabili i valori AIC o FPE per ogni ordine del modello autoregressivo analizzato (la barra plottata in giallo indica l'ordine ottimo stimato per la current sweep).

Nel riquadro superiore sono presenti le varie informazioni relative all'identificazione AR mentre in quello inferiore vengono elencati, tramite una listbox, i coefficienti del modello.

³Si può osservare questo pulsante nella pagine successive, in figura 5.5.

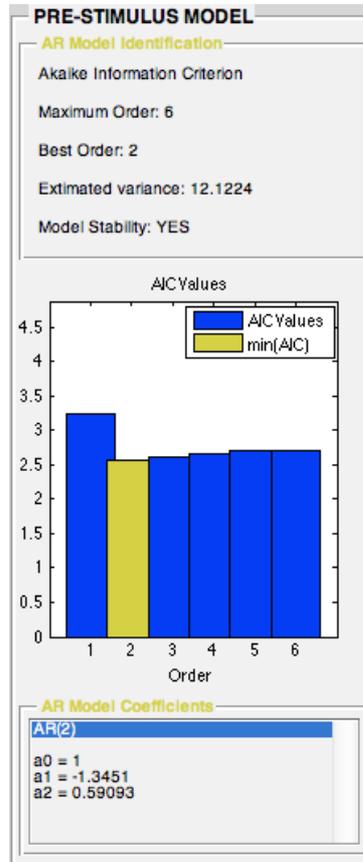


Figura 5.2: Pannello relativo al pre-stimolo.

Il codice MATLAB® utilizzato, nelle varie funzioni di 'Callback', per il plot dell'istogramma e l'aggiornamento dei parametri è il seguente:

```

global loss maxo variances stable coeff crit

% loss: valori della loss_function
% maxo: ordine massimo analizzato

a = round(get(S.sl,'value')); % valore dello Slider

[value indice] = min(loss(a,:));
bar(S.ax4,(1:1:maxo),loss(a,:),1.2,'FaceColor','b')
hold(S.ax4,'on');

```

```
bar(S.ax4,indice,value,1.2,'FaceColor',[204/255, 204/255, 0])
if indice < maxo
    bar(S.ax4,(indice+1:maxo),loss(a,indice+1:maxo),1.2,'FaceColor','b')
end
hold(S.ax4,'off');
str1 = [crit 'Values'];
str2 = ['min(' crit ')'];
legend(S.ax4,str1, str2);
xlabel(S.ax4,'Order');
title(S.ax4,str1);
val = 1.5*max(loss(a,:));
axis(S.ax4,([0 maxo+1 0 val]));

str = ['Best Order: ' num2str(indice)];
set(S.ar3,'string',str);

str = ['Extimated variance: ' num2str(variances(a,1))];
set(S.ar4,'string',str);
if stable(a,1) == 1
    str = 'Model Stability: YES';
else
    str = 'Model Stability: NO';
end
set(S.ar5,'string',str);

alltypes = {};
str = ['AR(' num2str(indice) ')'];
alltypes{1} = str;
alltypes{2} = ' ';
for i = 1:indice+1
    str = ['a' num2str(i-1) ' = ' num2str(coeff(a,i))];
    alltypes{i+2} = str;
```

```
end  
set(S.ar6,'string',alltypes);
```

5.2.2 Pannelli informativi

Questi pannelli sono collocati nella zona superiore al grafico principale della GUI. Essi offrono all'utente delle informazioni essenziali riguardo ai dati e al processo di regolarizzazione.



Figura 5.3: *Pannelli informativi nell'interfaccia grafica relativa a MTL.*

Il riquadro 'DATA' contiene il numero della sweep corrente sul numero totale, il nome del file in analisi, la frequenza di campionamento e il canale di acquisizione cui i tracciati visualizzati fanno riferimento. La prima informazione è aggiornata automaticamente in conseguenza allo scorrimento delle singole epoche.

Un ulteriore pannello informativo riferito ai dettagli relativi al metodo di regolarizzazione è stato posizionato a lato di quello precedentemente descritto. Esso contiene:

- il nome del criterio adottato per il parametro di regolarizzazione, il valore di quest'ultimo e i gradi di libertà concessi alla sweep corrente nell'interfaccia relativa a B2S;
- il nome del criterio adottato per i parametri di regolarizzazione e i rispettivi valori (uguali per tutte le epoche) nell'interfaccia grafica relativa a MTL.

5.2.3 Pannello contenente i vari comandi

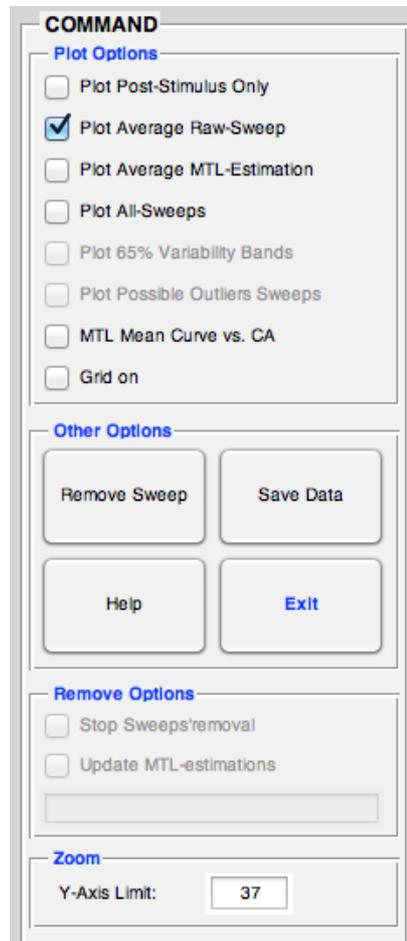


Figura 5.4: *Pannello dei comandi.*

Questo riquadro contiene al suo interno 4 sotto-pannelli ai quali è assegnata una funzione specifica:

- Plot Options: dotato di una lista di checkboxes che consentono di manipolare graficamente le varie sweeps;
- Other Options: dotato di una lista di pulsanti non deputati alla visualizzazione dei trials;
- Remove Options: consente di aggiornare le stime dopo un tot di rimozioni;
- Zoom: consente di cambiare la risoluzione (lungo l'asse delle ordinate) con la quale vengono visualizzate le varie epoche.

5.3 Plot Options

Una volta completato l'algoritmo di stima bayesiana si apre automaticamente una finestra che consente la visualizzazione delle varie epoche. Da tale videata è possibile non solo apprezzare i plot delle singole sweep grezze e delle relative stime, ma anche visualizzare il plot dei segnali medi (CA e Average MTL-estimation) semplicemente sfruttando i checkboxes omonimi.

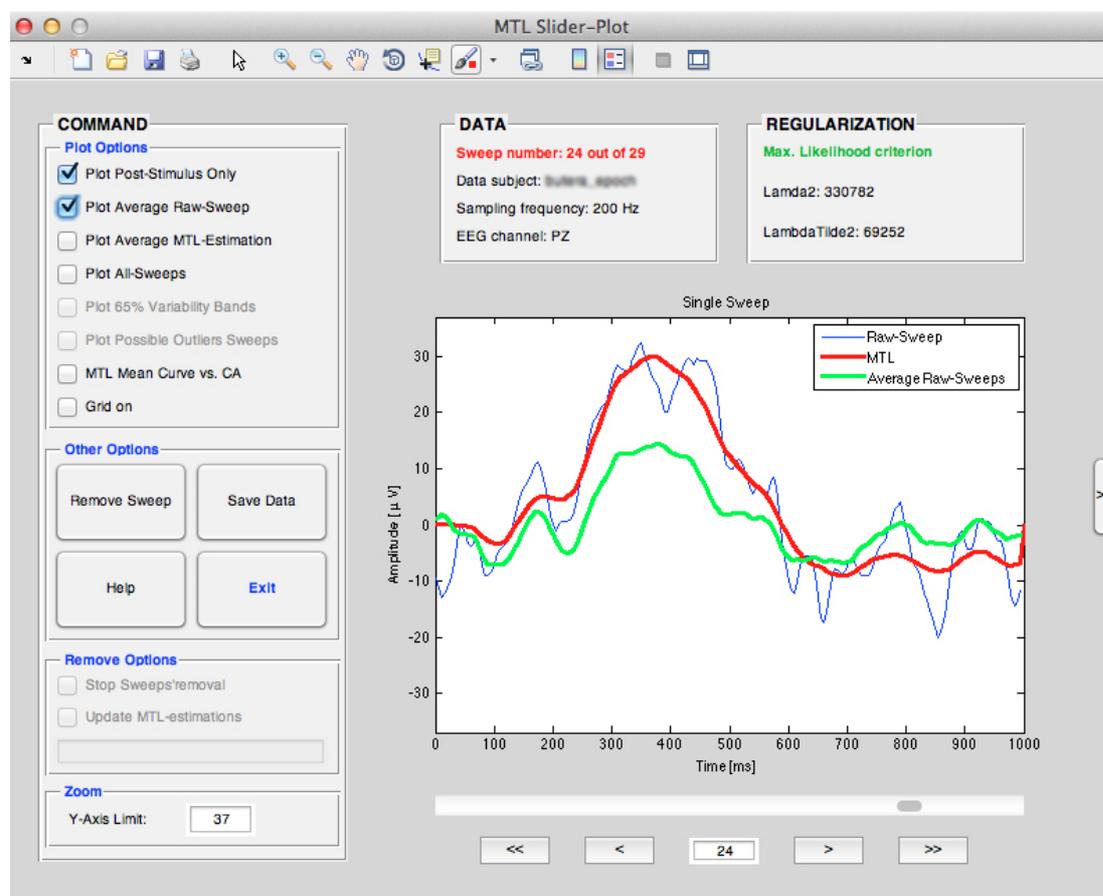


Figura 5.5: Visualizzazione dell'Average.

Questa possibilità consente di avere sempre a disposizione il riferimento della media delle epoche target, utile per capire quanto ogni singolo trial si discosti o disattenda quello che è l'andamento ottenuto in average. E' inoltre data facoltà all'utente di rimuovere un singolo trial dall'analisi tramite il tasto 'Remove Sweep', nell'eventualità in cui vi sia la necessità di eliminarla in modo manuale ad esempio per causa di evidenti artefatti. Alla prima pressione del tasto è stata predisposta

la visualizzazione di un warning che avvisa l'utente dell'irreversibilità dell'azione. La rimozione di una sweep comporta l'immediato ricalcolo dell'average, in quanto esso risente inevitabilmente dell'operazione effettuata.

Inoltre, il grafico delle stime può essere gestito attraverso la tipica toolbar di MATLAB®: sono in questo modo disponibili le funzioni di salvataggio e stampa dell'immagine, di esplorazione del grafico e di rilevazione dei valori delle forme d'onda (mediante gli strumenti zoom, pan e data cursor).

5.3.1 Variabilità

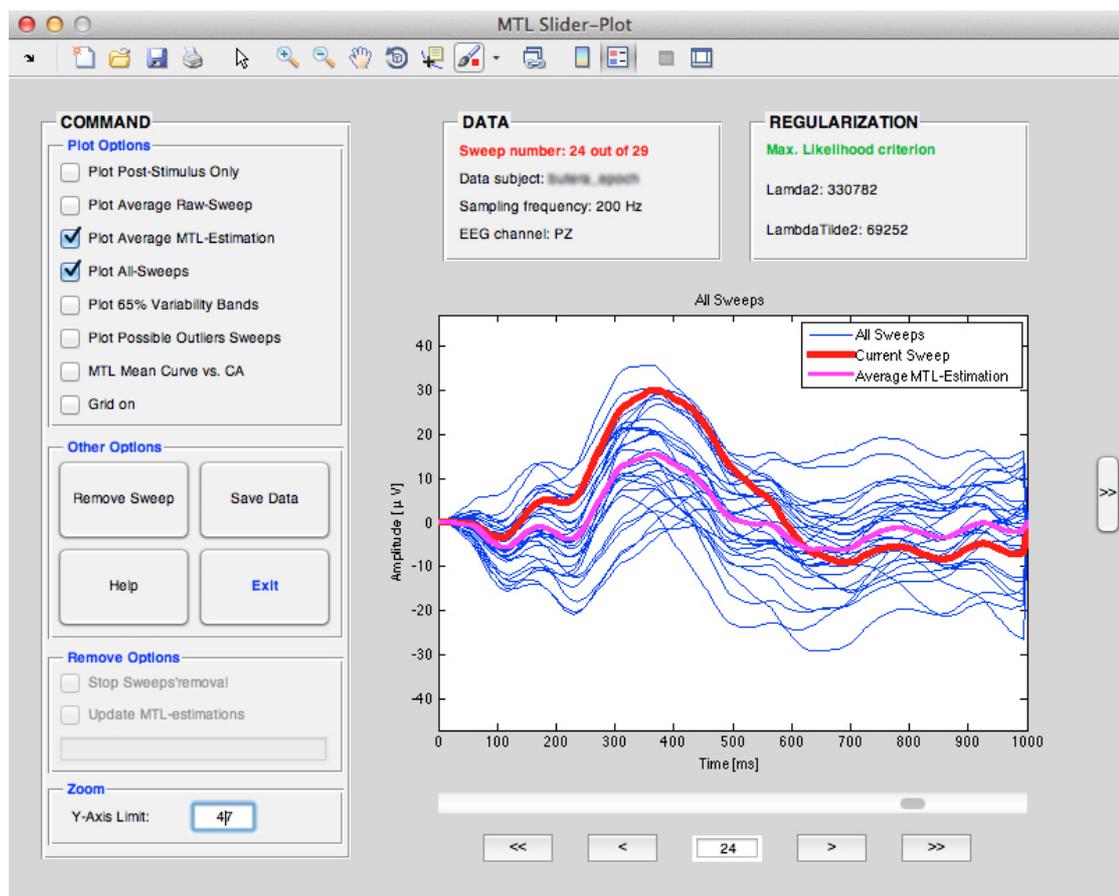


Figura 5.6: Funzionalità 'Plot All Sweeps' del tool MTL. In blu le stime single-trial fornite dall'algorithm, in rosso la stima corrente e in viola l'average delle stime MTL.

La variabilità si può analizzare utilizzando il checkbox 'Plot All Sweeps' il quale, una volta selezionato, consente di visualizzare le bande di confidenza al 65%⁴ e alcune possibili outliers sweeps. Questa valutazione statistica è stata concepita per aiutare l'utente nella scelta di rimozione di un'epoca: il confronto grafico della stima corrente con le altre stime single-trial del dataset ed eventualmente con l'average delle sweep grezze dà modo infatti di effettuare un'analisi diretta ed immediata sulla base della totalità dei dati a disposizione.

Il codice MATLAB[®] che permette di plottare tutti i trial stimati contemporaneamente è il seguente⁵:

```

global y1

% y1: vettore contenente le stime single-trial
% S.x1: vettore contenente la griglia temporale per le stime
% S.avg2: media non pesata delle stime MTL

S.avg2 = mean(y1,2);
hSLines = plot(S.ax,S.x1,y1,'b');

hold(S.ax,'on');
hCLines = plot(S.ax,S.x1,y1(:,a),'r','LineWidth',4);
hDLines = plot(S.ax,S.x1,S.avg2,'m','LineWidth',3);
hold(S.ax,'off');

hSGroup = hggroup('parent',S.ax);

```

⁴Costruire le bande di confidenza al 65% significa determinare istante per istante l'intervallo in cui cadono il 65% delle sweeps stimate. In altri termini bisogna trovare per ogni t , il 17° e l'83° percentile della distribuzione dei valori.

⁵Bisogna tenere conto che questa parte di programma consente di visualizzare tutte le sweeps stimate con l'aggiunta dell'average delle stime MTL. Esiste anche il caso in cui questo secondo segnale è sostituito dall'average convenzionale oppure quello in cui vengono plottati entrambi. Queste due differenti possibilità prevedono l'utilizzo di comandi leggermente diversi che per semplicità non vengono riportati.

```

hCGroup = hggroup('parent',S.ax);
hDGroup = hggroup('parent',S.ax);

set(hSLines,'Parent',hSGroup)
set(hCLines,'Parent',hCGroup)
set(hDLines,'Parent',hDGroup)

% Include these hgroups in the legend:

set(get(get(hSGroup,'Annotation'),'LegendInformation'),
'IconDisplayStyle','on');
set(get(get(hCGroup,'Annotation'),'LegendInformation'),
'IconDisplayStyle','on');
set(get(get(hDGroup,'Annotation'),'LegendInformation'),
'IconDisplayStyle','on');
legend(S.ax,'All Sweeps','Current Sweep','Average MTL-Estimation')

```

Il codice MATLAB® che permette, invece, di visualizzare le variability bands è il seguente:

```

m = get(S.sl,'max'); % valore massimo dello slider
z = y1';
YORD = sort(z);
ind17 = fix(m*0.17);
ind83 = fix(m*0.83);
y17 = YORD(ind17,:);
y83 = YORD(ind83,:);
yconf = [y17; flipud(y83)];
tconf = [S.x1; flipud(S.x1)];

hhh = fill(tconf,yconf,'y');

```

```

set(hhh,'parent',S.ax);
set(hhh,'edgecolor','none');

hold(S.ax,'on');
plot(S.ax,S.x1,y1(:,a),'r--')
plot(S.ax,S.x1,S.avg2,'m--')
hold(S.ax,'off');

legend(S.ax,'65% Variability Bands','Current Sweep','Average MTL')

```

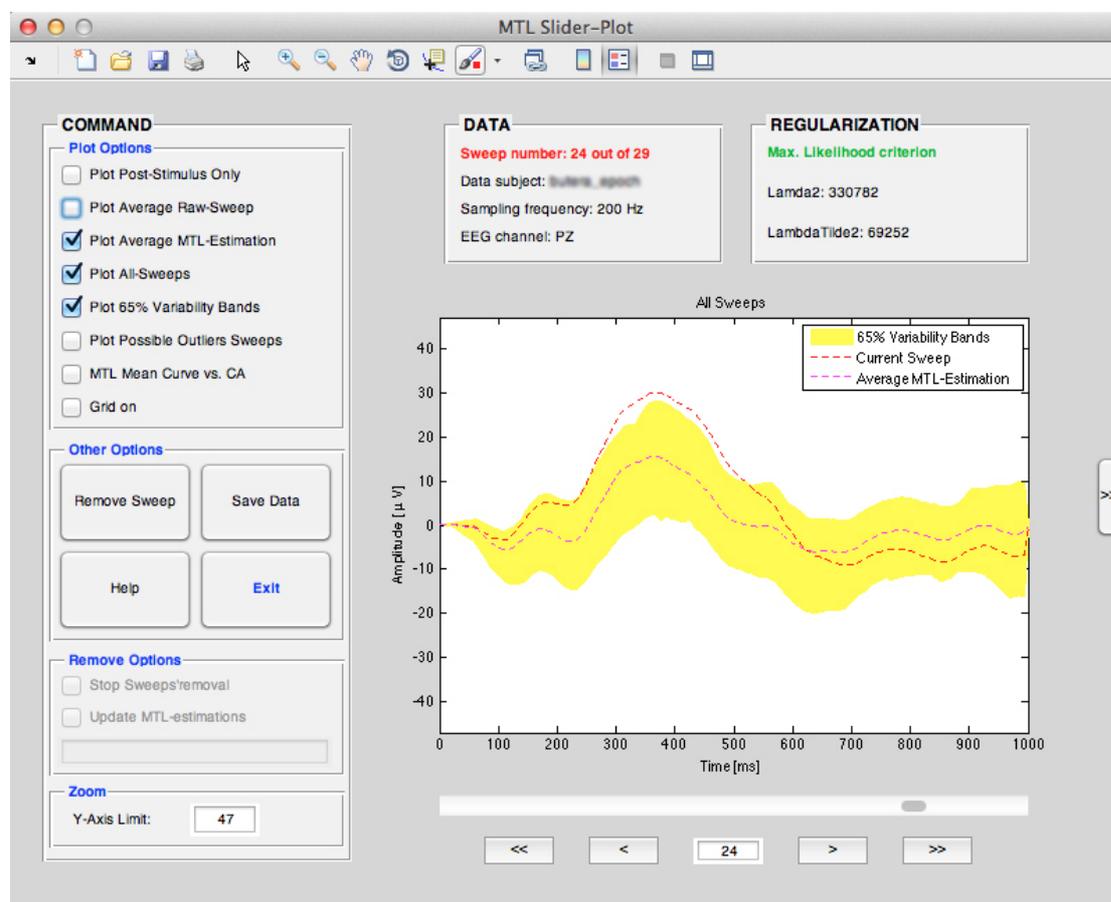


Figura 5.7: Funzionalità 'Plot 65% Variability Bands' del tool MTL.

Infine, come spiegato precedentemente, vengono plottati alcuni possibili outliers per rendere la rimozione ancora più agevole. Su di essi è inoltre applicata un'etichetta con il numero del trial stimato. In questo modo si può sfruttare con

immediatezza il campo numerico editabile e raggiungere la sweep d’interesse per operare la rimozione. Gli outliers vengono discriminati dal resto del gruppo tramite il calcolo della media del segnale: i trial con media maggiore e minore sono considerati dei candidati appetibili per l’eliminazione dal set di dati completo. Ovviamente, una volta rimosso un outlier, il software procederà ad un rapido aggiornamento che permetterà il calcolo di un “sostituto”.

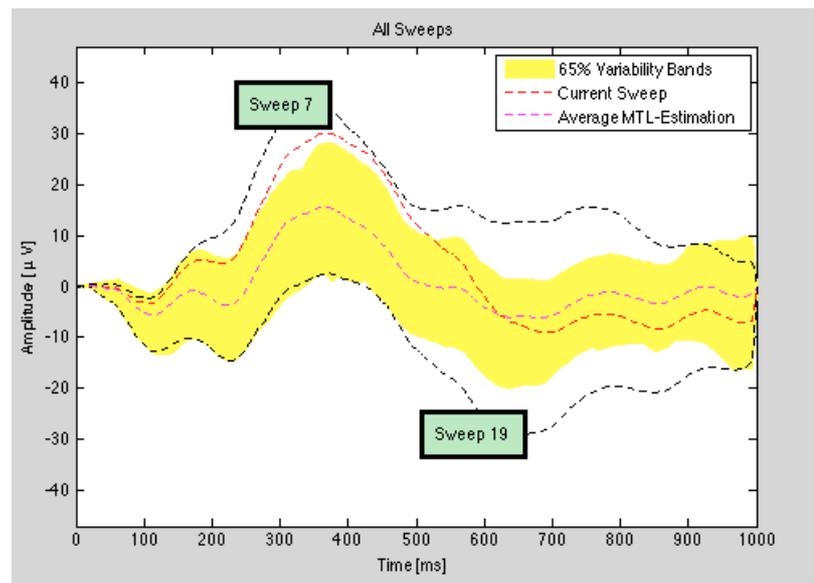


Figura 5.8: *Visualizzazione dei possibili candidati alla rimozione.*

5.3.2 Visualizzazione della curva media

La media calcolata a posteriori (Average MTL-Estimation o Average B2S-Estimation) non è una misura significativa per gli approcci di stima bayesiana. Infatti, la curva media viene stimata nel primo passo algoritmico con Bayesian-Two-Steps e contestualmente alle stime single-trial con Multi-Task-Learning. Si è scelto di plottare con immediatezza gli average calcolati a posteriori poichè, conseguentemente a una rimozione, essi vengono aggiornati rapidamente.

In ogni caso, viene concessa all’utente la possibilità di visualizzare la curva media calcolata nel contesto bayesiano in modo da confrontarla con l’average convenzionale⁶. Mediante il checkbox ‘MTL Mean Curve vs. CA’ si può apprezzare graficamente la differenza tra i due approcci di stima della sweep media.

⁶Uno degli obiettivi dell’approccio bayesiano è infatti quello di offrire le stime single-trial

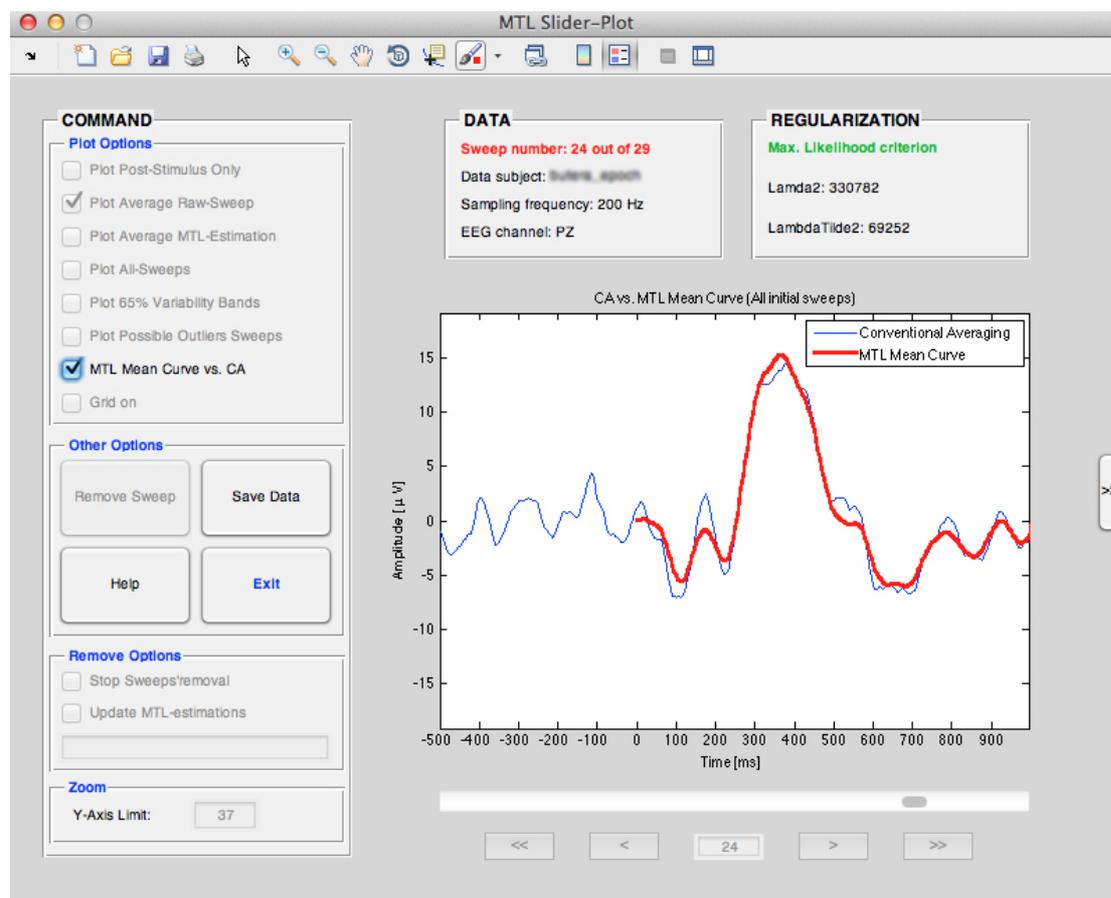


Figura 5.9: Visualizzazione dell'Average nel contesto bayesiano.

5.4 Altre opzioni

E' contemplato, prima dell'uscita dall'interfaccia mediante il tasto 'Exit', il salvataggio dei dati ricavati come output dall'algoritmo di stima ed eventualmente manipolati: la finestra di dialogo che si apre alla pressione del tasto 'Save Data' è tipica del sistema operativo. La posizione predefinita coincide con la directory impostata come current folder in MATLAB®. I dati vengono salvati come file di estensione .mat in modo tale da poter essere gestiti in maniera elementare dall'intera struttura di peak-detection. Qualunque sia il nome assegnato al file (predefinito come nome del soggetto), esso contiene una variabile di tipo struct chiamata 'MTL_erp' (o B2S_erp) la quale include a sua volta i seguenti campi:

contestualmente ad un miglioramento della stima media fornita dalla tecnica CA.

‘raw_sweep’ (che ospita le epoche grezze), ‘stime_mtl’ (che ospita le stime bayesiane), ‘filename’ (contenente il nome del file), ‘channel’ (che indica l’etichetta del canale da cui provengono i dati), ‘fs’ (che indica la frequenza di campionamento), ‘t_raw’ e ‘t_est’ (rispettivamente assi temporali per epoche grezze e stime), e infine ‘ylim’ (delimitatore per l’asse delle ordinate coincidente con l’ampiezza massima in valore assoluto assunta dalle epoche del dataset). Il tasto ‘Help’ consente infine di accedere ad una nuova finestra contenente le istruzioni per l’utilizzo del tool e le indicazioni operative di carattere generale.

5.5 Aggiornamento delle stime

Al contrario dell’average calcolato a posteriori, le curve medie stimate nel contesto bayesiano, per essere aggiornate, necessitano di un nuovo utilizzo dell’algoritmo di stima. Ciò comporta un costo computazionale notevole: per far fronte alle rimozioni delle epoche considerate dal clinico non significative vengono infatti ricalcolati i parametri di regolarizzazione e le nuove stime single-trial.

Nel pannello ‘Remove Options’ sono presenti i checkboxes che consentono di aggiornare le stime (essi sono inutilizzabili finchè non viene operata la prima eliminazione). Quando il clinico decide che non sono necessarie ulteriori rimozioni può utilizzare la componente grafica ‘Stop Sweeps’removal’ e procedere in cascata all’update delle stime. Se una volta operato l’aggiornamento ci si rende conto che nel dataset sono presenti ancora degli outliers si può agevolmente ripetere il ciclo remove-update.

Capitolo 6

GUI per l'analisi delle latenze

L'interfaccia grafica dedicata alla peak-detection è sistemata nel menu 'Tools' di EEGLAB. Il presente plug-in è stato concepito come naturale proseguimento della parte di stima appena illustrata, e ci si è pertanto limitati alla gestione di dati organizzati come nella struttura denominata 'B2S_erp' (o 'MTL_erp') in output al plug-in B2S (o MTL).

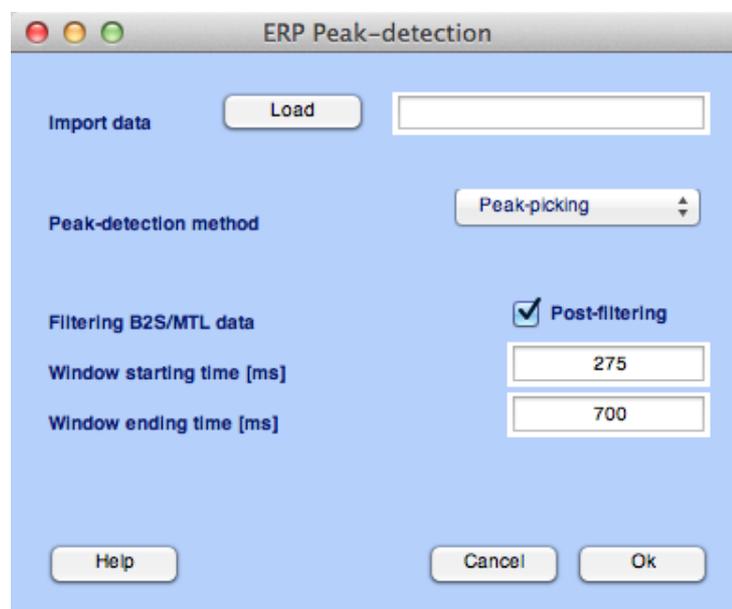


Figura 6.1: Schermata iniziale del plug-in di peak-detection.

L'accesso allo strumento della ricerca del picco è introdotto da una finestra dedicata ai setting di figura 6.1. Oltre al tasto 'Load', che permette di selezionare

la struttura .mat da importare attraverso un comodo browse preimpostato sulla current directory, la schermata dispone anche di un checkbox per comandare il post-filtraggio dei dati al fine di rendere le stime ancora più regolari, di due campi editabili in cui inserire i valori temporali entro cui limitare la ricerca del picco in ogni sweep, e un menu a tendina che consente di scegliere il metodo di peak-detection desiderato tra quelli implementati (scelta che incide sui valori predefiniti relativi alla finestra di ricerca). Analogamente ai plug-in B2S e MTL, sono nuovamente presenti i tasti standard 'Help', 'Cancel' e 'Ok' rispettivamente per accedere alla guida relativa alla function, annullare l'operazione e avviare la ricerca.

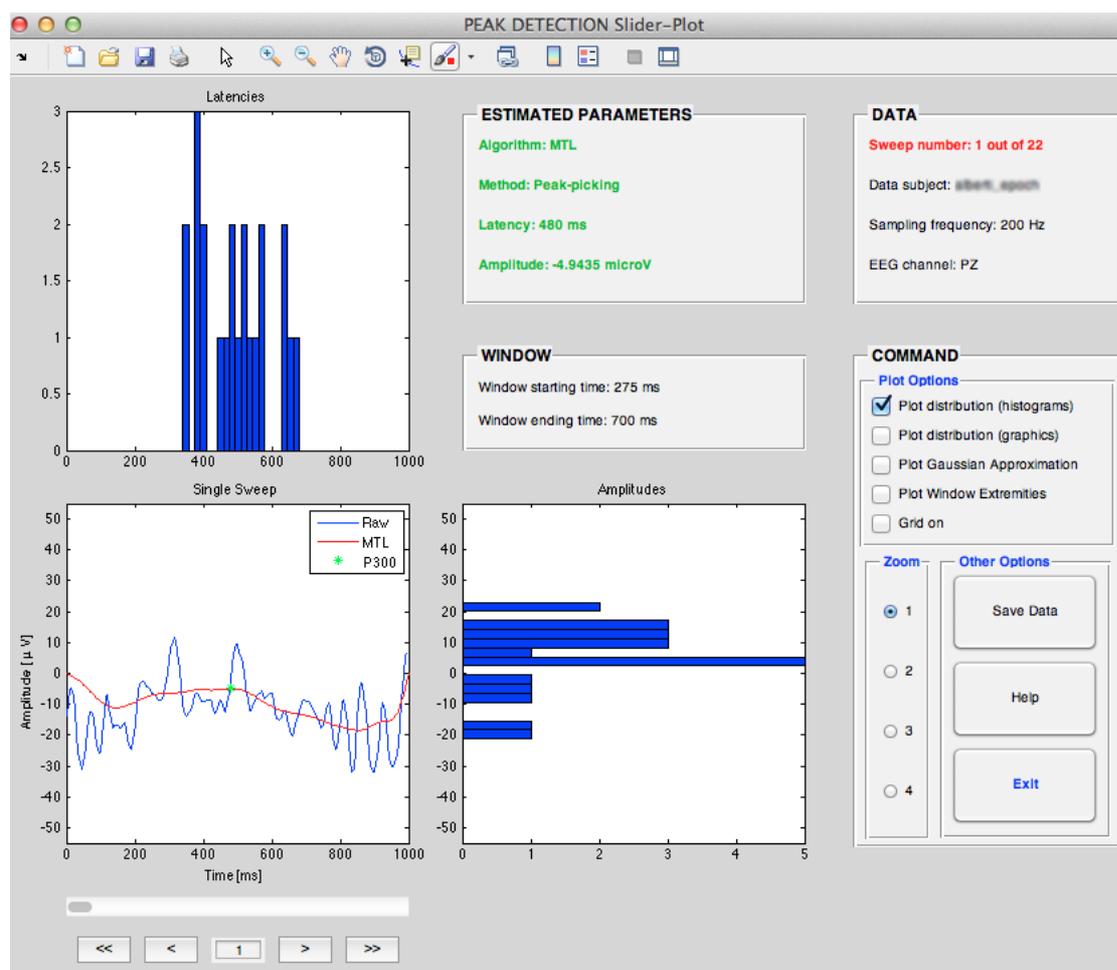


Figura 6.2: Schermata relativa alla presentazione grafica della ricerca del picco su singola sweep.

In figura 6.2 è presentata la finestra a cui conduce il completamento della fase di peak-detection. Il riquadro principale in basso a sinistra ospita il plot del post-stimolo relativo alla singola sweep grezza (in blu), della rispettiva stima bayesiana (in rosso), dei limiti temporali relativi alla finestra di ricerca (in ciano) e del picco identificato mediante il metodo selezionato (asterisco verde). Sotto al grafico sono stati sistemati i medesimi strumenti per lo scorrimento delle singole epoche utilizzati nei plug-in B2S e MTL: una loro descrizione è già stata fornita nella sezione precedente.

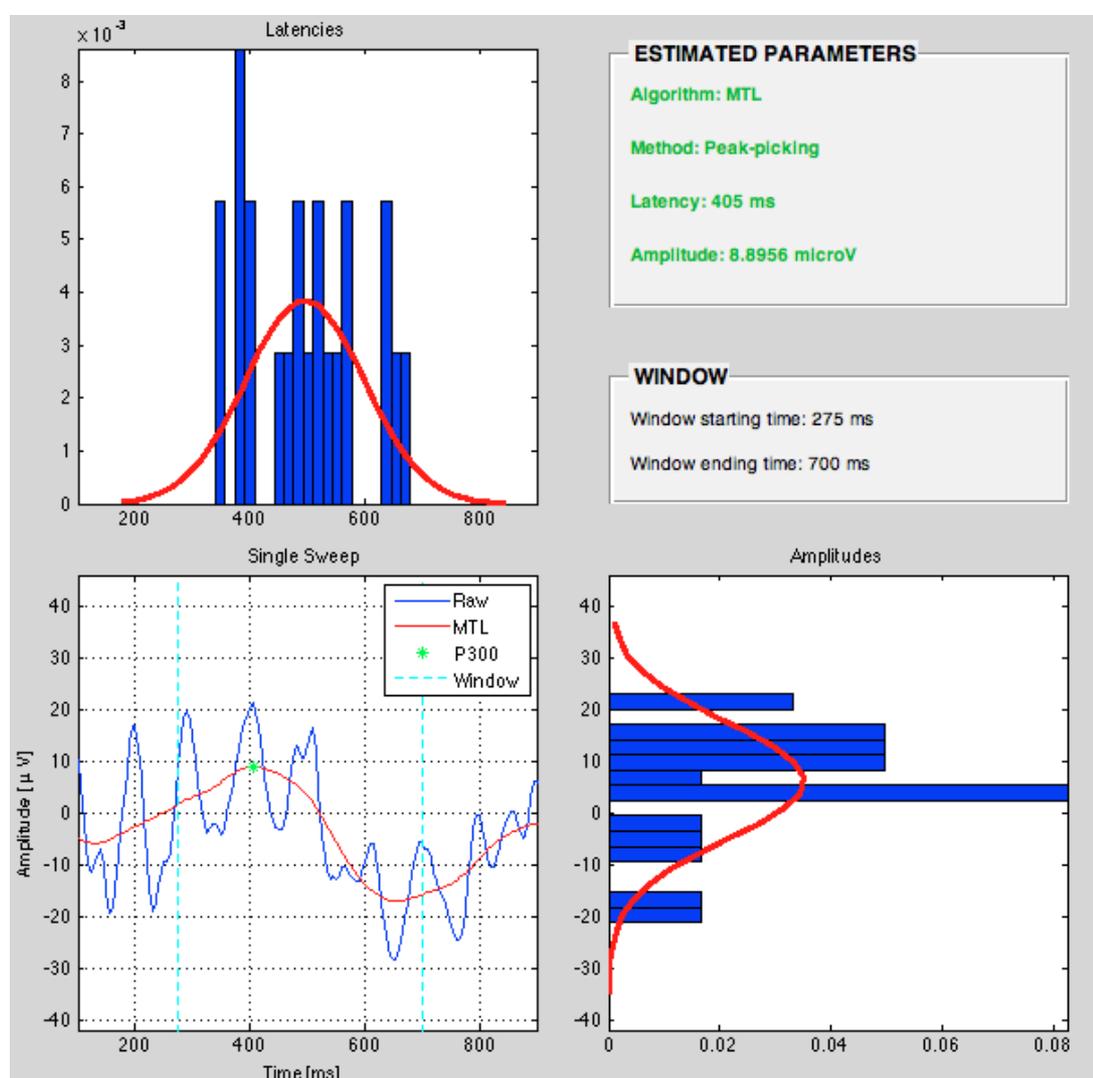


Figura 6.3: *Approssimazione Gaussiana.*

Il tratto più innovativo ed interessante è costituito senza dubbio dalla pre-

senza dei grafici allineati al plot principale in senso orizzontale e verticale. Essi costituiscono un utile strumento grazie al quale l'utente può tenere sotto controllo le distribuzioni di latenze e ampiezze single-trial rilevate nelle stime del dataset caricato. I checkboxes 'Plot Distribution' permettono di passare dagli istogrammi ai grafici di dispersione dei due parametri di interesse, dando modo di porre a confronto il picco corrente con la frequenza di ricorrenza delle sue latenza e ampiezza, oppure con i restanti valori di latenza e ampiezza. L'immediato impatto visivo è di sicura efficacia per facilitare operazioni di analisi quali la classificazione del picco come 'normale' o come 'outlier'.

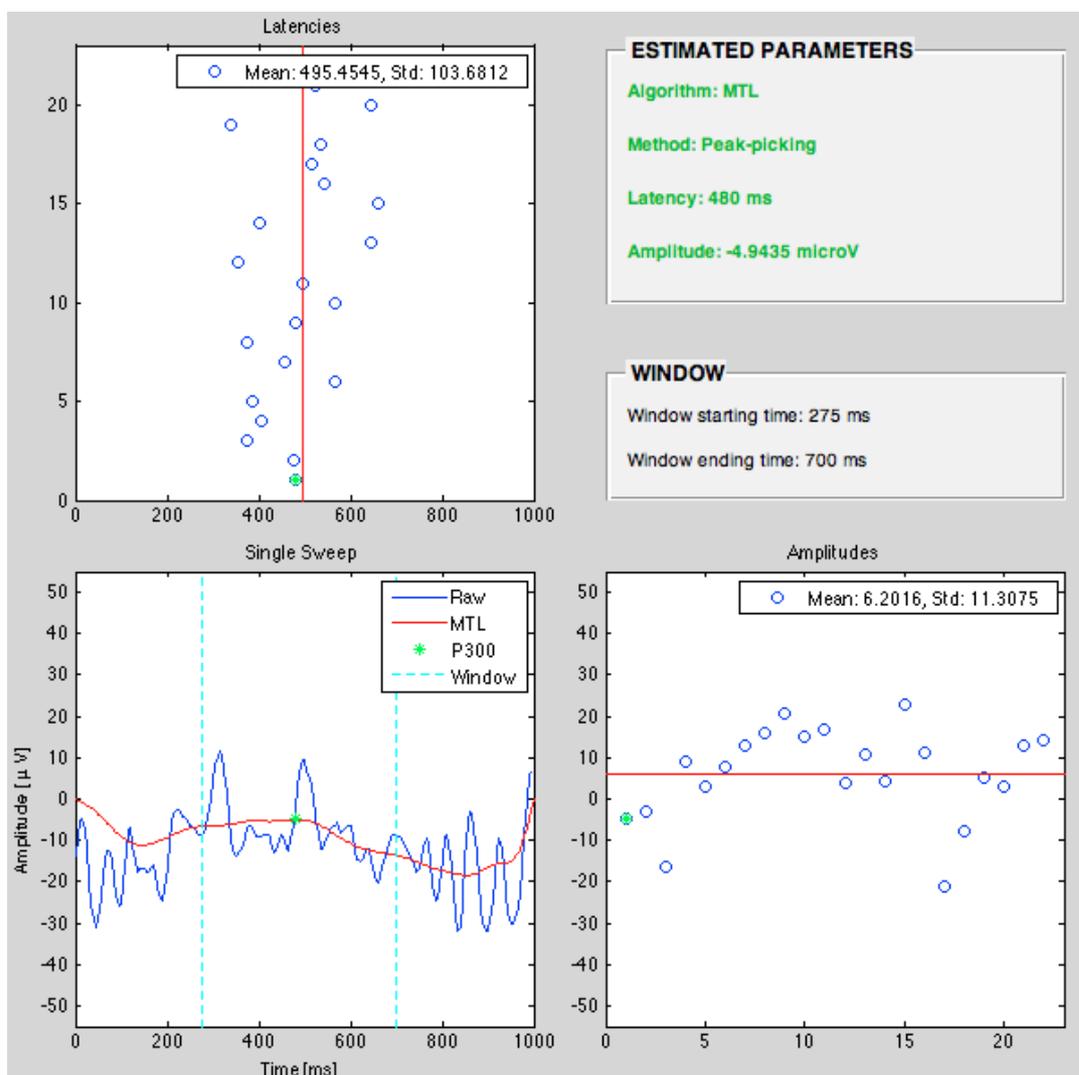


Figura 6.4: Grafici di dispersione.

Nelle Plot Options è stato inoltre inserito un checkbox che permette di visualizzare l'approssimazione degli istogrammi normalizzati con una curva gaussiana.

Le informazioni relative a soggetto, frequenza di campionamento, canale in analisi, numero della sweep corrente, e valori relativi a latenza e ampiezza del picco della componente P300 sono contenute all'interno del pannello 'DATA'.

Come nei tools B2S e MTL, sono infine resi disponibili i tasti 'Help', 'Save Data' ed 'Exit' con proprietà analoghe ai casi precedentemente presentati.

Un pannello verticale destro dedicato allo zoom consente di raggiungere fino a quattro livelli di ingrandimento relativi non solo al plot principale, ma anche ai grafici ad esso correlati, mantenendo gli assi del sistema di visualizzazione allineati. Per esplorare i dati mediante gli strumenti offerti dalla toolbar classica di MATLAB®[®], è sufficiente selezionare il tool desiderato ed applicarlo al grafico di interesse.

Capitolo 7

Altri aspetti implementativi

Le singole epoche dell'EEG grezzo registrato su ogni soggetto sono state, nell'ordine:

- sottoposte ad una analisi atta all'esclusione delle sweep corrotte da artefatti oculari;
- date in input all'algoritmo di stima bayesiana;
- ulteriormente regolarizzate mediante l'applicazione di un filtro passa-basso e la sottrazione sistematica della media dal post-stimolo;
- sottoposte all'algoritmo di peak-detection.

7.1 Artifact rejection

Per la rimozione delle sweep contenenti artefatti, è stato implementato un algoritmo automatico in grado di rilevare nella fattispecie i disturbi causati da movimenti oculari. I tracciati EEG ed EOG co-registrato sono stati oggetto di un'analisi parallela volta alla precisa individuazione di eventuali artefatti oculari. Benché le comuni procedure siano basate esclusivamente su un rilevamento a soglia, nel presente caso è stata strutturata una tecnica che indaga diverse caratteristiche:

- sogliatura: applicazione di una threshold pari a $100\mu V$ alla porzione post-stimolo del tracciato EOG, la quale rappresenta il livello di accettabilità

dei movimenti oculari. Nel caso in cui il soggetto dia adito ad artefatti oculari rilevanti in corrispondenza del segmento temporale post-stimolo, una o più deflessioni dell'EOG si dimostreranno superare la soglia impostata. La sweep relativa all'intervallo temporale corrispondente risulterà dunque corrotta dal disturbo e verrà rimossa dal dataset. Eventuali artefatti circoscritti al pre-stimolo risultano d'altro canto accettati in quanto ritenuti non compromettenti la componente P300 di interesse;

- prodotto scalare: viene effettuato il calcolo del prodotto scalare tra i segnali EEG registrati in Fz e Cz, e l'elettrooculogramma. Esso è interpretabile infatti come funzione di correlazione tra due segnali x e y , è cioè in grado di misurarne la somiglianza. Il valore critico fissato in questo caso coincide con il valore che assume la funzione coseno a 45° : il presente criterio esclude le sweep il cui prodotto scalare con il tracciato EOG supera il valore di ≈ 0.707 . Quando infatti l'angolo tra i due segnali vale 90° , il prodotto scalare indica una correlazione nulla dato che i segnali risultano ortogonali, mentre quando l'angolo tra i due segnali vale 0° , il prodotto scalare presenta valore massimo e pari ad uno, indicando una correlazione massima tra i due segnali che risultano infatti allineati. Il prodotto scalare non rappresenta tuttavia un ottimo indicatore di correlazione perché il profondamente influenzato dall'eventuale ritardo di un segnale rispetto all'altro. Basti pensare a come, ponendo $x(t) = \sin 2\pi ft$ e $y(t) = \cos 2\pi ft$, si ottenga $x \bullet y = 0$ nonostante la sinusoidale e cosinusoidale siano il medesimo segnale in due versioni traslate nel tempo. E' quindi necessario ricorrere ad un altro indicatore di somiglianza che contempli l'eventuale presenza di ritardi (sfasamenti) tra i due segnali;
- correlazione: viene calcolato infine l'indice di correlazione di Pearson tra il segnale EOG e i tracciati EEG registrati nei canali Fz e Cz, verosimilmente inficiati in misura maggiore dall'attività oculare rispetto ai restanti. E' infatti noto come l'attività elettrica e muscolare cagionate dai cosiddetti blink oculari si propaghino ai canali di registrazione in modo inversamente proporzionale alla loro distanza: mentre in Fz tale attività si manifesterà in modo consistente, nei canali Pz e Oz il suo effetto sarà tangibilmente atte-

nuato. Il coefficiente di correlazione ricavato esprime dunque la linearità tra la covarianza delle due variabili x, y (ovvero il segnale EOG e il segnale EEG di Fz o Cz) e il prodotto tra le rispettive deviazioni standard. Il coefficiente di correlazione ρ_{xy} risulta sempre compreso tra -1 e 1 , e più in dettaglio: se $\rho_{xy} > 0$ le due variabili x e y si dicono correlate positivamente (o direttamente), se $\rho_{xy} = 0$ esse si dicono incorrelate, mentre se $\rho_{xy} < 0$ esse si dicono correlate negativamente (o inversamente). Per garantire l'esclusione delle sole sweep in correlazione forte con il tracciato EOG, si è posto come valore soglia $\rho_{xy} = 0.75$ con un p-value pari a 0.05 . Vengono dunque rigettate esclusivamente le epoche caratterizzate da un coefficiente di correlazione maggiore di 0.75 con un livello di significatività del 5% , considerando come detto i canali frontale mediano e centrale mediano.

I tre metodi sono stati applicati singolarmente e separatamente, ma la loro azione è stata resa sinergica poiché l'algoritmo provvede a rigettare una determinata epoca solo quando almeno due metodi su tre la classifichino come corrotta da artefatto oculare.

7.2 Approccio bayesiano

Le epoche target "sopravvissute" allo step di reiezione degli artefatti oculari sono state sottoposte alla procedura di filtraggio bayesiano descritta in precedenza. Considerato infatti che l'obiettivo ultimo è l'individuazione di latenza e ampiezza della componente P300, si è ritenuto appropriato e conveniente applicare l'algoritmo di stima alle sole sweep che si riferiscono a stimoli target. E' stato ottenuto in questo modo un dataset di tracce ERPs corrispondenti alle stime bayesiane dei singoli trial.

7.3 Post-filtraggio e rimozione della media

Le stime prodotte dall'algoritmo bayesiano presentano, solitamente, ancora un carattere irregolare e frastagliato. Per regolarizzarne l'andamento e rendere più

agevole la ricerca del picco di interesse, esse sono state sottoposte ad una ulteriore fase di filtraggio digitale.

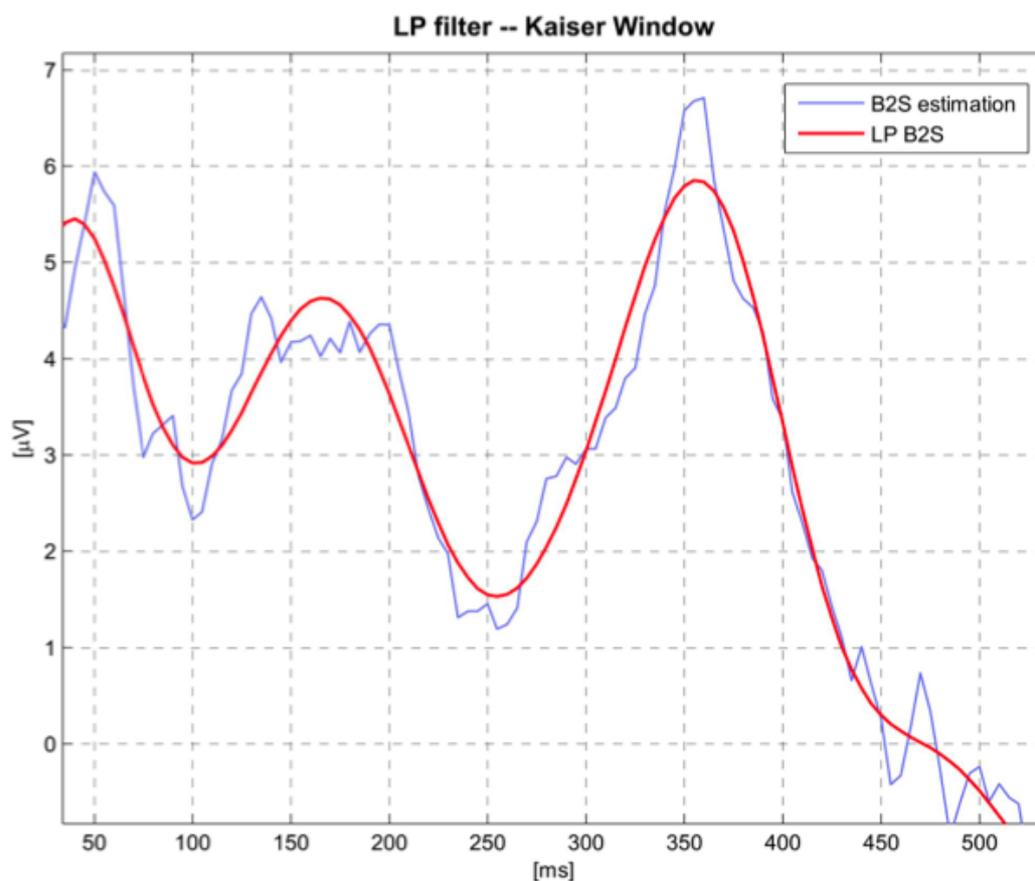


Figura 7.1: Particolare ingrandito di una singola sweep stimata mediante algoritmo B2S (in pervinca) e sottoposta a filtraggio passa-basso (in rosso). Le asperità residue appaiono smussate e l'andamento risulta regolare e fedele alla stima originale.

Più precisamente è stato adottato un filtro FIR¹ progettato mediante finestra di Kaiser. Il filtro passa-basso è stato caratterizzato con una banda passante compresa tra 0 e 8 Hz, una banda oscura per valori superiori a 15 Hz e una conse-

¹Un filtro Finite Impulse Response (FIR) è un sistema lineare tempo-invariante (LTI) causale con risposta finita all'impulso (diversamente dai filtri Infinite Impulse Response, IIR). Ogni filtro FIR è causale e stabile, la sua funzione di trasferimento risulta essere un polinomio in z^{-1} (ovvero esiste un solo polo in $z = 0$, interno al cerchio di raggio unitario), e la sua fase può essere lineare.

guente banda di transizione compresa tra 8 e 15 Hz. Il ripple (oscillazioni residue indesiderate del modulo della funzione di trasferimento del filtro) consentito in banda passante e l'attenuazione imposta in banda oscura sono stati settati rispettivamente a 0.05 e 0.01. L'uso della funzione MATLAB® `filtfilt.m` ha assicurato un doppio passaggio del filtro, da destra a sinistra e viceversa. E' stato in questo modo possibile realizzare un filtraggio a fase zero in grado di compensare eventuali ritardi in fase dovuti al processo in quanto tale. Il risultato dell'operazione di filtraggio è illustrato in figura 7.1.

La scelta di sottrarre la media dalle stime, e rendere dunque il post-stimolo a media nulla, si è rivelata necessaria per centrare in ampiezza gli ERPs e renderli paragonabili tra loro. Essendo infatti immersi in un'attività di fondo talvolta irregolare, i potenziali evento-correlati di alcune epoche sono risultati evidentemente traslati verso l'alto o verso il basso. La rimozione di una componente continua nel tempo quale è la media del post-stimolo, non comporta chiaramente il sacrificio di informazioni riguardanti la fisionomia dei potenziali evento-correlati, trattandosi di una componente a frequenza nulla. Considerando che si è adottata una tecnica di rilevamento picco-baseline anziché picco-picco, le tracce sono state dunque "centrate" proprio per annullare l'influenza del fattore offset ed evitare che esso infici irrimediabilmente la misura dell'ampiezza.

7.4 Aspetti computazionali

Gli algoritmi sviluppati in questo lavoro di tesi devono risultare efficaci anche dal punto di vista dell'onere computazionale. Bisogna dare la possibilità all'utente/clinico di ottenere gli output con dei tempi relativamente ridotti: proprio per questo motivo è stata studiata una procedura ad hoc che permette di sottocampionare i dati nel caso essi risultino in numero eccessivo. Dopo alcune prove, si è deciso che ogni sweep non deve contenere più di 150-200 campioni (il fattore di downsampling viene calcolato di default tenendo presente questo vincolo). Vengono quindi presentati alcuni esempi significativi (tabella 7.1) che diano un'idea generale della velocità con la quale l'algoritmo MTL giunge a compimento.

Nr. subject	Nr. sweep	fs	Downsampling step	time
1	49	200 Hz	2	42 s
2	22	200 Hz	2	19 s
2	22	200 Hz	3	12 s
3	21	200 Hz	2	18 s
4	44	200 Hz	2	32 s
4	44	200 Hz	4	20 s
5	38	200 Hz	2	28 s
6	41	1000 Hz	10	67 s
7	29	200 Hz	2	24 s

Tabella 7.1: *Tempi computazionali - Esempi*

7.5 Utilizzo su diversi sistemi operativi

L'interfaccia grafica è stata costruita su un computer facente uso del sistema operativo Mac OS. Sono state eseguite delle prove su altri calcolatori dotati di Windows o Linux e i risultati nella visualizzazione grafica sono pressochè identici. I test non sono stati molteplici ma tendenzialmente si può affermare che sui pc con formato 16:9 dotati di sistemi operativi diversi non si presentano problemi di risoluzione o eventuali necessità di normalizzazione. In particolare, le variabili geometriche delle varie componenti grafiche sono state settate con l'opzione 'unit' - 'pixel' (il software MATLAB[®] determina automaticamente la dimensione dello schermo e posiziona di conseguenza la figura) ad eccezione dei pannelli che sono stati normalizzati in base all'estensione della figura principale.

Conclusioni e sviluppi futuri

Lo sviluppo di una GUI per l'estrazione single-sweep apre nuovi orizzonti in ambito clinico dove c'è una scarsa disponibilità di software flessibili e automatici: il tool risponde dunque alla necessità di rendere il presente tipo di analisi alla portata di tutti. Sebbene la sua interfaccia grafica sia senza dubbio suscettibile di miglioramenti ed ampliamenti (è bene ricordare che la logica open-source di EEGLAB consente a chiunque di apportare eventuali cambiamenti alle GUI progettate in questa sede), esso costituisce a tutti gli effetti uno strumento flessibile, usabile, funzionale e pronto a confrontarsi con le sfide offerte dal contesto clinico.

L'integrazione al toolbox EEGLAB, ampiamente riconosciuto nell'ambiente elettrofisiologico, permette al clinico di operare in un contesto già noto favorendo inoltre l'affermazione delle tecniche single-trial i cui vantaggi non sono ancora pienamente riconosciuti. La potenza di tale indagine risulta decisamente maggiore, sia dal punto di vista dell'effettiva significatività del campione, sia nell'ottica della profondità di esplorazione dei dati stessi. In tal senso, mentre viene meno la necessità di collezionare un dataset estremamente ampio, sottoponendo il soggetto a lunghi e stressanti task cognitivi, si acquisisce la facoltà di esaminare l'evoluzione temporale di parametri tutt'altro che stazionari quali il fattore attentivo e di concentrazione.

Con la convinzione che questo software possa risultare estremamente utile nell'ambiente clinico, è stata infine avviata una fase di β -test in collaborazione con il Dipartimento di Ingegneria dell'Informazione, il Dipartimento di Psicologia Generale e il Dipartimento di Medicina Clinica e Sperimentale.

Appendice A

Stima single-trial mediante modelli ARX

A.1 Un'alternativa all'approccio bayesiano

Un'alternativa all'approccio bayesiano è rappresentata da una tecnica parametrica altrettanto sofisticata e versatile, che prevede una descrizione non funzionale del potenziale single-trial: quella dei modelli ARX [46][47][48][49][50]. Per la stima della singola risposta, si descrive il segnale misurato con un modello dinamico lineare autoregressivo e con ingresso esogeno. I modelli ARX sono una sottoclasse di una più generale famiglia di processi stocastici ARMAX per i quali il segnale $s(k)$ è visto come somma di una componente deterministica $ERP(k)$ e di una componente aleatoria $EEG(k)$.

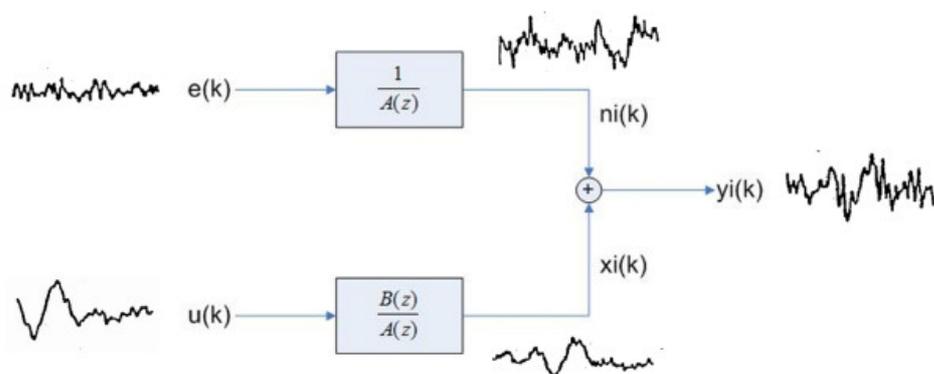


Figura A.1: Modello ARX del segnale

Confrontando la notazione di figura A.1 con quella precedentemente espressa risulta in particolare $x(k) = ERP(k)$ (componente deterministica) e $n(k) = EEG(k)$ (componente aleatoria).

In pratica, spiegando accuratamente il modello:

- la componente aleatoria $EEG(k)$ si ottiene mediante filtraggio AR di ordine n di un processo di rumore bianco $e(k)$ a media nulla e varianza nota;
- la componente deterministica $ERP(k)$ è data da un filtraggio ARMA di ordini (n, m) di un segnale deterministico $u(k)$ che funge da riferimento, eventualmente ritardato o anticipato di d istanti temporali;
- il segnale registrato $s(k)$ (in figura A.1 rappresentato da $y(k)$) è visto come somma di una componente deterministica $ERP(k)$ e di una componente aleatoria $EEG(k)$;

Descrivendo quanto appena spiegato nel dominio del tempo si ha:

$$EEG(k) = - \sum_{i=1}^n a_i EEG(k-i) + e(k) \quad (A.1)$$

$$ERP(k) = - \sum_{i=1}^n a_i ERP(k-i) + \sum_{j=d}^{m+d-1} b_j u(k-j) \quad (A.2)$$

$$s(k) = ERP(k) + EEG(k) \quad (A.3)$$

Sostituendo le prime due espressioni nell'ultima equazione si ottiene, per la descrizione del modello, una forma più generale e compatta:

$$s(k) = - \sum_{i=1}^n a_i s(k-i) + \sum_{j=d}^{m+d-1} b_j u(k-j) + e(k) \quad (A.4)$$

Nel dominio della trasformata zeta, posto che z^{-1} sia l'operatore di ritardo, dopo semplici passaggi algebrici si ottiene:

$$S(z)(1 + \sum_{i=1}^n a_i z^{-i}) = U(z)(\sum_{j=d}^{m+d-1} b_j z^{-j}) + E(z) \quad (A.5)$$

che può essere riscritta nella forma

$$S(z)A(z) = U(z)B(z) + E(z) \quad (\text{A.6})$$

Per ogni epoca è identificato un modello ARX con parametri diversi, permettendo così di rilassare il vincolo di tempo-invarianza della risposta evocata. Identificazione e validazione dei modelli sono passi fondamentali per la stima delle singole risposte.

Sinteticamente gli aspetti più importanti sono:

- la selezione del numero ottimale di parametri della parte AR ed MA;
- la scelta del segnale di riferimento.

In particolare, l'ingresso deterministico del sistema deve descrivere le caratteristiche principali delle singole risposte cercate, la cui variabilità è limitata dalla funzione di trasferimento del filtro ARMA $\frac{B(z)}{A(z)}$. Proprio per questo $u(k)$ dovrebbe fornire una descrizione il più possibile accurata della componente ricercata all'interno del segnale registrato originario.

In genere sono possibili due scelte:

- se la finalità è uno studio a posteriori della variabilità delle risposte, si può utilizzare la media di tutte le sweep a disposizione, che è il miglior riferimento che l'averaging può fornire, a patto che sia verificata la riproducibilità del potenziale in realizzazioni successive. In pratica il segnale d'ingresso risulterà:

$$u(t) = \frac{1}{N_{max}} \sum_{i=1}^{N_{max}} s_i(t) \quad (\text{A.7})$$

- l'altra possibilità è considerare come segnale di riferimento la media delle epoche acquisite fino a quell'istante (identificazione on-line), oppure la media di n segnali acquisiti per ultimi, questo per poter avere traccia nel riferimento di quella che potrebbe essere una lenta variazione del potenziale evocato:

$$u_{ol}(t) = \frac{1}{N_{ol}} \sum_{i=1}^{N_{ol}} s_i(t) \quad (\text{A.8})$$

E' fin troppo evidente il potenziale “pericolo” derivato dalla registrazione di un numero limitato di sweep: se il segnale EEG di fondo risulterà troppo “rumoroso” anche la qualità del riferimento ne risentirà.

A.2 Aspetti implementativi

A livello di codice MATLAB[®], per la stima del ritardo tra il riferimento esogeno e la raw-sweep da analizzare si può procedere come segue:

- creazione di un oggetto

```
data = iddata(sweep(:,i),riferimento,1/fs);
```

- stima del ritardo per ogni coppia di ordini del modello:

```
nk(n,m) = delayest(data,n,m);
```

Per poter eseguire l'ultima istruzione è necessario stabilire un ordine massimo da analizzare, sia per la parte autoregressiva sia per quella a media mobile. Operare una scelta vantaggiosa non è semplice poichè:

- modelli con ordini alti seguono in maniera soddisfacente i dati e offrono stime migliori dell'ampiezza dei picchi;
- analizzare solo modelli con ordini bassi (ad esempio minori di 10) consente di avere meno onere computazionale, si verifica però il fenomeno dell'over-smoothing: vengono stimati valori di ampiezza dei picchi più ridotti e in molti casi insoddisfacenti.

La possibile soluzione potrebbe essere data dalla scelta di una soglia minima e di una soglia massima.

La parte più delicata risulta la stima dell'ordine ottimo per la parte AR e per la parte MA del modello. A livello di codice MATLAB[®] si può procedere in vari modi.

Una strategia ottimale prevede che, per ogni coppia (n, m) e il corrispettivo valore del ritardo nk stimato si calcolino i coefficienti del modello e si valuti la loss function inserendo da tastiera

```
mi = arx(data, [n,m,nk]);  
matricelossfcn(n,m) = mi.EstimationInfo.LossFcn;
```

Oppure si può sfruttare il seguente codice:

```
V = arxstruc(data,data, struc(1:nmax,1:mmax,1:nkmax));  
best = selstruc(V,0);  
mi = arx(data, [best(1) best(2) best(3)]);
```

E' evidente che con il secondo metodo la function *arx.m* viene utilizzata una volta per tutte, solo dopo aver stimato gli ordini ottimi. Tuttavia restano in sospenso alcuni aspetti critici:

- il secondo argomento della funzione *arxstruc.m* indica il validation set la cui scelta non è mai banale (in questo caso test set e validation set coincidono ma non si può affermare con certezza che sia una decisione algoritmica vantaggiosa);
- per come è impostata la funzione *struc.m* il programmatore è “costretto” a valutare la coppia (n, m) per ogni nk compreso tra 1 a $nkmax$ (elevata complessità);
- la scelta del criterio di parsimonia come secondo argomento della function *selstruc.m*.

Una volta calcolati gli ordini ottimi si stima l'ERP con la funzione *filter* di MATLAB®:

```
stima(:,i) = filter(mi.b,mi.a,riferimento);
```


Appendice B

Singular Value Decomposition

In algebra lineare, la decomposizione ai valori singolari (o SVD, Singular Value Decomposition) è una particolare fattorizzazione basata sull'uso di autovalori e autovettori, utilizzata per produrre un'approssimazione della matrice originaria con minor rango.

Con lo smoothing/regolarizzazione, per trovare γ che soddisfa un determinato criterio bisogna calcolare per diverse volte la soluzione. E' evidente che questo problema aumenta notevolmente il carico computazionale, infatti invertire una matrice di dimensione n è possibile con vari algoritmi che richiedono una complessità $o(n^3)$, affrontata per ogni γ tentativo. La strada che passa per l'inversione di matrice è numericamente inefficiente quindi risulta di fondamentale importanza studiare algoritmi ad hoc per la regolarizzazione.

Il modello delle misure, nel caso in esame, è

$$y = Gu + v \tag{B.1}$$

dove G è una matrice di dimensione $n \times N$ (l'ipotesi fondamentale è che si lavori con la virtual grid).

Come noto il valore calcolato dallo smoother è

$$\hat{u} = (G^T B^{-1} G + \gamma F^T F)^{-1} G^T B^{-1} y \tag{B.2}$$

dove $B = (A^T A)^{-1}$ nel caso dell'algoritmo Bayesian Two Steps (B2S).

Si può procedere sfruttando un cambio di coordinate con conseguente diagonalizzazione. Viene definita la matrice $H = B^{-1/2}GF^{-1}$ di dimensione $n \times N$ e vengono calcolate le matrici U, V (quadrato unitarie di dimensione rispettivamente n e N) e D che definiscono la sua Singular Value Decomposition.

Deve risultare

$$U^T H V = D \quad (\text{B.3})$$

Il punto cruciale è che la SVD, pur avendo complessità $o(n^3)$, viene eseguita una volta sola e non dipende dal parametro di regolarizzazione γ .

Infine viene effettuato il cambio di coordinate $\xi = U^T B^{-1/2}y$ nel dominio delle misure che consente di diagonalizzare il problema.

La matrice D ha la forma seguente:

$$D = \begin{pmatrix} d_1 & 0 & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & d_2 & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & d_{n-1} & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & 0 & d_n & 0 & \dots & 0 \end{pmatrix} \quad (\text{B.4})$$

La stima bayesiana è data da:

$$\hat{\eta} = (D^T D + \gamma^0 I_N)^{-1} D^T \xi \quad (\text{B.5})$$

Il calcolo della stima e degli intervalli di confidenza, nelle nuove coordinate, non richiede di manipolare alcuna matrice (tutte operazioni scalari con complessità $o(n)$).

La stima quindi può essere riscritta in notazione scalare nel modo seguente:

$$\hat{\eta}_i = \frac{d_i}{d_i^2 + \gamma} \xi_i \quad (\text{B.6})$$

con $i = 1, \dots, n$ e $\hat{\eta}_i = 0$ per $i = n+1, \dots, N$ dove i dati non sono “osservabili” dalle misure.

Solo dopo aver trovato il γ ottimo si ritorna quindi alle coordinate originarie tramite un calcolo che richiede $o(N^2 n)$ operazioni:

$$\hat{u} = F^{-1}V\hat{\eta} \tag{B.7}$$

In pratica, per ognuna delle N righe di V bisogna eseguire n prodotti e moltiplicare infine per F^{-1} (si può fare in $o(N)$ se si lavora in un ambito operativo in cui è possibile applicare le equazioni alle differenze).

A livello di programmazione MATLAB®[®], la Singular Value Decomposition viene implementata con la funzione *svd.m* mentre, quando si utilizzano le equazioni alle differenze, si sfrutta la function *filter.m* (questo tipo di operazione può risultare decisiva non solo nel “ritorno” alle coordinate originarie ma anche nel calcolo delle matrici F e H).

Appendice C

Scheletro della GUI

In seguito viene riportato il codice MATLAB® che consente di costruire graficamente e geometricamente l'interfaccia grafica relativa a MTL (si riferisce ad una versione del software precedente a quella presentata nel capitolo 4). Il codice definisce la GUI da un punto di vista unicamente strutturale¹.

```
S.fh = figure('units','pixels',...  
'position',[1030 600 1030 600],...  
'menubar','none',...  
'name','MTL Slider-Plot',...  
'numbertitle','off',...  
'resize','off');
```

```
set(S.fh,'toolbar','figure');  
(S.fh,'center');  
% per zoomare movegui
```

```
S.ax = axes('unit','pix',...  
'position',[310 130 422 300]);
```

¹Le funzioni che descrivono le varie azioni che la GUI è in grado di compiere non vengono riportate.

```
S.ax2 = axes('unit','pix',...
'position',[312 470 150 100]);

S.ar = uipanel('Title','PRE-STIMULUS MODEL','FontSize',12,...
'fontweight','bold',...
'Position',[.03 .06 .22 .9]);

S.ax4 = axes('parent',S.ar,...
'units','normal',...
'position',[0.1 0.27 0.8 0.4]);

S.ar_data = uipanel('parent',S.ar,...
'Title','AR Model Identification',...
'ForegroundColor',[204/255, 204/255, 0],...
'fontweight','bold',...
'position',[0.015 0.72 0.97 0.27]);

S.ar1 = uicontrol('parent',S.ar_data,...
'style','text',...
'string','Criterion',...
'HorizontalAlignment','left',...
'position',[10 107 200 15]);

S.ar2 = uicontrol('parent',S.ar_data,...
'style','text',...
'string','Maximum Order: ',...
'HorizontalAlignment','left',...
'position',[10 83 200 15]);

S.ar3 = uicontrol('parent',S.ar_data,...
'style','text',...
'string','Best Order: ',...
```

```
'HorizontalAlignment','left',...
'position',[10 59 200 15]);

S.ar4 = uicontrol('parent',S.ar_data,...
'style','text',...
'string','Estimated variance: ',...
'HorizontalAlignment','left',...
'position',[10 35 200 15]);

S.ar5 = uicontrol('parent',S.ar_data,...
'style','text',...
'string','Model Stability: ',...
'HorizontalAlignment','left',...
'position',[10 11 200 15]);

S.ar_coef = uipanel('parent',S.ar,...
'Title','AR Model Coefficients',...
'ForegroundColor',[204/255, 204/255, 0],...
'fontweight','bold',...
'position',[0.015 0.01 0.97 0.19]);

S.ar6 = uicontrol('parent',S.ar_coef,...
'style','listbox',...
'string','',...
'HorizontalAlignment','left',...
'position',[5 5 205 80]);

S.hp = uipanel('Title','COMMAND','FontSize',12,...
'fontweight','bold',...
'Position',[.755 .06 .215 .67]);

S.hpp = uipanel('parent',S.hp,...
```

```
'Title','Other Options',...
'ForegroundColor','blue',...
'fontweight','bold',...
'position',[0.015 0.18 0.97 0.37]);

S.hpp2 = uipanel('parent',S.hp,...
'Title','Plot Options',...
'ForegroundColor','blue',...
'fontweight','bold',...
'position',[0.015 0.57 0.97 0.42]);

S.hpp3 = uipanel('parent',S.hp,...
'Title','Zoom',...
'ForegroundColor','blue',...
'fontweight','bold',...
'position',[0.015 0.015 0.97 0.155]);

S.lm = uicontrol('parent',S.hpp3,...
'style','text',...
'string','Y-Axis Limit: ',...
'HorizontalAlignment','left',...
'position',[15 13 80 20]);

S.lm2 = uicontrol('parent',S.hpp3,...
'style','edit',...
'string',' ',...
'HorizontalAlignment','left',...
'position',[100 14 50 25]);

S.c1 = uicontrol('parent',S.hpp2,...
'style','checkbox',...
'string','Plot Post-Stimulus Only',...
```

```
'position',[5 125 200 20]);

S.c2 = uicontrol('parent',S.hpp2,...
'style','checkbox',...
'string','Plot Average Raw-Sweep',...
'position',[5 101 200 20]);

S.c3 = uicontrol('parent',S.hpp2,...
'style','checkbox',...
'string','Plot Average MTL-Estimation',...
'position',[5 77 200 20]);

S.c4 = uicontrol('parent',S.hpp2,...
'style','checkbox',...
'string','Plot All-Sweeps',...
'position',[5 53 150 20]);

S.c5 = uicontrol('parent',S.hpp2,...
'style','checkbox',...
'string','Grid on',...
'position',[5 5 150 20]);

S.c6 = uicontrol('parent',S.hpp2,...
'style','checkbox',...
'string','Pop. Avg (MTL) vs. Conv. Avg',...
'position',[5 29 200 20]);

S.p1 = uicontrol('parent',S.hp,...
'string','Remove Sweep',...
'position',[10 140 97 60]);

S.p2 = uicontrol('parent',S.hp,...
```

```
'string','Save Data',...
'position',[111 140 97 60]);

S.p7 = uicontrol('parent',S.hp,...
'string','Help',...
'position',[10 77 97 60]);

S.p8 = uicontrol('parent',S.hp,...
'string','Exit',...
'ForegroundColor','blue',...
'fontweight','bold',...
'position',[111 77 97 60]);

S.hp2 = uipanel('Title','REGULARIZATION','FontSize',12,...
'fontweight','bold',...
'Position',[.755 .78 .215 .18]);

S.t1 = uicontrol('parent',S.hp2,...
'style','text',...
'HorizontalAlignment','left',...
'string','criterion',...
'ForegroundColor',[0, 0.7, 0],...
'fontweight','bold',...
'position',[10 66 200 20]);

S.t2 = uicontrol('parent',S.hp2,...
'style','text',...
'HorizontalAlignment','left',...
'string','Lambda: ',...
'position',[10 37 200 20]);

S.t3 = uicontrol('parent',S.hp2,...
```

```
'style','text',...
'HorizontalAlignment','left',...
'string','LambdaTilde: ',...
'position',[10 8 200 20]);

S.hp3 = uipanel('Title','DATA','FontSize',12,...
'fontweight','bold',...
'Position',[.482 .78 .23 .18]);

S.text1 = uicontrol('parent',S.hp3,...
'style','text',...
'ForegroundColor','red',...
'fontweight','bold',...
'HorizontalAlignment','left',...
'string','Sweep number: ',...
'position',[10 70 200 15]);

S.text2 = uicontrol('parent',S.hp3,...
'style','text',...
'HorizontalAlignment','left',...
'string','Subject: ',...
'position',[10 50 200 15]);

S.text3 = uicontrol('parent',S.hp3,...
'style','text',...
'HorizontalAlignment','left',...
'string','Sampling frequency: ',...
'position',[10 30 200 15]);

S.text4 = uicontrol('parent',S.hp3,...
'style','text',...
'HorizontalAlignment','left',...
```

```
'string','EEG channel: ',...
'position',[10 10 200 15]);

S.sl = uicontrol('style','slide',...
'unit','pix',...
'position',[310 65 422 20],...
'min',1,'max',25,'val',1,...
'sliderstep',[1/(25-1) 1/(25-1)]);

S.e1 = uicontrol('style','edit',...
'string',1,...
'position',[492 35 500 20]);

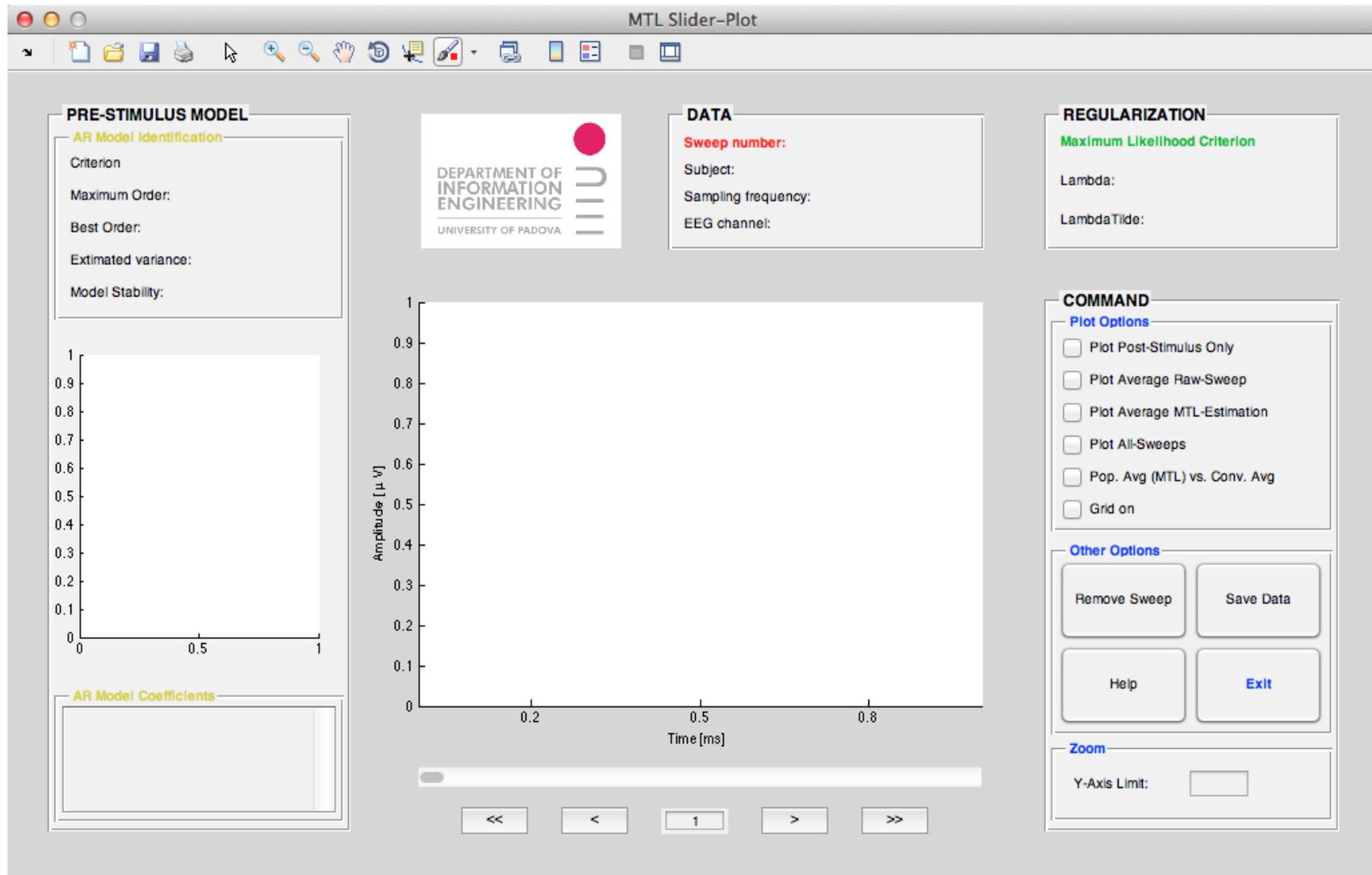
S.p3 = uicontrol('style','pushbutton',...
'string','>',...
'position',[567 35 500 20]);

S.p4 = uicontrol('style','pushbutton',...
'string','<',...
'position',[417 35 500 20]);

S.p5 = uicontrol('style','pushbutton',...
'string','>>',...
'position',[642 35 500 20]);

S.p6 = uicontrol('style','pushbutton',...
'string','<<',...
'position',[342 35 500 20]);
```

Figura C.1: Scheletro della GUI derivato dal codice appena presentato.



Bibliografia

- [1] C. D. 'Avanzo, P. Amodio, S. Schiff, and G. Sparacino, "A bayesian method to estimate single-trial event-related potentials with application to the study of the p300 variability," *Journal of neuroscience methods*, vol. 198, no. 1, pp. 114–124, 2011.
- [2] C. D. Avanzo, A. Goljahani, G. Pillonetto, G. D. Nicolao, and G. Sparacino, "A multi-task learning approach for the extraction of single-trial evoked potentials," *Comput. Methods Programs Biomed*, 2012.
- [3] A. Delorme and S. Makeig, "Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis," *Journal of neuroscience methods*, vol. 134, no. 1, pp. 9–21, 2004.
- [4] M. Fabiani, G. Gratton, and M. Coles, "Event-related brain potentials: Methods, theory," *Handbook of psychophysiology*, pp. 53–84, 2000.
- [5] A. Mouraux and G. Iannetti, "Across-trial averaging of event-related eeg responses and beyond," *Magnetic resonance imaging*, vol. 26, no. 7, pp. 1041–1054, 2008.
- [6] S. J. Luck, "Ten simple rules for designing erp experiments," *Event-related potentials: A methods handbook*, 2005.
- [7] C. C. Duncan, R. J. Barry, J. F. Connolly, C. Fischer, P. T. Michie, R. Naatanen, J. Polich, I. Reinvang, and C. V. Petten, "Event-related potentials in clinical research: guidelines for eliciting, recording, and quantifying mismatch negativity, p300, and n400," *Clinical Neurophysiology*, vol. 120, no. 11, pp. 1883–1908, 2009.

- [8] E. Donchin, W. Ritter, and W. C. M. et al., "Cognitive psychophysiology: The endogenous components of the erp," *Event-related brain potentials in man*, pp. 349–411, 1978.
- [9] S. J. Luck, "Event-related potentials," *DL Long, APA handbook of research methods in psychology. Washington, DC: American Psychological Association*, 2012.
- [10] T. Handy, "Event-related potentials: A methods handbook," *The MIT Press*, 2005.
- [11] J. Polich, "Updating p300: an integrative theory of p3a and p3b," *Clinical neurophysiology: official journal of the International Federation of Clinical Neurophysiology*, vol. 118, no. 10, 2007.
- [12] J. Polich, J. E. Alexander, L. O. Bauer, S. Kuperman, S. Morzorati, S. J. O' Connor, B. Porjesz, J. Rohrbaugh, and H. Begleiter, "P300 topography of amplitude/latency correlations," *Brain Topography*, vol. 9, no. 4, pp. 275–282, 1997.
- [13] A. M. Fjell, H. Rosquist, and K. B. Walhovd, "Instability in the latency of p3a/p3b brain potentials and cognitive function in aging," *Neurobiology of aging*, vol. 30, no. 12, pp. 2065–2079, 2009.
- [14] E. Donchin, M. Kubovy, M. Kutas, R. Johnson, and R. Tterning, "Graded changes in evoked response (p300) amplitude as a function of cognitive activity," *Perception and Psychophysics*, vol. 14, no. 2, pp. 319–324, 1973.
- [15] M. Kutas and S. A. Hillyard, "Reading senseless sentences: Brain potentials reflect semantic incongruity," *Science*, vol. 207, no. 4427, pp. 203–205, 1980.
- [16] M. Fatourechi, A. Bashashati, R. K. Ward, and G. E. B. et al., "Emg and eog artifacts in brain computer interface systems: A survey," *Clinical neurophysiology*, vol. 118, no. 3, pp. 480–494, 2007.
- [17] A. J. Shackman, B. W. McMenamin, H. A. Slagter, J. S. Maxwell, L. L. Greischar, and R. J. Davidson, "Electromyogenic artifacts and electroen-

- cephalographic inferences,” *Brain topography*, vol. 22, no. 1, pp. 7–12, 2009.
- [18] P. Comon, “Independent component analysis, a new concept?” *Signal processing*, vol. 36, no. 3, pp. 287–314, 1994.
- [19] C. Jutten and J. Herault, “Blind separation of sources, part i: An adaptive algorithm based on neuromimetic architecture,” *Signal processing*, vol. 24, no. 1, pp. 1–10, 1991.
- [20] G. Sparacino, S. Milani, E. Arslan, and C. Cobelli, “A bayesian approach to estimate evoked potentials,” *Computer methods and programs in biomedicine*, vol. 68, no. 3, pp. 233–248, 2002.
- [21] B. Blankertz, S. Lemm, M. Treder, S. Haufe, and K. Muller, “Single-trial analysis and classification of erp components, a tutorial,” *Neuroimage*, vol. 56, no. 2, pp. 814–825, 2011.
- [22] E. Niedermeyer and F. L. D. Silva, “Electroencephalography: basic principles, clinical applications, and related fields,” *Lippincott Williams and Wilkins*, 2005.
- [23] J. Cohen and J. Polich, “On the number of trials needed for p300,” *International Journal of Psychophysiology*, vol. 25, no. 3, pp. 249–255, 1997.
- [24] G. D. Nicolao, G. Sparacino, and C. Cobelli, “Nonparametric input estimation in physiological systems: problems, methods, and case studies,” *Automatica*, vol. 33, no. 5, pp. 851–870, 1997.
- [25] M. Neve, G. D. Nicolao, and L. Marchesi, “Nonparametric identification of population models via gaussian processes,” *Automatica*, vol. 43, pp. 1134–1144, 2007.
- [26] G. Pillonetto, G. D. Nicolao, M. Chierici, and C. Cobelli, “Fast algorithms for nonparametric population modeling of large data sets,” *Automatica*, vol. 45, pp. 173–179, 2009.

-
- [27] C. D. Avanzo, "Bayesian estimation techniques for the extraction of event-related potentials in neuroscience," Ph.D. dissertation, University of Padova, 2011.
- [28] C. Yang, Z. He, and W. Yu, "Comparison of public peak detection algorithms for maldi mass spectrometry data analysis," *BMC bioinformatics*, vol. 10, no. 1, p. 4, 2009.
- [29] P. J. et al., "An evaluation of methods for single-trial estimation of p3 latency," *Psychophysiology*, vol. 37, no. 2, pp. 153–162, 2000.
- [30] A. Paggiaro, "Sviluppo di una gui per l'analisi automatizzata della variabilit  del potenziale evento correlato p300 con test su dati reali di soggetti sani e affetti da sla," Master's thesis, University of Padova, 2012-2013.
- [31] A. Piersol and J. Bendat, "Engineering applications of correlation and spectral analysis," 1993.
- [32] S. J. Luck, "Sources of dual-task interference: Evidence from human electrophysiology," *Psychological Science*, vol. 9, no. 3, pp. 223–227, 1998.
- [33] C. D. Woody, "Characterization of an adaptive filter for the analysis of variable latency neuroelectric signals," *Medical and biological engineering*, vol. 5, no. 6, pp. 539–554, 1967.
- [34] K. Vijayalakshmi and A. M. Abhishek, "Spike detection in epileptic patients eeg data using template matching technique," *International Journal of Computer Applications*, vol. 2, no. 6, 2010.
- [35] F. T. Smulders, J. Kenemans, and A. Kok, "A comparison of different methods for estimating single-trial p300 latencies," *Electroencephalography and Clinical Neurophysiology/Evoked Potentials Section*, vol. 92, no. 2, pp. 107–114, 1994.
- [36] A. Cabasson and O. Meste, "Time delay estimation: A new insight into the woody's method," *Signal Processing Letters, IEEE*, vol. 15, pp. 573–576, 2008.

- [37] H. Akima, "A method of smooth curve fitting, Ó dtic document," *Tech. Rep.*, 1969.
- [38] J. C. Hansen and S. A. Hillyard, "Endogeneous brain potentials associated with selective auditory attention," *Electroencephalography and clinical neurophysiology*, vol. 49, no. 3, pp. 277–290, 1980.
- [39] S. J. Luck and S. A. Hillyard, "Electrophysiological evidence for parallel and serial processing during visual search," *Perception and psychophysics*, vol. 48, no. 6, pp. 603–617, 1990.
- [40] A. Kiesel, J. Miller, P. Jolicoeur, and B. Brisson, "Measurement of erp latency differences: A comparison of single-participant and jackknife-based scoring methods," *Psychophysiology*, vol. 45, no. 2, pp. 250–274, 2008.
- [41] P. D. Tuan, J. Mocks, W. Kohler, and T. Gasser, "Variable latencies of noisy signals: estimation and testing in brain potential data," *Biometrika*, vol. 74, no. 3, pp. 525–533, 1987.
- [42] P. Jaskowski, "Amplitudes and latencies of single-trial erps estimated by a maximum-likelihood method," *Biomedical Engineering, IEEE Transactions on*, vol. 46, no. 8, pp. 987–993, 1999.
- [43] <http://www.mathworks.it/discovery/matlab-gui.html>.
- [44] <http://sccn.ucsd.edu/eeglab/>.
- [45] A. Delorme, H. Serby, and S. Makeig, "Eeglab tutorial," *wartz Center for Computational Neuroscience, Institute for Neural Computation, University of California San Diego. Retrieved on January*, vol. 15, 2006.
- [46] S. Cerutti, G. Baselli, D. Liberati, and G. Pavesi, "Single sweep analysis of visual evoked potentials trough a model of parametric identification," *Biological Cybernetics*, vol. 56, pp. 111–120, 1987.
- [47] S. Cerutti, G. Chiarenza, D. Liberati, P. Mascellani, and G. Pavesi, "A parametric method of identification of single trial event-related potentials in

- the brain,” *IEEE Transaction on Biomedical Engineering*, vol. 35, no. 9, pp. 701–711, 1988.
- [48] D. Liberati, S. Cerutti, E. D. Ponzio, V. Ventimiglia, and L. Zaninelli, “The implementation of an autoregressive model with exogenous input in a single sweep visual evoked potential analysis,” *J. Biomedical Engineering*, vol. 11, pp. 285–292, 1989.
- [49] L. Rossi, A. M. Bianchi, A. Merzagora, A. Gaggiani, S. Cerutti, and F. Bracchi, “Single trial somatosensory evoked potential extraction with arx filtering for a combined spinal cord intraoperative neuromonitoring technique,” *BioMedical Engineering OnLine*, vol. 6, no. 2, 2007.
- [50] A. Montin, “Sviluppo di un programma software per l’analisi single sweep di potenziali evocati cognitivi,” Master’s thesis, University of Padova, 2003–2004.

Ringraziamenti

Un non troppo antico canto Tuareg afferma: «...la mia testa, una pentola a pressione, acchiappatela e il mio cervello vi esploderà fra le mani...»

Stravolgendone il significato, ho subito pensato che questa è la comune condizione dello studente tesista.

Tutti si aspettano, quindi, che l'introduzione di questo breve paragrafo sia del tipo «Dopo mesi e mesi di estenuante fatica, insonnia e peripezie ho finalmente portato a termine questa impervia scalata...» invece posso tranquillamente dichiarare di aver avuto l'opportunità di lavorare ad un progetto talmente stimolante che i momenti di stesura della tesi, nonostante le innumerevoli difficoltà, son risultati una fresca brezza di primavera se paragonati al fin troppo noto blizzard del periodo esami.

E' quindi doveroso un ringraziamento al prof. Giovanni Sparacino per avermi fatto affrontare una sfida ardua ma decisamente appagante e per avermi sempre consigliato prontamente e con chiarezza. Un ringraziamento anche ad Andrea che ha condiviso con me lo sviluppo dell'interfaccia grafica. Sempre disponibile e molto preparato, è stato un piacere lavorare con lui e lo sarà, mi auguro, anche in futuro.

Ringrazio i miei genitori. Da una parte mi hanno incasinato la vita rendendomi un bipolo di raziocinio e filosofia/superstizione, dall'altra mi hanno insegnato la generosità d'animo e la generosità più pratica e meno trascendente ma altrettanto importante. Mi hanno reso una persona migliore.

Ringrazio Pietro e Elsa che mi hanno cresciuto come un figlio e hanno sempre creduto in me giustificando troppo spesso i miei comportamenti ribelli e da marmocchio viziato.

Ringrazio mia sorella che si è dimostrata un'ottima allieva del corso «Parole a caso e altre bestialità in dialetto noalese».

Ringrazio Trilli, cagnaccia. Vuole bene a modo suo.

Ringrazio i me amighi de Noae, che se i me becca a scivar in itaian i me desfa de stramusoni. In particolare Ciccio e Steva. Amici fragili.

Ringrazio i miei amici di Mestre e le eroiche serate da Berto. In particolare
Stefano. Fedele compagno.

Ringrazio anche tutti i restanti parenti, amici e amiche che festeggeranno con
me questo traguardo.

Avevo lasciato uno spazio ulteriore per ringraziare una possibile futura morosa
ma a quanto pare, in queste ultime settimane, non sono stato abbastanza
fortunato da trovarla.

Ringrazio Fabrizio De Andrè, che mi ha permesso di andare verso il mare con
una barca per traversare.

Una barca per naufragare.