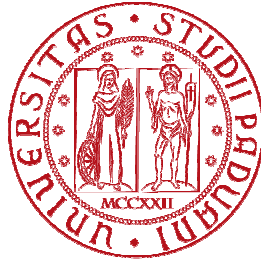


Università degli Studi di Padova

Facoltà di Ingegneria

Dipartimento di Ingegneria dell'informazione



Corso di Laurea in Bioingegneria

UN METODO DI MULTI-TASK LEARNING PER LA STIMA SINGLE-TRIAL DI POTENZIALI EVOCATI COGNITIVI

Relatore:

Prof. Giovanni Sparacino

Correlatori:

Ing. Costanza D'Avanzo e Ing. Anahita Goljahani

Laureando:

Andrea Schiavon

Anno Accademico **2009/2010**

Indice

Introduzione	5
1. Un approccio di multi-task learning: identificazione non parametrica di modelli di popolazione attraverso processi Gaussiani	7
1.1 Aspetti generali del metodo	8
1.1.1 <u>Impostazione del problema e assunzioni</u>	8
1.1.2 <u>Stima bayesiana</u>	11
1.1.2.1 <u>Stima della curva media</u>	11
1.1.2.2 <u>Stima della curva individuale</u>	12
1.2 Ottimizzazione per i casi di campionamento standardizzato	12
2. Stima dei potenziali evocati	13
2.1 I Potenziali evocati	13
2.2 Breve review sui metodi di stima dei potenziali evocati	15
2.2.1 <u>Stima di un potenziale evocato “medio”</u>	15
2.2.2 <u>Stima single-trial</u>	16
2.3 Applicazione dell’algoritmo di multi-task learning per la stima di PE	18
3. Applicazione a problemi simulati	21
3.1 Costruzione del data set simulato	21
3.2 Risultati delle simulazioni	23
3.2.1 <u>Caso con rumore a varianza realistica</u>	23
3.2.2 <u>Caso con rumore a varianza elevata</u>	30

3.3 Conclusioni.....	35
4. Applicazione a dati reali registrati durante un compito cognitivo	37
4.1 Descrizione del data set	37
4.2 Risultati dell'analisi per i diversi gruppi di soggetti	39
4.2.1 <u>Soggetti giovani e sani</u>	39
4.2.2 <u>Soggetti anziani e sani</u>	44
4.2.3 <u>Soggetti cirrotici senza encefalopatia epatica (HE)</u>	49
4.2.4 <u>Soggetti cirrotici con encefalopatia epatica (HE)</u>	54
4.3 Conclusioni	58
4.4 Problemi aperti	69
5. Conclusioni e sviluppi futuri	63
Appendice A	65
Appendice B	71
Bibliografia	79

Introduzione

In molti ambiti clinici e di ricerca sono di fondamentale interesse i potenziali evocati (PE), ovvero quelle variazioni dell'attività elettrica cerebrale che si sovrappongono al tracciato elettroencefalografico (EEG) come conseguenza di uno stimolo esterno.

La stima dei potenziali evocati è resa difficile a causa del basso valore assunto dal rapporto segnale-rumore (SNR); infatti, il segnale d'interesse è oscurato dall'EEG di base che ha, rispetto al segnale utile, un'ampiezza molto più elevata e contenuto in frequenza comune. Per risolvere questo problema, N segnali (sweeps), registrati sottoponendo il soggetto ad una serie di N stimoli identici, vengono di solito mediati in modo da ottenere una stima del potenziale medio. Questa semplice tecnica di media chiamata "Averaging Convenzionale" (CA) si basa sulle assunzioni, nella realtà mai verificate, che il PE non vari durante le registrazioni e che l'EEG di fondo sia stazionario. Inoltre, essa necessita spesso di un numero N di registrazioni molto elevato.

In letteratura sono state proposte moltissime tecniche, alcune finalizzate al miglioramento della stima fornita dal CA, altre focalizzate sulla stima della risposta evocata in ogni singola registrazione. Finora però nessuna di queste tecniche è diventata uno standard e gli studi di psicologi e clinici riguardanti i PE si basano, di fatto, su considerazioni derivanti dall'analisi del potenziale medio.

In questa tesi verrà presentata una tecnica di tipo multi-task learning per la stima media e single-trial di potenziali evocati cognitivi. Il metodo stima gli N potenziali evocati processando contemporaneamente tutte le N sweeps a disposizione. Esso sfrutta alcune informazioni a priori sulla regolarità dei segnali incogniti, risultando adatto sia a migliorare la stima del PE medio, permettendo l'impiego di un numero inferiore di sweeps rispetto alla tecnica convenzionale, sia a fornire le risposte a livello del single-trial. Questo tipo di approccio è utilizzato dal settore della farmacocinetica e farmacodinamica (Verrotta et al.,1997) [8] dove è conosciuto con il nome di "analisi di popolazione". Difatti, le singole N registrazioni (e.g. sweeps) sono viste come soggetti di una popolazione omogenea caratterizzati da una componente media, comune alla popolazione, e da una componente individuale.

In particolare, questa tesi riprende un lavoro già intrapreso da Suman [1] con l'intento di ampliare l'applicazione ai dati reali e di risolvere alcuni problemi aperti. Il primo punto critico da migliorare era l'elevata complessità computazionale in quanto, per la stima delle singole risposte di un soggetto, l'algoritmo richiedeva troppo tempo. Vi erano poi dei problemi legati alla definizione delle condizioni iniziali e finali delle curve e, nei casi di rumore sovrapposto troppo elevato, le stime single-trial "collassavano" nella media, ottenendo così delle curve molto simili che non descrivevano la variabilità tra le risposte.

Per quanto riguarda l'applicazione, sono stati presi in considerazione segnali EEG registrati durante un compito di Simon. Nello specifico, ogni soggetto veniva sottoposto a due differenti

stimoli visivi, ognuno dei quali richiedeva una differente risposta da parte del paziente. In questo modo, la tecnica descritta nella tesi è stata valutata per la stima single-trial di potenziali evocati cognitivi.

L'originalità di questo e del precedente lavoro sta nell'adattare ed utilizzare l'approccio di multi-task learning quando i limiti del processo di misura sono nella rumorosità dei campioni, come nel caso dei potenziali evocati.

La tesi è organizzata come segue.

Nel capitolo 1 verrà esposta la teoria che sta alla base dell'algoritmo sviluppato nella Tesi. In particolare, si farà riferimento alla variante del metodo, numericamente "ottimizzata", presentata da (Pillonetto et al.) nel 2007 [4].

Nel capitolo 2 si parlerà in generale dei potenziali evocati e della loro stima facendo un breve richiamo sui metodi più noti in letteratura. La tecnica di multi-task learning impiegata verrà quindi contestualizzata nell'ambito dei PE.

Successivamente, nel capitolo 3, si esporranno i risultati dell'applicazione dell'algoritmo ad una serie di dati simulati che ricreano alcune situazioni reali. Sarà possibile vedere come l'algoritmo fornisca delle stime buone sia del potenziale medio che delle curve single-trial, riuscendo così a descrivere molto bene la variabilità che c'è tra le varie registrazioni.

Nel capitolo 4, visto il buon esito delle simulazioni, si passerà all'applicazione ai dati reali. Nello specifico, il metodo verrà utilizzato per estrarre i PE da segnali EEG acquisiti durante un compito di Simon. Verranno presentati i risultati per diverse tipologie di soggetti sottoposti al compito (giovani sani, anziani sani, cirrotici senza encefalopatia epatica e cirrotici con encefalopatia epatica).

1. Un approccio di multi-task learning: identificazione non parametrica di modelli di popolazione attraverso processi Gaussiani

In questo capitolo sarà presentato , in termini generali, il metodo multi-task learning che sarà poi impiegato nella stima dei potenziali evocati.

Il cosiddetto “*multi-task learning*” è un approccio alla stima che prevede l’analisi congiunta di diversi data set per stimare simultaneamente diverse funzioni multidimensionali (tasks) [6].

L’idea alla base di questo metodo è che, se esiste una correlazione tra le varie funzioni, le informazioni ricavate dal data set relativo ad un task possono essere sfruttate per le stime di tutti gli altri tasks .

Negli ultimi anni, questo metodo ha acquisito sempre maggiore interesse nella ricerca ed è stato impiegato, ad esempio, nel riconoscimento di immagini, nella decisione in ambito clinico, nella predizione di risultati scolastici, nell’identificazione della dinamica di motori. In particolare nel campo di applicazione d’interesse per questa tesi, il multi-task learning è stato finora utilizzato per studi di farmacocinetica e farmacodinamica nella cosiddetta “analisi di popolazione” [11]. In questo caso, i dati a disposizione sono acquisiti tramite esperimenti su soggetti diversi che presentano alcune caratteristiche in comune.

Un esempio è dato dalla figura 1.1 dove sono rappresentate le concentrazioni di un farmaco dopo sollecitazione intravenosa.

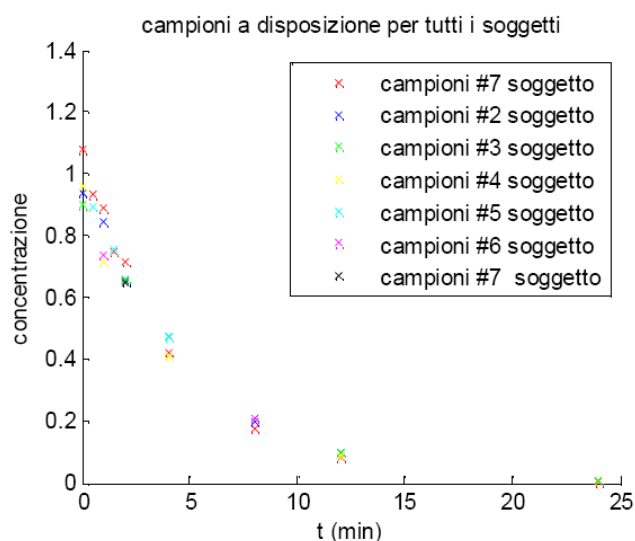


Figura 1.1. Campioni di sette soggetti appartenenti ad una popolazione.

Considerando i dati di un solo soggetto, sarebbe molto difficile ricostruire il profilo esatto di concentrazione del farmaco, l'impiego di un'analisi congiunta di tutti i soggetti a disposizione, che tiene conto delle caratteristiche comuni alla popolazione, aiuta invece ad ottenere una buona stima sia della curva media che delle singole curve individuali (Figura 1.2).

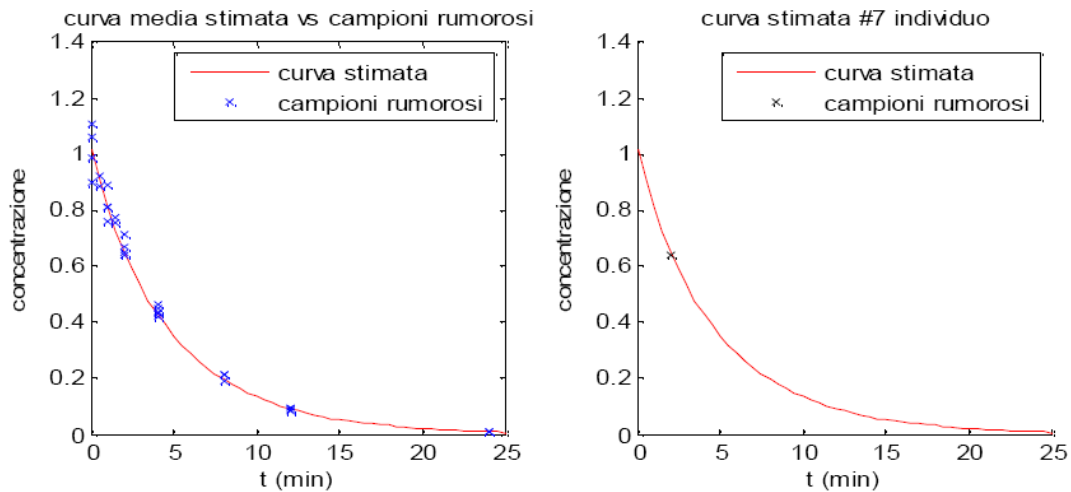


Figura 1.2. Curva media di popolazione (sx) e curva individuale del soggetto #7 (dx)

Ogni funzione da stimare può essere quindi modellata come la combinazione di una componente media, comune a tutto l'insieme di funzioni, e uno shift individuale che rappresenta lo scostamento della funzione da stimare dalla componente media.

Il metodo multi-task learning utilizzato in questa tesi verrà descritto nella prima parte del presente capitolo. Si tratta di un metodo non parametrico, presentato da De Nicolao et al. 2005 che assume solo che i segnali da stimare abbiano delle caratteristiche di regolarità. Nella seconda parte verrà invece trattato l'aspetto computazionale. In particolare, una versione efficiente del metodo, recentemente presentata da Pillonetto et al. 2007, sarà trattata in dettaglio.

1.1 Aspetti generali del metodo

1.1.1 Impostazione del problema e assunzioni

Si vogliono stimare N funzioni a partire da N corrispondenti insiemi di campioni rumorosi mediante un approccio multi-task learning in un contesto bayesiano. Si suppone quindi che le N funzioni incognite siano N soggetti appartenenti ad una popolazione omogenea e che

l'informazione derivante dai campioni rumorosi associati ad una funzione possa essere utilmente impiegata nella stima di tutte le altre funzioni.

Per i campioni associati alla j -esima funzione, con $j=1,\dots,N$, si supponga valere il seguente modello additivo per le misure:

$$y^j(t_k) = z^j(t_k) + v^j(t_k) \quad k=1,\dots,n_j \quad (1.1)$$

dove k è l'indice degli istanti di campionamento e n_j è il numero di campioni disponibili per il soggetto j -esimo, $y^j(t_k)$ è il valore rumoroso associato alla funzione j -esima e misurato all'istante t_k , $z^j(t_k)$ è il valore della j -esima funzione incognita all'istante t_k mentre $v^j(t_k)$ è l'errore di misura associato a quest'ultima.

Si noti che, in generale, non è richiesto che i campionamenti delle N funzioni siano uniformi tra di loro in quanto ogni soggetto può avere la sua griglia di campionamento specifica. Nel caso in cui tutti i soggetti siano invece campionati agli stessi istanti si parla di *campionamento standardizzato*.

Le funzioni individuali incognite vengono modellate come la somma di due componenti:

$$z^j(t_k) = \bar{z}(t_k) + \tilde{z}^j(t_k) \quad k=1,\dots,n_j \quad (1.2)$$

con $\bar{z}(t_k)$ che rappresenta la curva "media" di popolazione all'istante t_k mentre $\tilde{z}^j(t_k)$ lo scostamento rispetto alla componente media all'istante t_k .

Introduciamo ora una notazione matriciale definendo:

$$\mathbf{y} = [y_1^1 \dots y_{n_1}^1 \ y_1^2 \dots y_{n_2}^2 \ y_1^N \dots y_{n_N}^N]^T \quad (1.3)$$

$$\bar{\mathbf{z}} = [\bar{z}(t_1^1) \dots \bar{z}(t_{n_1}^1) \ \dots \ \bar{z}(t_1^N) \dots \bar{z}(t_{n_N}^N)]^T \quad (1.4)$$

$$\tilde{\mathbf{z}} = [\tilde{z}^1(t_1^1) \dots \tilde{z}^1(t_{n_1}^1) \ \dots \ \tilde{z}^N(t_1^N) \dots \tilde{z}^N(t_{n_N}^N)]^T \quad (1.5)$$

$$\mathbf{v} = [v_1^1 \dots v_{n_1}^1 \ v_1^2 \dots v_{n_2}^2 \ v_1^N \dots v_{n_N}^N]^T \quad (1.6)$$

dove \mathbf{y} , $\bar{\mathbf{z}}$, $\tilde{\mathbf{z}}$, \mathbf{v} sono vettori di dimensione $n_{\text{tot}} = n_1 + n_2 + \dots + n_N$ costituiti da N blocchi (dal primo all'ultimo soggetto).

Si può quindi riscrivere il modello delle misure in forma matriciale come:

$$\mathbf{y} = \bar{\mathbf{z}} + \tilde{\mathbf{z}} + \mathbf{v} \quad (1.7)$$

con $\mathbf{v} \sim N(0, \Sigma_v)$, dove Σ_v è la matrice di covarianza dell'errore di misura (dimensione $n_{\text{tot}} \times n_{\text{tot}}$).

Si suppone che per ogni soggetto gli errori di misura siano tra loro mutuamente indipendenti, gaussiani, a media nulla e varianza pari a $(\sigma_k^j)^2$.

Si assume che \bar{z} e \tilde{z}^j siano entrambi processi stocastici gaussiani indipendenti tra di loro e indipendenti rispetto agli errori di misura e che siano dei segnali regolari modellabili tramite processi di Wiener integrati.

Il valore atteso di \tilde{z}^j può essere assunto nullo in quanto rappresenta lo shift rispetto alla curva media, la stessa ipotesi può essere fatta sul valore atteso di \bar{z} se i dati sono stati opportunamente scalati sottraendone la media. Per le funzioni di auto-covarianza di \bar{z} e \tilde{z}^j si sfrutta il modello di Wiener, ottenendo, come dimostrato in [3]

$$\bar{\mathbf{R}}(t_k, \tau) = \bar{\lambda}^2 \begin{cases} \frac{t_k^2}{2} \left(\tau - \frac{t_k}{3} \right) & t_k \leq \tau \\ \frac{\tau^2}{2} \left(t_k - \frac{\tau}{3} \right) & t_k > \tau \end{cases} \quad (1.8)$$

e

$$\tilde{\mathbf{R}}(t_k, \tau) = \tilde{\lambda}^2 \begin{cases} \frac{t_k^2}{2} \left(\tau - \frac{t_k}{3} \right) & t_k \leq \tau \\ \frac{\tau^2}{2} \left(t_k - \frac{\tau}{3} \right) & t_k > \tau \end{cases} \quad (1.9)$$

Si noti che, modellando gli shift individuali tramite processi di Wiener integrati, la varianza tende ad infinito con t (osservare le equazioni (1.8) e (1.9)). 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.

Gli iper-parametri $\bar{\lambda}^2$ e $\tilde{\lambda}^2$ determinano, rispettivamente, quanto i segnali incogniti \bar{z} e \tilde{z}^j 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:

$$\theta = [\bar{\lambda}^2 \tilde{\lambda}^2]^T \quad (1.10)$$

che massimizzano la funzione di verosimiglianza.

Ciò equivale a determinare

$$\operatorname{argmin}_{\theta} J(\mathbf{y}; \theta) \quad (1.11)$$

dove $J(\mathbf{y}; \theta) = \mathbf{y}^T \operatorname{Var}[\mathbf{y}]^{-1} \mathbf{y} + \log[\det(\operatorname{Var}[\mathbf{y}])]$.

Si noti che il calcolo della funzione costo $J(\mathbf{y};\theta)$ richiede l'inversione e il computo del determinante di una matrice di dimensioni n_{TOT} , le quali hanno una complessità computazionale di $O(n_{TOT}^3)$. Dal momento che il minimo di $J(\mathbf{y};\theta)$ viene calcolato iterando il calcolo della funzione per diversi valori di θ , il tempo di computazione diventa oneroso nel caso in cui n_{TOT} sia molto grande.

Dalle ipotesi sul vettore degli errori di misura e sui segnali incogniti segue che il valore atteso di \mathbf{y} è nullo e che la sua varianza è pari a:

$$\text{var}[\mathbf{y}] = \text{var}[\bar{\mathbf{z}}] + \text{var}[\hat{\mathbf{z}}] + \Sigma_v \quad (1.12)$$

1.1.2 Stima bayesiana

Le informazioni a priori di tipo probabilistico disponibili sui segnali incogniti vengono sfruttate in un contesto bayesiano.

Sia x un generico vettore aleatorio da stimare e sia \mathbf{y} il vettore delle osservazioni. Nel caso di processi congiuntamente gaussiani, quale è il caso preso in esame, si dimostra che lo stimatore a minimo errore quadratico medio è dato da:

$$\hat{x} = E[x | \mathbf{y}] = E[x] + \text{cov}[x, \mathbf{y}] \text{var}[\mathbf{y}]^{-1} (\mathbf{y} - E[\mathbf{y}]) \quad (1.13)$$

Sostituendo nella precedente espressione al vettore x le funzioni \bar{z} e \hat{z}^j si ottengono le stime della curva media e delle curve individuali.

1.1.2.1 Stima della curva media

Nel caso della curva media \bar{z} , tenendo conto di tutte le assunzioni precedentemente esposte, la (1.13) diventa:

$$\hat{z}(t) = \text{cov}[\bar{z}(t), \bar{\mathbf{z}}] \text{var}[\mathbf{y}]^{-1} \mathbf{y} = [\bar{\mathbf{R}}(t, t_1^1) \dots \bar{\mathbf{R}}(t, t_{n_1}^1)] \text{var}[\mathbf{y}]^{-1} \mathbf{y} \quad (1.14)$$

1.1.2.2 Stima della curva individuale

Nel caso invece delle curve individuali z^j con $j=1,\dots,N$, tenendo ancora conto di tutte le assunzioni precedentemente esposte, la (1.13) diventa:

$$\hat{z}^j(t) = \hat{z}(t) + \text{cov}[\tilde{z}^j(t), \tilde{z}] \text{var}[\mathbf{y}]^{-1} \mathbf{y} = \hat{z}(t) + [\tilde{\mathbf{R}}^j(t, t_1^1) \dots \tilde{\mathbf{R}}^j(t, t_{n1}^1)] \text{var}[\mathbf{y}]^{-1} \mathbf{y} \quad (1.15)$$

1.2 Ottimizzazione per i casi di campionamento standardizzato

Consideriamo ora il caso in cui tutti i segnali siano campionati agli stessi istanti di tempo come avviene nella registrazione dei potenziali evocati. Nell'appendice A vengono riportati gli algoritmi sviluppati da Pillonetto et al. 2007 che permettono di calcolare le stime della curva media e delle curve individuali in modo efficiente.

In particolare, dall'espressione delle stime si vede che bisogna invertire la $\text{var}[\mathbf{y}]$, una matrice di dimensione $n_{\text{TOT}} \times n_{\text{TOT}}$. Inoltre, nel calcolo degli iper-parametri sono ripetute più volte inversioni e calcoli di determinanti per minimizzare la funzione costo. Questa è la parte del codice che presenta il più alto onere computazionale ed è su questi punti che si sfrutta il campionamento standardizzato per diminuire i tempi di attesa.

Dove è coinvolto il calcolo dell'inversa si possono svolgere le operazioni con complessità pari a $O(n^3)$ invece di $O(n_{\text{tot}}^3)$, con n che corrisponde al numero di campioni per soggetto. Per quanto riguarda il calcolo del determinante, utilizzando la fattorizzazione di cholesky si riesce a ridurre la complessità da $O(n_{\text{tot}}^3)$ a $O(mn^3)$ dove m è il numero di soggetti.

Data la mole di dati a disposizione, il codice che è stato sviluppato per questa tesi sfrutta questi algoritmi che permettono così di avere le stime della curva media e delle curve individuali in tempi più ragionevoli (si passa da decine di minuti a poco più di un minuto).

2. Stima dei potenziali evocati

Come è stato già detto nell'introduzione, negli ultimi anni sono state presentate molteplici tecniche per la stima dei potenziali evocati. Dopo aver introdotto i potenziali evocati, in questo capitolo verrà presentata una breve review dei diversi metodi che si possono trovare in letteratura. Alla fine verrà descritta l'applicazione del nuovo approccio di multi-task learning alla stima dei potenziali evocati. Questo rappresenta una novità perché, da quanto sappiamo, il multi-task learning è stato finora utilizzato, in ambito biomedico, solo in studi di farmacocinetica e farmacodinamica dove la principale sfida stava nei pochi campioni a disposizione e non nella elevata rumorosità dei campioni come nei PE.

2.1 Potenziali evocati

I potenziali evocati (PE) sono potenziali causati dall'attività elettrica generata dal nostro cervello a causa di uno stimolo sensoriale. Nello specifico, i PE consistono in variazioni monofasiche dell'attività elettrica cerebrale che si presentano sul tracciato elettroencefalografico (EEG) con un ritardo più o meno fisso rispetto all'evento che le evoca.

I potenziali evocati possono essere classificati in due distinte categorie: i potenziali evocati *esogeni*, determinati esclusivamente dalle caratteristiche fisiche dello stimolo, e i potenziali evocati *endogeni*, determinati invece dal significato che lo stimolo assume a livello psicologico. Un esempio di potenziali evocati endogeni sono i potenziali evocati cognitivi. Si definiscono inoltre potenziali *event-related* (ERP) quei potenziali che vengono generati a livello cerebrale senza un preciso stimolo fisico esterno.

L'andamento nel tempo di questi segnali presenta successioni di picchi (peak) e di valli (nadir) come si può notare dalla figura 2.1.

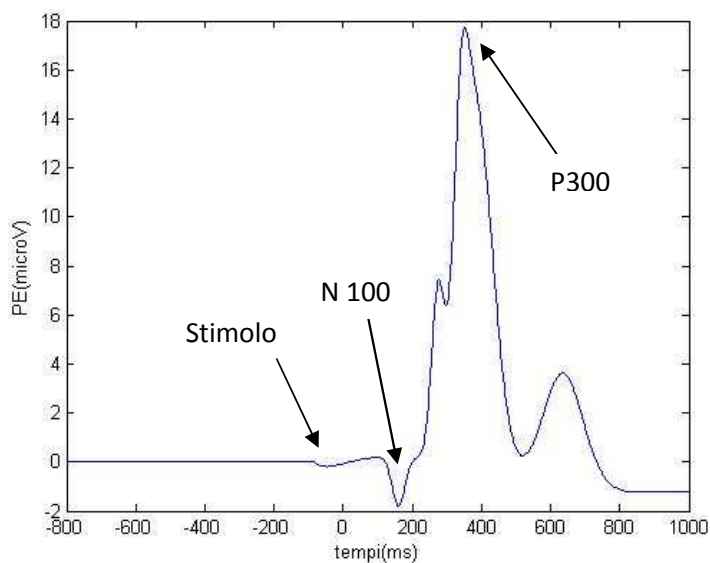


Figura 2.1: esempio di potenziale evocato(simulazione), a circa 300 ms si può vedere il picco della P300.

Dal punto di vista clinico, si è interessati soprattutto alle ampiezze e alle latenze di queste componenti. In particolare, le componenti principali sono la N100 che corrisponde al primo picco negativo che si trova circa 100 ms dopo lo stimolo e la P300 che corrisponde al picco maggiore che si registra circa 300 ms dopo lo stimolo. Il loro studio risulta spesso di fondamentale importanza, sia nella diagnosi di diverse patologie che interessano il sistema nervoso, sia nello studio e nella comprensione dei processi neurofisiologici.

Generalmente questi segnali hanno un'ampiezza dell'ordine di pochi microvolt e sono immersi nell'elettroencefalogramma di base (EEG) che ha, invece, un'ampiezza di decine di microvolt. Questo fatto rende la stima dei potenziali evocati un problema complesso perché il rapporto segnale-rumore (SNR) risulta molto basso e non sono sufficienti le semplici tecniche di filtraggio per estrarre il segnale d'interesse dall'EEG di fondo.

2.2 Breve review sui metodi di stima dei potenziali evocati

2.2.1 Stima di un potenziale evocato "medio"

Le prime tecniche adottate per estrarre il PE sono le cosiddette tecniche di *averaging* che consistono nel mediare un numero N di sweeps (registrazioni). In questo modo, si migliora il rapporto segnale rumore, ottenendo così una risposta media che però non tiene conto della variabilità di risposta che ci può essere tra uno stimolo e il successivo.

Ad esempio, sia $y_i(t)$ il segnale i-esimo misurato dopo lo stimolo e si assuma per esso il modello additivo:

$$y_i(t) = u_i(t) + v_i(t) \quad (2.1)$$

dove $u_i(t)$ è il potenziale evocato e $v_i(t)$ rappresenta un rumore a media nulla e varianza σ^2 , non correlato allo stimolo.

Applicando l'*averaging convenzionale* (AC) si trova che:

$$\bar{y}(t) = \frac{1}{N} \sum_{i=1}^N y_i(t) = u(t) + \frac{1}{N} \sum_{i=1}^N v_i(t) \quad (2.2)$$

il cui valore atteso è il nostro potenziale d'interesse:

$$E[\bar{y}(t)] = u(t) \quad (2.3)$$

I principali limiti di questa tecnica sono che l'EEG viene considerato stazionario, mentre nella realtà questo è vero per non più di 2 secondi, ed è necessario un numero elevato di registrazioni per ottenere una buona stima.

Successivamente, per ridurre ulteriormente il rumore, è stato introdotto l'*averaging pesato* (Lutkenhoner et al., 1985) che attribuisce un diverso peso alle varie sweeps in base al rumore associato. Un'importante limitazione di questo metodo sta nel modello di rumore che si assume. In accordo con questo modello, il rumore varia tra le epoche ma è stazionario in ogni sweep, mentre molti tipi di rumore (non solo artefatti) possono variare all'interno di ogni registrazione. Di conseguenza, i pesi possono essere sottostimati in parti di epoca dove il rumore è elevato e sovrastimati in parti dove il rumore è ridotto.

Nel 2002 (Sparacino et al.) hanno pubblicato un approccio di tipo *bayesiano* che sfrutta informazioni statistiche di secondo ordine sull'EEG e sui potenziali evocati che vengono rispettivamente calcolate dal pre-stimolo e dai dati di post-stimolo grazie ad un criterio di

regolarizzazione. Il rapporto segnale-rumore (SNR) risulta migliorato, permettendo così l'identificazione di latenza e ampiezza dei picchi principali, e servono molte meno sweeps del CA per ottenere una buona stima del potenziale medio, calcolato tramite una media pesata delle sweeps filtrate.

In seguito si è poi cercato di mediare utilizzando altri tipi di stimatori come ad esempio la *mediana*. Per combinare i vantaggi della media e della mediana, (Leonowicz et al.,2005) hanno deciso di utilizzare i *trimmed-estimators* (efficienti e robusti stimatori locali) migliorando così la stima media quando il numero di epoche risultava ridotto.

Recentemente (Rahne et al.,2008) hanno proposto il *sorted averaging*: questa tecnica, basata sull'assunzione di intercambiabilità delle sweeps di un data set, si applica ordinando le epoche in base al valore del Root Mean Square (RMS) stimato. Iniziando con sweeps poco rumorose, l'SNR aumenta fino ad un massimo per poi diminuire a causa dell'arrivo di registrazioni sempre più rumorose. Il valore massimo viene poi utilizzato per decidere il numero ottimo di sweeps.

Nei diversi metodi precedentemente riportati si considerano uguali tutti i potenziali evocati. Questa assunzione risulta però limitativa perché da vari studi si è visto che, tra una registrazione e l'altra, le latenze e le ampiezze dei picchi principali possono variare a causa di processi di adattamento e/o apprendimento. Di conseguenza, stimando solo il potenziale medio, si perde molta informazione ed è necessario un numero elevato di sweeps per ottenere un buon segnale. Molte volte, per motivi pratici legati all'esperimento, non è possibile ottenere il numero desiderato di registrazioni.

Da tutte queste considerazioni emerge l'importanza di studiare approcci diversi che permettano l'analisi single-sweep, in modo da classificare le singole risposte ed analizzarne l'eventuale variabilità.

2.2.2 Stima single-trial

Negli ultimi anni sono state proposte molteplici tecniche per la stima single-trial dei potenziali evocati. Riportiamo in questa sezione alcuni degli approcci più rilevanti per avere un'idea generale dello stato dell'arte nell'ambito della stima dei PE.

Le tecniche *bayesiane* si basano sulla possibilità di stimare il vettore dei parametri incognito non solo utilizzando un insieme di dati misurati (*posterior*), ma anche sfruttando tutte le possibili

informazioni a priori (*prior*) sul segnale stesso eventualmente disponibili. Un problema fondamentale è però stabilire se si debba dare più peso al prior piuttosto che al posterior, o viceversa. A questo scopo si ricorre ai cosiddetti *metodi di regolarizzazione*, ossia ad algoritmi che permettono di stabilire un compromesso “ottimo” tra prior e posterior in base a specifiche uguaglianze che dovranno essere verificate. (P.A.Karjalainen, 1997) ha proposto un’applicazione del metodo della *regolarizzazione in sottospazi* per la stima dei PE single-trial. L’applicazione di questa tecnica a dati simulati e dati reali ha portato a risultati molto promettenti.

Negli stessi anni (Nati-Ali et al.,1996) hanno introdotto una tecnica per la stima di *Brainstem Auditory Evoked Potentials* (BAEP): il sistema fisiologico è stato modellato attraverso un modello lineare e la stima è stata considerata come un problema inverso. I potenziali evocati vengono quindi stimati massimizzando un’appropriata distribuzione a posteriori (MAP).

(Friston et al.,2006) hanno poi utilizzato l’approccio empirico di Bayes nelle risposte evocate ed indotte nell’ambito della magnetoencefalografia/elettroencefalografia (MEG/EEG). Il metodo era basato sulla stima della componente di covarianza attraverso la *restricted maximum likelihood* (ReML).

Nel 2007 (Georgiadis et al.) hanno valutato le informazioni a priori sulle proprietà dei PE tramite il sottospazio dei segnali stimati e la decomposizione degli 17uto valori. In questo caso, nelle situazioni di fluttuazione dinamica tra uno stimolo e il successivo, l’informazione a priori può essere utilizzata attraverso metodi di stima ai minimi quadrati bayesiana (*filtro di Kalman*). Si è dimostrato che pochi 17uto valori dominanti della matrice di correlazione dei dati sono in grado di modellare i cambiamenti di alcune componenti dei potenziali evocati.

Citiamo ora qualche altra tecnica fuori dall’ambito bayesiano.

L’*RBF Neural Network* di (Merzagora et al.,2004) utilizza reti neurali a tre strati con funzione di attivazione a simmetria radiale per stimare i singoli potenziali. Questo approccio permette di ottenere segnali regolari con un tempo computazionale moderato.

L’*iterative Independent Component Analysis* di (Zouridakis et al.,2007) è un metodo che processa un data set multivariato e che permette di estrarre anche solo una componente d’interesse. Questa tecnica sfrutta il fatto che le sorgenti, le diverse zone dello scalpo, sono statisticamente indipendenti mentre i segnali misurati non lo sono. Si cercherà quindi di risalire alle sorgenti del PE, eliminando il contributo delle sorgenti del rumore EEG.

Ci sono poi diverse tecniche tempo-frequenza che utilizzano la *Continuous Wavelet Transform* (Mouraux et al.,2008) che permette di identificare la locazione temporale e l’ampiezza delle componenti principali. L’idea su cui si basa la WT è paragonabile a quella su cui si basa l’analisi in frequenza di Fourier, ossia scomporre il segnale da analizzare in termini di funzioni elementari, ma mentre nell’analisi di Fourier tali funzioni sono rappresentate da sinusoidi, nella WT esse

sono delle funzioni, dette appunto “wavelets”, oscillatorie e limitate nel tempo, in modo da permettere la rappresentazione di segnali costituiti da componenti oscillatorie variabili in ampiezza e/o frequenza. Nello specifico, le funzioni “wavelets” si ottengono da una funzione base per traslazione o cambiamento di scala nel tempo.

Esistono ancora molti altri metodi tra cui il *filtraggio spazio temporale* per la stima single-trial della P300 (Li et al.,2009), il *Selective Eigen-Rate Principal Components* (SER) di (Andrew et al.,2005) che sfrutta gli 18uto valori delle componenti d’interesse e diversi *metodi parametrici*.

In letteratura sono presenti molteplici tecniche oltre a quelle già nominate e sarebbe impossibile elencarle tutte.

2.3 Applicazione dell’algoritmo di multi-task learning per la stima di potenziali evocati

Adattiamo ora l’approccio di multi-task learning alla stima dei potenziali evocati.

L’algoritmo riportato nel capitolo 2 calcola la stima di N segnali relativi ad N soggetti che appartengono ad una popolazione omogenea. Nel caso della stima di potenziali evocati, gli N segnali sono rappresentati dalle N sweeps registrate da un paziente. Questi segnali possono essere considerati appartenenti alla stessa popolazione perché ,essendo registrati dallo stesso paziente sottoposto sempre allo stesso stimolo, saranno molto simili tra di loro.

Assumiamo il seguente modello delle misure per la sweep j-esima, con $j=1,\dots,N$:

$$y^j(t_k) = z^j(t_k) + v^j(t_k) \quad k=1,\dots,n \quad (2.4)$$

dove y^j è il segnale che viene registrato, z^j il j-esimo potenziale evocato, v^j il rumore sovrapposto al segnale utile ed n il numero di campioni di ogni sweep.

Il nuovo metodo sfrutta l’informazione di tutte le registrazioni per calcolare le stime individuali che vengono modellate come la somma di tre componenti: curva media $\bar{z}(t)$, shift individuale $\tilde{z}^j(t)$ e rumore v^j con $j=1,2,\dots,N$. La curva media è il potenziale medio, che può essere confrontato con quello stimato dalle tecniche di averaging, mentre lo shift individuale rappresenta lo scostamento dalla componente media di ogni singola sweep.

Curva media e curve individuali vengono modellate tramite processi di *Wiener* integrati. Questi modelli presentano solo un iper-parametro che descrive la regolarità del segnale e che viene stimato dai dati tramite *Maximum Likelihood*.

Il rumore, l’EEG di fondo, viene identificato con un modello auto regressivo i cui coefficienti vengono calcolati dal pre-stimolo di ogni sweep. In particolare, per i coefficienti vengono usate

le equazioni di *Yule Walker* e per l'ordine ottimo del modello si fa riferimento al valore assunto dal *Final Prediction Error*. Sfruttando quindi l'ipotesi che l'EEG è un processo stazionario per al più 2 secondi, si può costruire la matrice di covarianza del rumore EEG utilizzando le varianze σ^2 e i coefficienti a_1, \dots, a_p calcolati dal pre-stimolo. Si ha quindi che:

$$\Sigma_v = \sigma^2 (\mathbf{A}^T \mathbf{A})^{-1} \quad (2.5)$$

con \mathbf{A} matrice nxn di Toeplitz triangolare inferiore con prima colonna $[1 \ a_1 \ a_2 \ \dots \ a_p \ 0 \ \dots \ 0]$.

Per i dettagli si faccia riferimento al codice presente nell'appendice B.

Nel prossimo capitolo verrà riportata la valutazione preventiva del codice che è stata effettuata su problemi simulati, costruiti in modo da ricreare alcune situazioni reali.

3. Applicazione a problemi simulati

In questo capitolo verranno esposti i risultati dell'applicazione dell'algoritmo di multi-task learning a segnali EEG simulati che ricreano delle possibili situazioni reali. In questo modo, avendo a disposizione il reale andamento dei potenziali evocati da stimare, sarà possibile valutare la performance del metodo.

Dopo una breve descrizione del data set, si passerà alla presentazione dei risultati ottenuti su due gruppi di segnali simulati caratterizzati rispettivamente da rumore di fondo "realistico" e rumore di fondo "elevato". La stima del PE medio sarà messa a confronto con quella fornita dalla media convenzionale in termini di errore percentuale di stima. Delle stime single-trial sarà valutata la capacità di descrivere la variabilità dei segnali veri e saranno calcolate medie e deviazioni standard dei parametri che descrivono la P300. In particolare, la correlazione tra latenze vere e latenze stimate e il confronto tra l'istogramma delle latenze vere e quello delle latenze stimate saranno presi in considerazione.

Nel lavoro precedente si era fatto ricorso alle stesse simulazioni e si erano riscontrati problemi nella stima delle curve individuali nel caso di varianza "elevata". Nello specifico, si ottenevano delle stime dei singoli potenziali evocati che combaciavano con il potenziale medio stimato. Inoltre, nel nostro caso è stato utilizzato un numero inferiore di trials (25 al posto di 73) in modo da verificare che le stime risultino ancora buone. Questo aspetto è molto importante perché nella realtà, per problemi legati all'esperimento, spesso si hanno a disposizione poche registrazioni per ogni soggetto.

3.1 Costruzione del data set simulato

Le simulazioni sono state costruite prendendo come esempio potenziali evocati cognitivi registrati durante un compito di Simon, descritto in dettaglio nel capitolo successivo. Si è partiti da una curva "modello", ottenuta tramite tecniche di media e filtraggio su segnali reali misurati dall'elettrodo Pz (zona parietale, linea mediana), per poi introdurre delle variazioni casuali agli istanti di occorrenza e alle ampiezze dei picchi P300 e N100. In questo modo, a partire dalla curva "modello", sono state create le varie curve individuali che dovrebbero simulare la variabilità di quelle vere; a queste curve è stato poi aggiunto del rumore in modo da ricreare il disturbo generato dall'EEG nei casi reali.

Un esempio di simulazione di questo tipo di potenziale è rappresentato nella figura 3.1 dove si possono notare le due principali componenti d'interesse: la N100 e la P300. Osservando la figura a destra si può notare che il rumore di fondo è molto elevato ed oscura completamente il segnale utile.

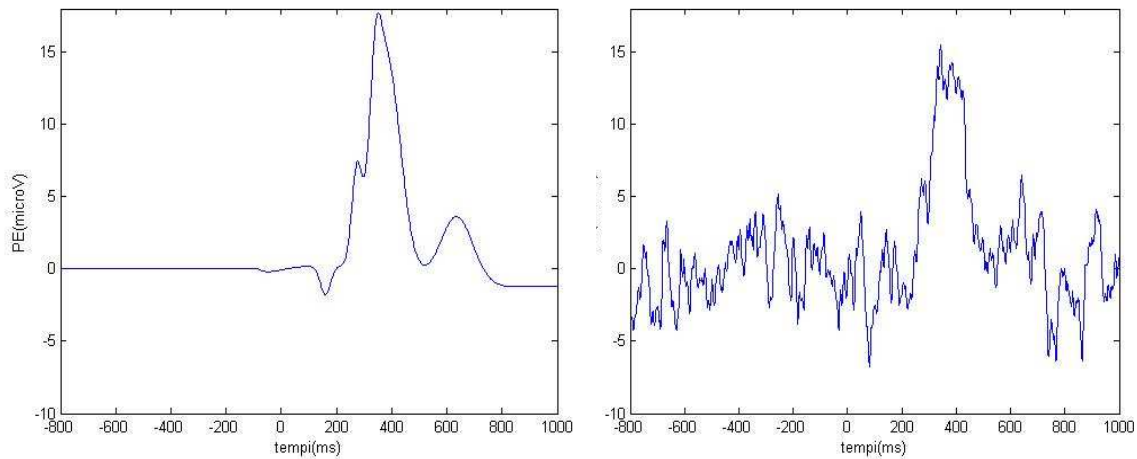


Figura 3.1: potenziale evocato simulato (sx) e potenziale evocato simulato con l'aggiunta di rumore realistico (dx).

Per rendere il rumore più simile a quello reale, si è cercato di simulare le caratteristiche del segnale EEG di fondo. Il modello del rumore è stato ottenuto dalle porzioni di pre-stimolo del segnale EEG registrato durante il compito su un soggetto giovane e sano. In particolare, è stato identificato un modello AR, il cui ordine è stato determinato dal principio di parsimonia FPE (final prediction error).

Il caso di rumore elevato è stato poi ottenuto semplicemente raddoppiando la varianza del rumore bianco che pilota il modello AR identificato dai tracciati reali.

3.2 Risultati delle simulazioni

3.2.1 Caso con rumore a varianza realistica

Di seguito verranno mostrati i risultati, relativi al caso di varianza realistica, sia per quanto riguarda la stima del potenziale medio che le stime single-trial.

Per la stima del potenziale medio si consideri il seguente errore percentuale di stima:

$$\text{Err} = 100 \frac{\|\bar{z} - \hat{z}\|^2}{\|\bar{z}\|^2} \quad (3.1)$$

dove \bar{z} è un vettore che contiene i campioni del potenziale medio vero e \hat{z} un vettore che contiene i campioni del potenziale medio stimato.

Nella figura 3.2 si confronta l'andamento dell'errore percentuale di stima associato al nostro algoritmo (linea blu) con quello dato dall'averaging convenzionale (linea verde) al variare del numero di registrazioni acquisite.

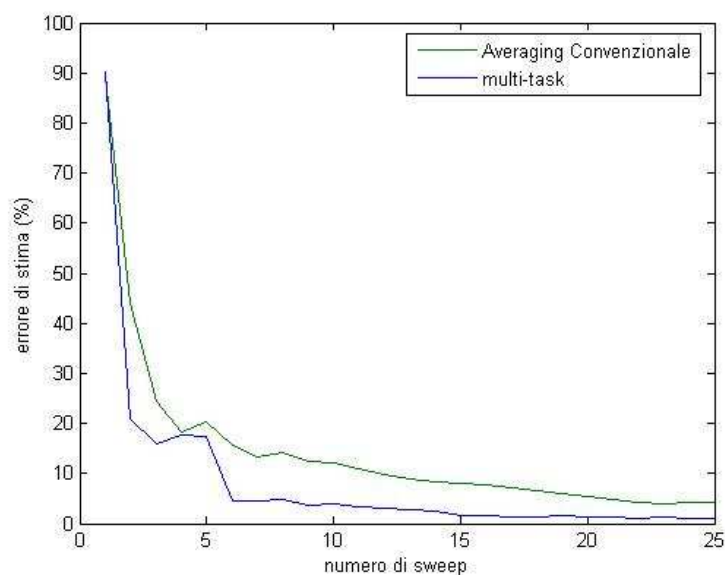


Figura 3.2: errore percentuale di stima all'aumentare del numero di sweeps utilizzate: stima fornita dal metodo di multi-task learning in blu e stima fornita dall'averaging convenzionale in verde.

Dal grafico si può vedere come l'errore del nostro metodo sia sempre minore rispetto a quello dato dall'averaging convenzionale, soprattutto quando il numero di sweeps utilizzate è piccolo. Si noti che l'errore è inferiore all'1% quando il numero di registrazioni è pari a circa 25: questo significa che per le nostre stime si potranno ottenere dei buoni risultati avendo come data set solo 25 sweeps.

Vediamo ora come viene stimato il potenziale medio. In figura 3.3 sono riportate la stima della curva media ottenuta con l'algoritmo di multi-task learning (linea blu), la stima ottenuta tramite averaging convenzionale (linea verde) e il potenziale medio vero (linea rossa).

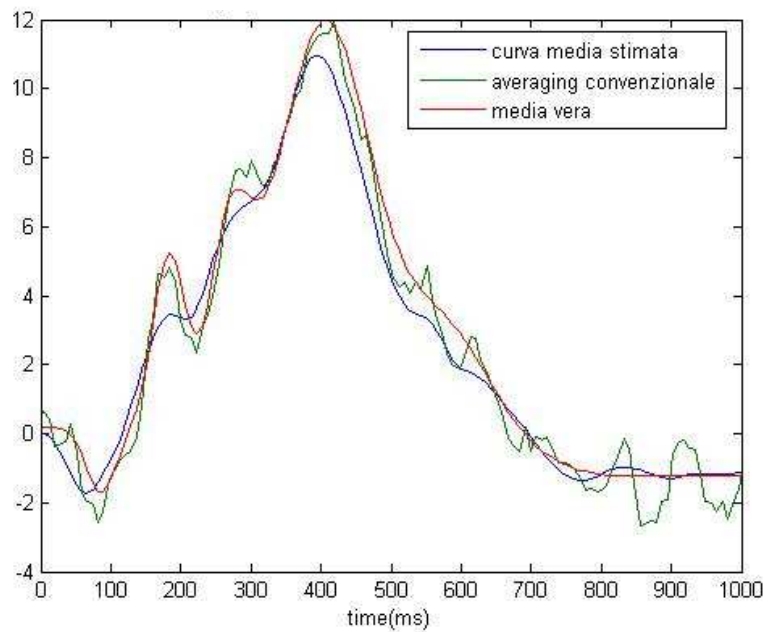


Figura 3.3:Stima del potenziale medio: curva vera in rosso, curva stimata con il metodo multi-task learning in blu e curva stimata con l'averaging convenzionale in verde.

Come si può vedere dal grafico, la curva stimata è simile al potenziale vero e più regolare rispetto a quella calcolata con l'averaging convenzionale. Inoltre, il picco P300 è identificato perfettamente e, quindi, si può dire che, nel caso di potenziali evocati simulati, l'algoritmo fornisca una stima molto buona del potenziale evocato medio.

Prendiamo ora in considerazione le stime delle curve individuali. Riportiamo come esempio, nelle figure 3.4 e 3.5, la stima di due sweeps casuali tra le 25 che sono state usate come data set.

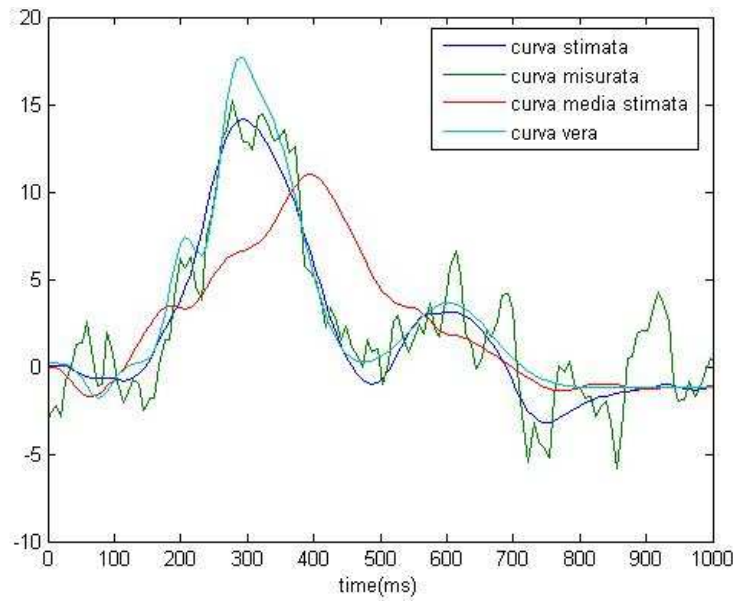


Figura 3.4: Stima single-trial per la sweep # 15: potenziale vero (ciano), curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in rosso, sweep grezza in verde.

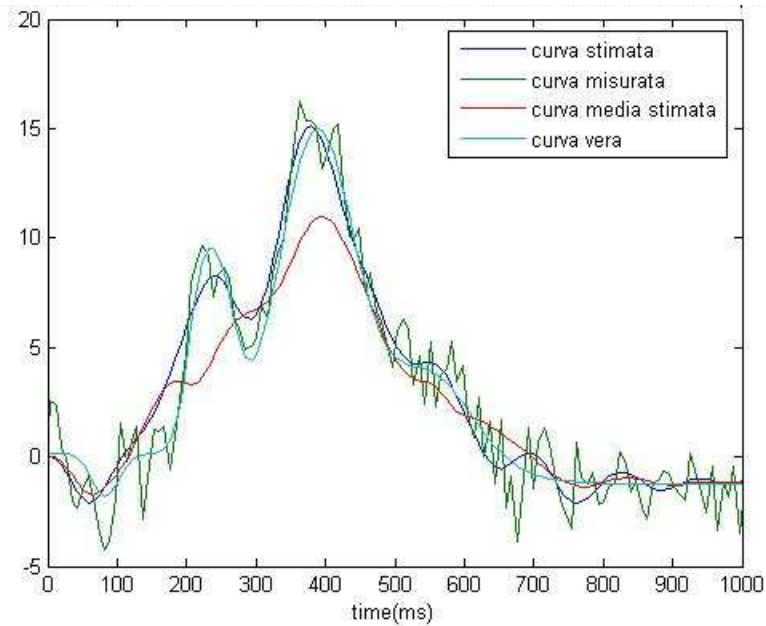


Figura 3.5: Stima single-trial per la sweep # 20: potenziale vero (ciano), curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in rosso, sweep grezza in verde.

Analizzando le due figure, si vede che le curve stimate seguono abbastanza bene l'andamento del potenziale vero. Per quanto riguarda la P300, le latenze sono stimate molto bene mentre, in alcuni casi, le ampiezze tendono ad essere sottostimate (come nella figura 3.4).

Inoltre, si noti come l'algoritmo riesca a stimare bene la curva individuale (in blu) discostandosi dalla media (in rosso) e mantenendone più o meno il profilo. Questo è dovuto al fatto che il metodo considera le curve individuali come la somma di due componenti: la curva media e lo scostamento dal potenziale medio.

Valutiamo ora come l'algoritmo stimi la variabilità tra le curve individuali e confrontiamola con la variabilità delle curve vere (fig. 3.6 e 3.7).

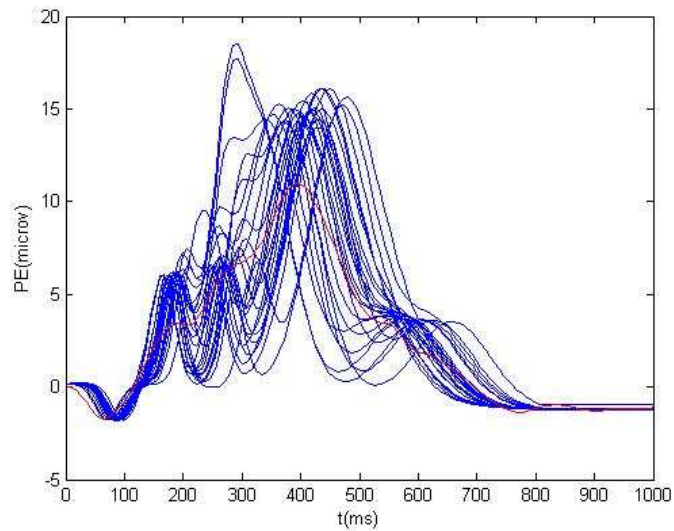


Figura 3.6: Variabilità dei PE simulati (blu) nel caso di varianza realistica, in rosso è rappresentato il PE medio vero.

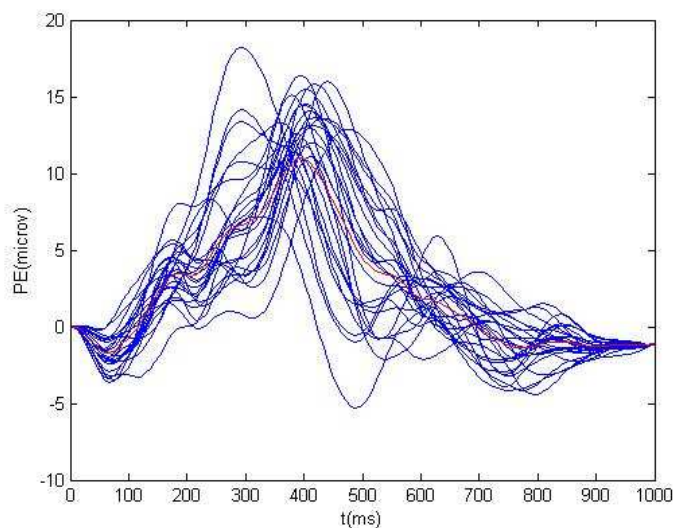


Figura 3.7: Variabilità dei PE stimati (blu) nel caso di varianza realistica, in rosso è rappresentato il PE medio stimato.

Confrontando i due grafici precedenti, si può vedere come la variabilità delle curve individuali sia stata stimata molto bene.

Facciamo ora un'analisi più dettagliata per quanto riguarda i parametri della P300. I valori medi di latenza e ampiezza della P300 dei profili veri e di quelli stimati dal metodo di multi-task learning sono riportati nella tabella 3.1.

	Media latenze [ms]	SD latenze	Media Ampiezza [μV]	SD Ampiezza
Profili veri	400.4688	47.2683	15.2592	1.0440
Multi-task learning	397.7273	49.4106	13.4321	2.2347

Tabella 3.1: Valori medi veri e stimati di latenza e ampiezza della P300 con relative SD nel caso di varianza realistica.

Dalla tabella si può vedere che i valori medi trovati con il nuovo metodo sono simili a quelli veri, soprattutto per quanto riguarda le latenze. Un'analisi di correlazione tra valori stimati e valori veri conferma questo risultato.

In figura 3.8 è riportata la correlazione tra le ampiezze vere e stimate della P300 mentre in quella successiva la correlazione tra le latenze vere e stimate della P300.

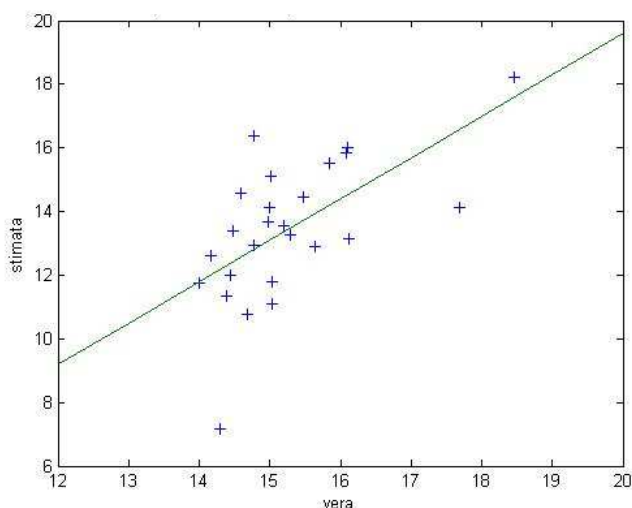


Figura 3.8: Correlazione tra le ampiezze vere e stimate della P300 nel caso di varianza realistica.

L'algoritmo tende a disperdere un po' le ampiezze stimate rispetto a quelle vere. In questo caso, il coefficiente di correlazione risulta infatti pari a 0.6.

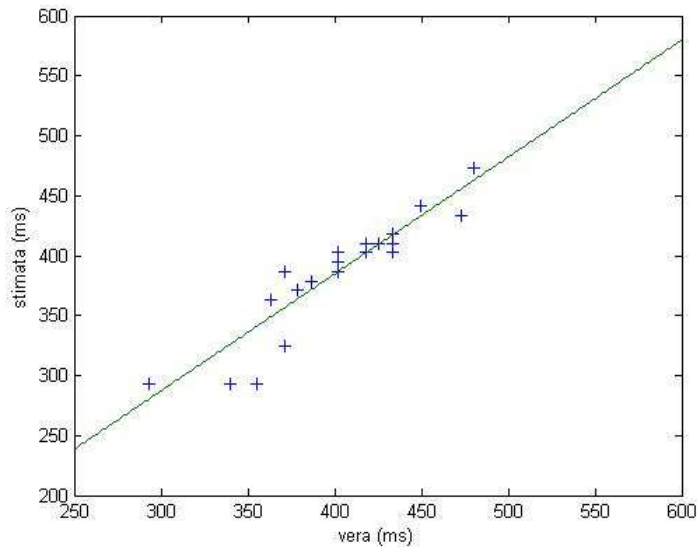


Figura 3.9: Correlazione tra le latenze vere e stimate della P300 nel caso di varianza realistica.

I risultati dell'analisi di correlazione tra le latenze della P300 sono, invece, molto buoni e poco dispersi. In pratica, i valori sono tutti vicinissimi alla retta di regressione (in verde) e il coefficiente di correlazione risulta essere molto elevato, pari a circa 0.93.

Nelle figure 3.10 e 3.11 sono riportati gli istogrammi che mostrano le distribuzioni delle latenze vere e stimate della P300.

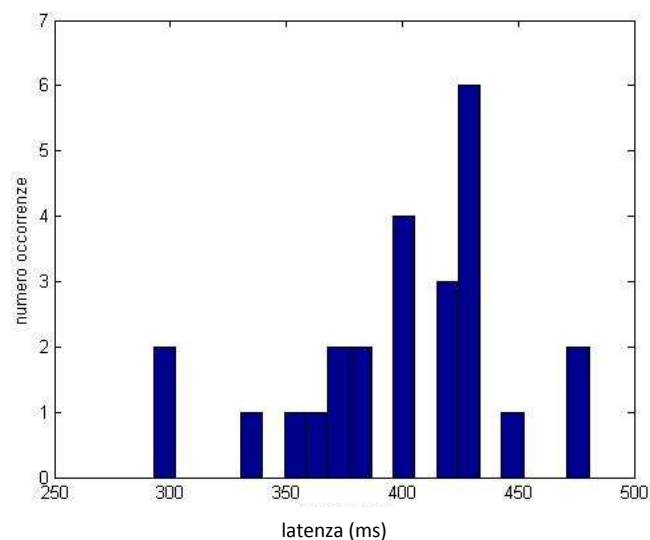


Figura 3.10: Distribuzione delle latenze vere della P300 nel caso di varianza realistica.

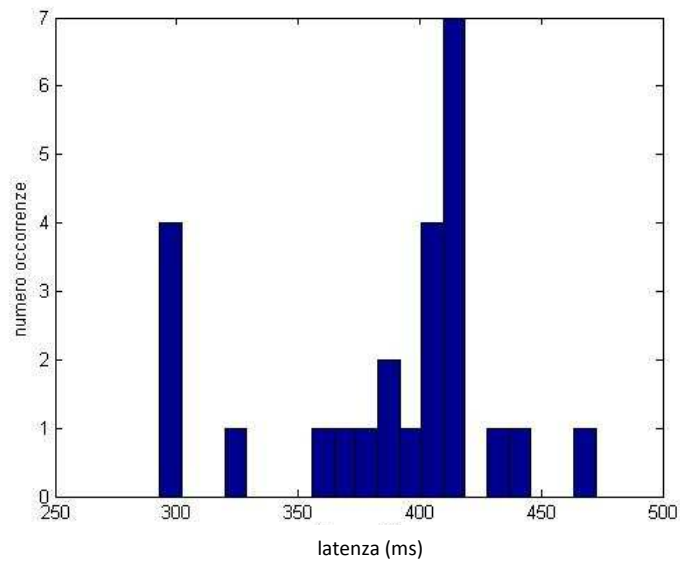


Figura 3.11: Distribuzione delle latenze stimate della P300 nel caso di varianza realistica.

Si può affermare che le distribuzioni delle latenze stimate della P300 siano molto simili a quelle reali. L'algoritmo riesce, quindi, a stimare molto bene la variabilità che c'è tra le latenze dei singoli potenziali evocati.

In conclusione, possiamo affermare che nel caso di rumore correlato a varianza realistica le stime individuali e della curva media risultano affidabili, soprattutto per quanto riguarda le latenze della P300.

Se il rumore dovesse aumentare, ci aspettiamo ovviamente che la situazione peggiori. Per verificarlo, nel prossimo paragrafo applicheremo l'algoritmo alle stesse simulazioni di prima con l'unica differenza che la varianza del rumore è stata raddoppiata.

3.2.2 Caso con rumore a varianza elevata

Riguardo alla stima del potenziale medio, in figura 3.12 è riportato l'andamento della curva media stimata dall'algoritmo di multi-task learning (linea blu), quella calcolata tramite l'averaging convenzionale (linea verde) e il potenziale medio vero (linea rossa).

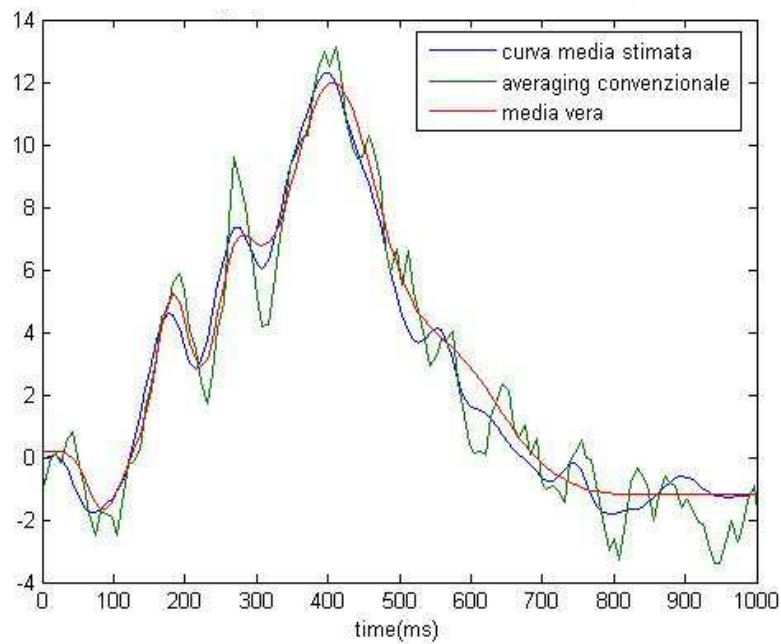


Figura 3.12: Stima del potenziale medio: curva vera in rosso, curva stimata con il metodo multi-task learning in blu e curva stimata con l'averaging convenzionale in verde.

Anche in questo caso il potenziale medio stimato è molto simile a quello vero e decisamente più regolare di quello calcolato con l'averaging convenzionale. In particolare, sia l'ampiezza che la latenza del picco P300 sono ben identificati.

Ora passiamo ad analizzare di nuovo le stime delle curve individuali, cioè come l'algoritmo stima le singole sweeps acquisite da un paziente.

Nelle prossime figure vengono riportate le stesse registrazioni del paragrafo precedente in modo da poter confrontare i risultati dei due casi di varianza realistica e varianza elevata.

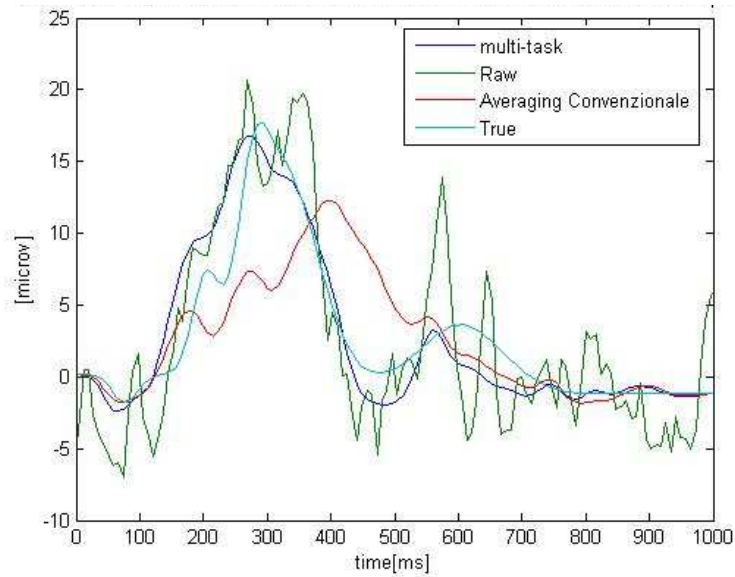


Figura 3.13: Stima single-trial per la sweep # 15: potenziale vero (ciano), curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in rosso, sweep grezza in verde.

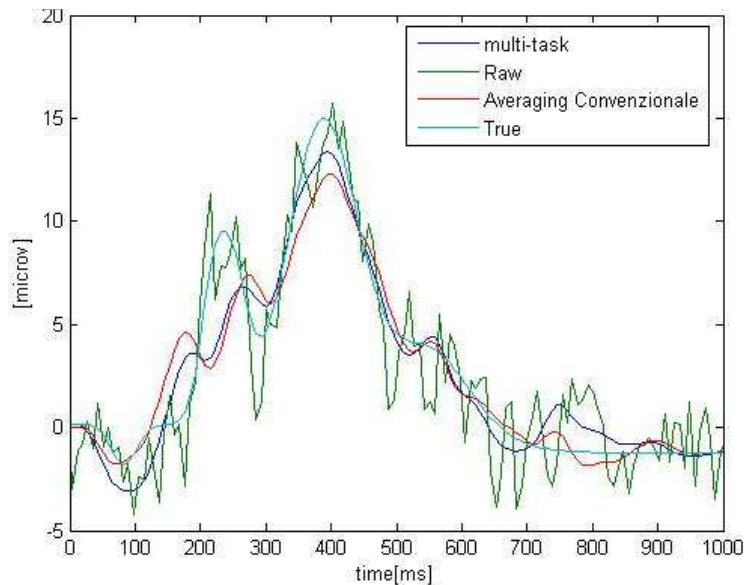


Figura 3.14: Stima single-trial per la sweep # 20: potenziale vero (ciano), curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in rosso, sweep grezza in verde.

Analizzando i due grafici, si può affermare che le stime date dal nuovo metodo siano ancora buone pur avendo la varianza del rumore in ingresso al modello AR doppia rispetto al caso precedente.

Anche in questo caso si può notare che le stime si discostano dalla curva media (si faccia riferimento soprattutto alla fig. 3.13), descrivendo così la variabilità inter-sweeps.

Confrontiamo ora la variabilità tra le curve individuali (fig. 3.15 e 3.16).

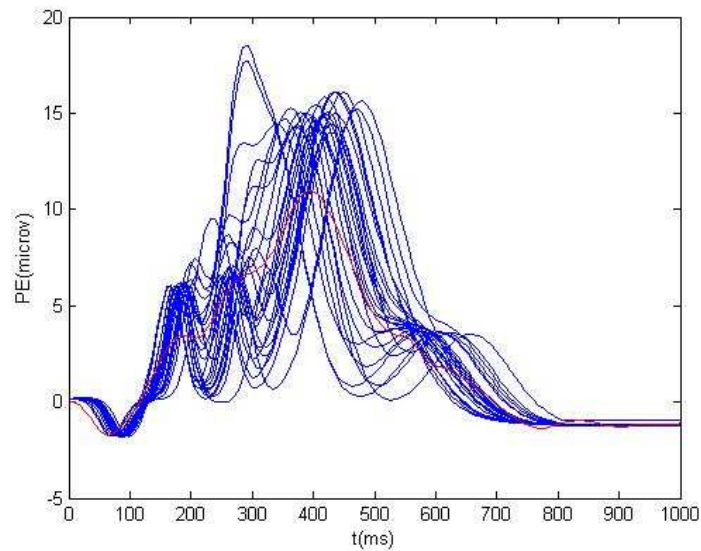


Figura 3.15: Variabilità dei PE simulati (blu) nel caso di varianza elevata, in rosso è rappresentato il PE medio vero.

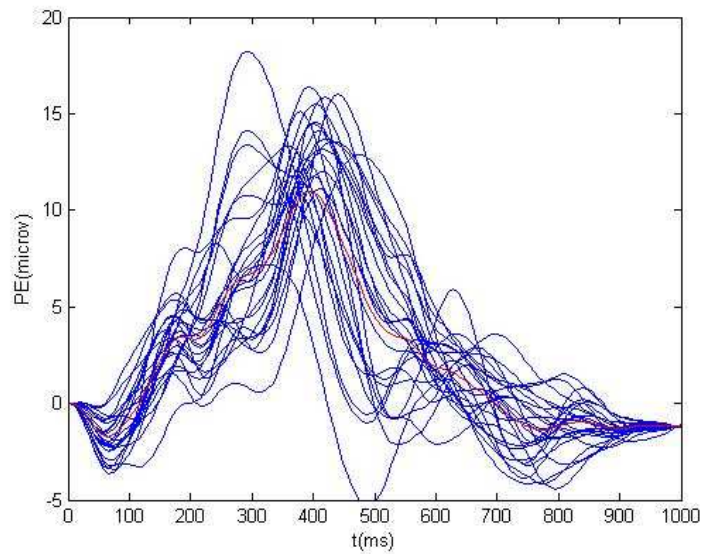


Figura 3.16: Variabilità dei PE stimati (blu) nel caso di varianza elevata, in rosso è rappresentato il PE medio stimato.

I due grafici risultano ancora abbastanza simili tra loro. Se poi confrontiamo i profili stimati con quelli del caso di varianza realistica, vediamo ovviamente dei peggioramenti perché è stato aumentato il rumore sovrapposto al segnale utile.

I valori medi di latenza e ampiezza della P300 dei profili veri e di quelli stimati dal nuovo algoritmo di multi-task learning sono riportati in tabella 3.2.

	Media latenze [ms]	SD latenze	Media Ampiezza [μ V]	SD Ampiezza
Profili veri	400.4688	47.2683	15.2592	1.0440
Multi-task learning	390.1563	46.9877	13.4662	2.2848

Tabella 3.2: Valori medi veri e stimati di latenza e ampiezza della P300 con relative SD nel caso di varianza elevata.

Confrontando questi valori con quelli del paragrafo precedente, si può dire che la latenza media sia stata stimata meglio nel caso di varianza realistica, mentre le ampiezze risultano sottostimate come prima.

I risultati sono comunque molto buoni se teniamo in considerazione il fatto che si è studiato un caso sfavorevole.

In fig. 3.17 e 3.18 mostriamo i grafici di correlazione tra ampiezze vere e stimate e tra latenze vere e stimate della P300 con i rispettivi coefficienti di correlazione.

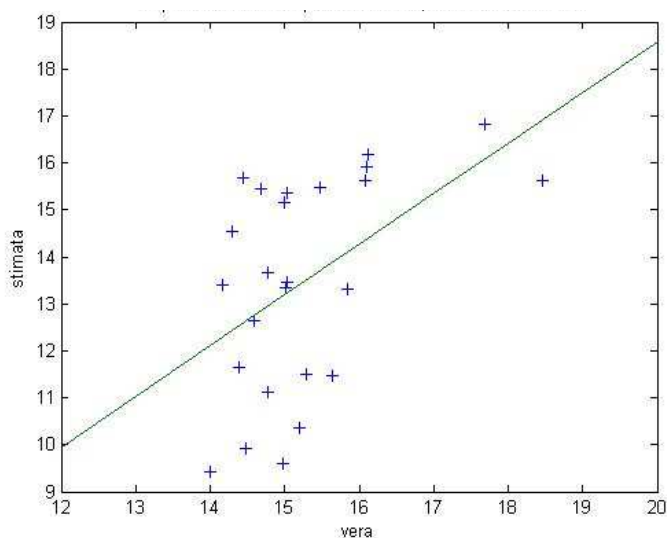


Figura 3.17: Correlazione tra le ampiezze vere e stimate della P300 nel caso di varianza elevata.

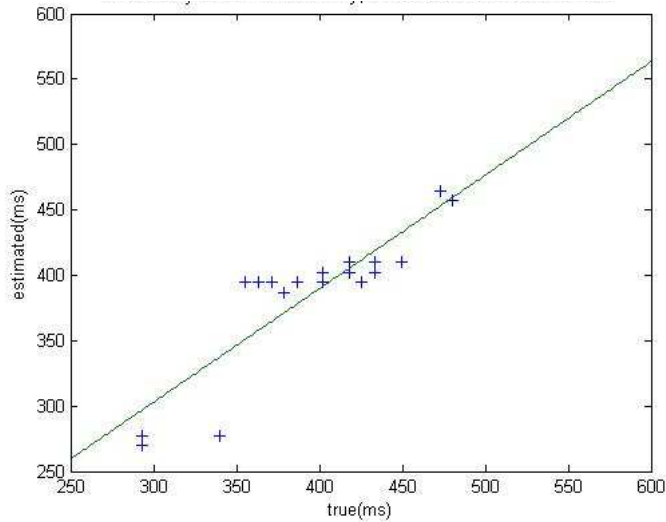


Figura 3.18: Correlazione tra le latenze vere e stimate della P300, caso di varianza elevata.

Dalla fig. 3.17 si può notare che l’algoritmo tende a disperdere le ampiezze stimate rispetto a quelle vere e ciò è confermato anche da un valore basso del coefficiente di correlazione pari a 0.49.

Per quanto riguarda le latenze, in fig. 3.18 si vede che queste vengono stimate meglio delle ampiezze e che presentano ancora un buon coefficiente di correlazione (0.87) che è, come previsto, più basso del valore ottenuto nel caso di varianza realistica.

Nelle figure 3.19 e 3.20 sono riportati gli istogrammi che mostrano le distribuzioni delle latenze vere e stimate. Si può vedere come le latenze stimate abbiano una distribuzione che presenta meno variabilità rispetto a quelle vere.

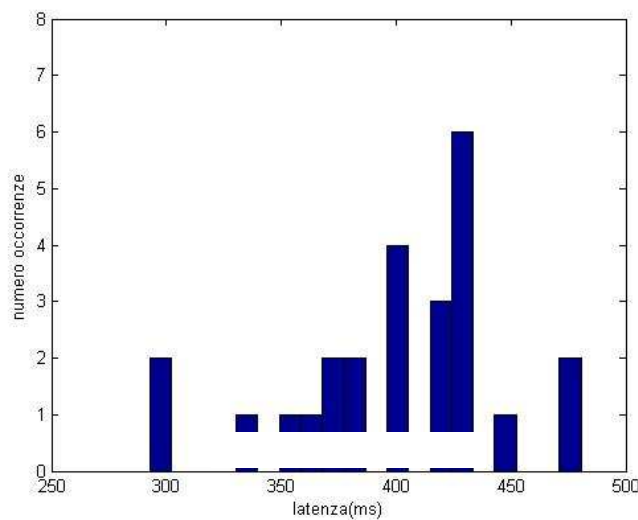


Figura 3.19: Distribuzione latenze vere della P300 nel caso di varianza elevata.

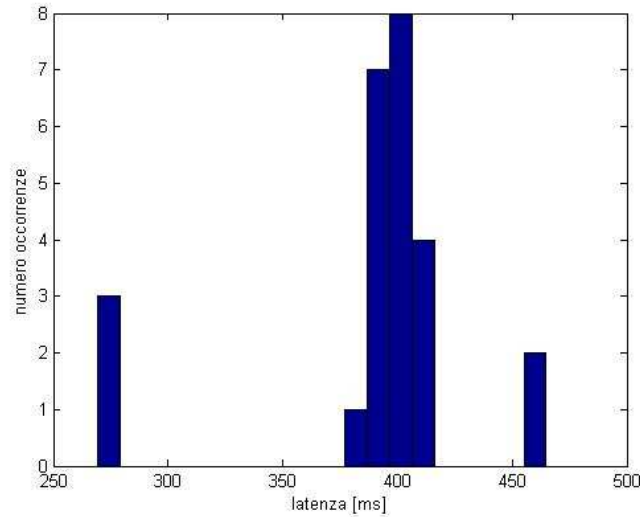


Figura 3.20: Distribuzione latenze stimate della P30 nel caso di varianza elevata.

3.3 Conclusioni

In questo capitolo è stata valutata la performance del metodo nella stima del potenziale medio e dei potenziali single-trial su dati simulati. Simulazioni di questo tipo sono già state presentate nel capitolo 5 di [1]. In questa tesi, in particolare, sono state prese in considerazione due situazioni: una con rumore di fondo “realistico”, l’altra con rumore di fondo “elevato”.

Nel precedente lavoro si erano riscontrati dei problemi nella stima single-trial nel caso di rumore “elevato”. In quella circostanza si ottenevano delle stime individuali che non si differenziavano significativamente dalla media. Dai risultati riportati precedentemente, si può vedere che questo problema non si ripresenta più. Inoltre, è stato diminuito l’onere computazionale utilizzando meno trials e sottocampionando il segnale nella parte di codice in cui vengono stimati gli iperparametri.

Nello specifico, dall’analisi del potenziale medio si è visto come l’algoritmo di multi-task learning fornisca sempre delle buone stime della curva media con molte meno sweeps di quelle necessarie all’averaging convenzionale (CA).

Per quanto riguarda l’analisi single-trial, il nuovo metodo riesce a riprodurre abbastanza bene la variabilità dei potenziali veri e a fornire delle buone stime. L’analisi dettagliata della componente P300 ha dato come risultato una stima molto vicina al valore vero per quanto riguarda la latenza

e un po' sottostimata per quanto riguarda l'ampiezza. In particolare, si ottengono dei valori molto alti del coefficiente di correlazione tra le latenze vere e stimate della P300.

4. Applicazione a dati reali registrati durante un compito cognitivo

Il capitolo che segue presenta l'applicazione del metodo proposto a dati EEG reali registrati durante un compito cognitivo. Ad una breve descrizione dell'esperimento e del dataset a disposizione, seguiranno i risultati dell'analisi ottenuti per le diverse categorie di soggetti sottoposti al compito. La stima del potenziale evocato medio sarà messa a confronto con il risultato della media convenzionale. Delle stime single-trial sarà invece valutata la complessiva variabilità e saranno stimate media e standard deviation dei parametri che descrivono la componente P300.

4.1 Descrizione del data set

L'esperimento durante il quale vengono registrati i segnale EEG utilizzati in questo lavoro di tesi consiste in un compito cognitivo chiamato compito Simon. Il soggetto, posto in una stanza silenziosa e poco illuminata, è sottoposto a due differenti stimoli visivi che possono presentarsi sulla parte destra o sinistra di un monitor. Ognuno di questi stimoli richiede una differente risposta da parte del soggetto: se lo stimolo è una scacchiera rosso-nera il soggetto deve premere un tasto con la mano sinistra, se la scacchiera è verde-nera il soggetto deve premere un tasto con la mano destra. In base quindi alla posizione sul monitor e al colore dello scacchiera proiettata, è possibile riconoscere quattro diverse condizioni sperimentali come schematizzato in Fig. 4.1.

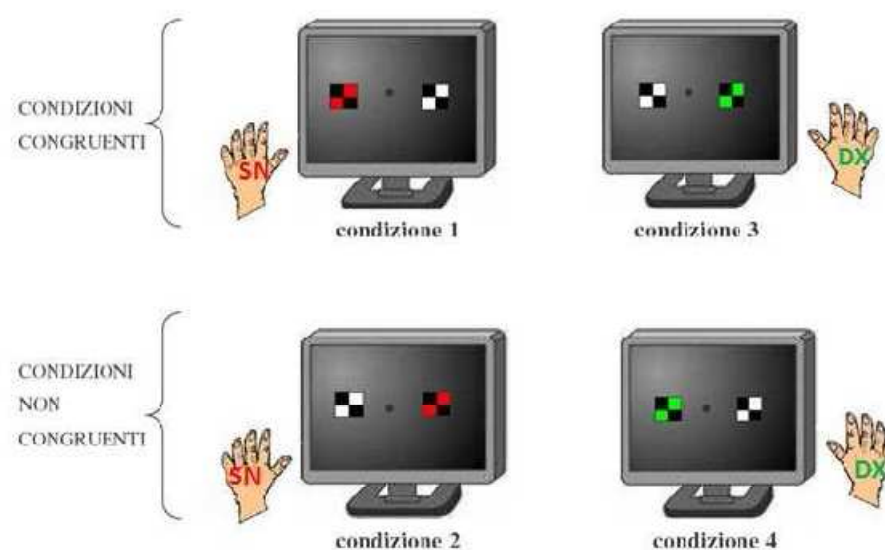


Figura 4.1. Descrizione del compito di Simon.

In particolare, se la posizione dello stimolo sul monitor (sinistra o destra) coincide con la mano da utilizzare nella risposta (sinistra o destra), si parla di condizioni *congruenti*, viceversa si parla di condizioni *non congruenti*. L'evidenza sperimentale osservata stabilisce che i tempi di reazione, ovvero i tempi che soggetti impiegano a premere il tasto corrispondente con la mano ritenuta giusta, risultano più brevi nel caso delle condizioni congruenti rispetto alle non congruenti. Tale fenomeno è noto come effetto Simon, da qui il nome del compito.

Il dataset a nostra disposizione comprende dati EEG registrati su 42 soggetti suddivisi in: 11 giovani sani, 10 anziani sani, 14 cirrotici senza encefalopatia epatica (HE) e 11 cirrotici con encefalopatia epatica. Il segnale EEG è stato registrato con un elettroencefalografo digitale (System II, Micromed, Mogliano Veneto, Italia) per mezzo di una cuffia elastica. Sono stati utilizzati 29 elettrodi Ag/AgCl posizionati secondo il sistema 10-20; come riferimento è stata adottata la media dei due segnali registrati sui lobi delle orecchie. Ogni elettrodo è dotato di un convertitore analogico-digitale con frequenza di campionamento di 512 Hz e risoluzione di 0.9 $\mu\text{V}/\text{digit}$. L'impedenza di elettrodo è stata mantenuta sotto i 5 k Ω ; inoltre sono stati acquisiti gli elettroculogrammi orizzontale e verticale tramite due elettrodi posizionati rispettivamente sul canthus e sotto l'occhio sinistro.

Per semplicità e a titolo esemplificativo, tra tutti i canali a disposizione è stato scelto l'elettrodo Pz e i segnali ivi registrati per la condizione 1 (v. Fig. 4.1) sono stati divisi in sweeps della durata di 1.8 secondi, di cui 0.8 secondi di pre-stimolo, che servirà per dedurre le caratteristiche del rumore, e 1 secondo di post-stimolo in cui ci si aspetta che il potenziale da stimare venga evocato.

Successivamente, ai segnali è stato applicato un filtraggio passa banda (tra 0.7 e 30 Hz) in modo da eliminare sia eventuali trend lenti sia rumori ad alte frequenze che corrompono il tracciato EEG. I trial con artefatti (oculari, muscolari, da rete elettrica etc) non sono stati rimossi ma corretti usando l'ICA offerta dal toolbox di Matlab EEGLAB.

Per ogni soggetto sono state considerate le prime 25 sweeps. Per alleggerire il carico computazionale, ogni sweep è stata sottocampionata a 128 Hz. Inoltre, per la stima degli iperparametri è stato preso un campione ogni 4 in modo da ridurre il tempo di calcolo per stimare le 25 sweeps da decine di minuti a un minuto.

Da conoscenze a priori sull'esperimento, è noto che il potenziale evocato dovrebbe essere nullo prima dello stimolo ed essersi esaurito dopo circa 1 secondo. Per questo si è deciso di porre rispettivamente a zero e a zero con varianza nulla il primo e l'ultimo campione del segnale di post-stimolo, in modo anche da ridurre gli intervalli di confidenza delle stime.

4.2 Risultati dell'analisi per i diversi gruppi di soggetti

4.2.1 Soggetti giovani e sani

Analizziamo ora il primo gruppo, quello dei soggetti giovani e sani, riportando gli esempi di due soggetti rappresentativi.

In fig.4.2 e 4.3 sono riportate le curve medie stimate (in blu) e l'averaging convenzionale (in verde), rispettivamente del soggetto #2 e del soggetto #7.

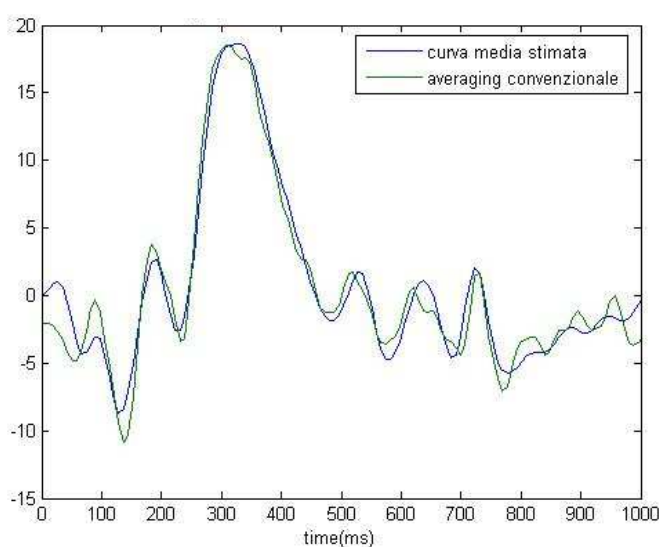


Figura 4.2: Curva stimata del potenziale medio del soggetto #2: curva stimata con il metodo multi-task learning in blu, curva stimata con l'averaging convenzionale in verde.

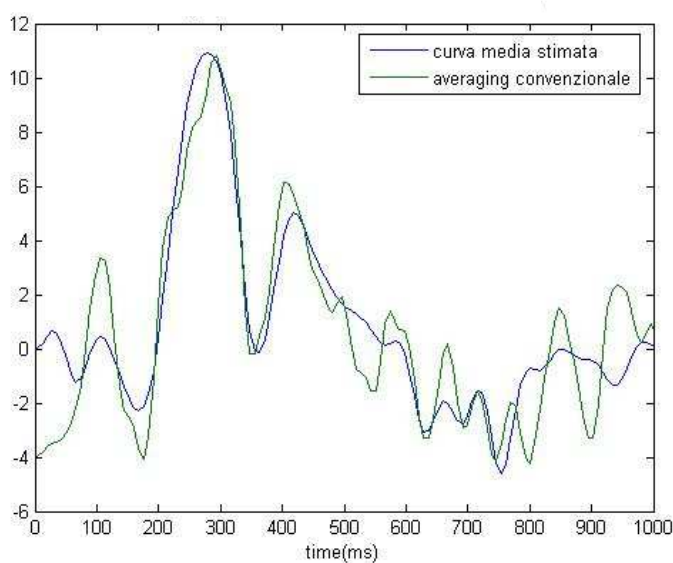


Figura 4.3: Curva stimata del potenziale medio del soggetto #7: curva stimata con il metodo multi-task learning in blu, curva stimata con l'averaging convenzionale in verde.

La stima della curva media è molto buona e segue l'andamento dell'averaging convenzionale (CA) risultando, come previsto, più regolare.

Passiamo ad analizzare le curve individuali stimate. Nello stesso grafico sono state plottate, oltre alla singola sweep grezza (in verde) e alla sweep stimata (in blu), la media stimata (in magenta) e l'averaging convenzionale (in rosso). Per due soggetti rappresentativi vengono visualizzate 2 delle 25 sweeps stimate (fig. 4.4,4.5,4.6,4.8).

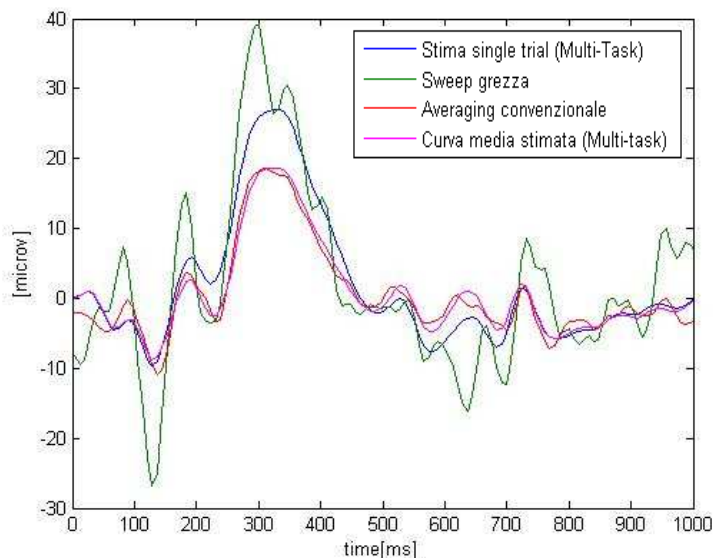


Figura 4.4: Stima single-trial per la sweep #1 del soggetto #2: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

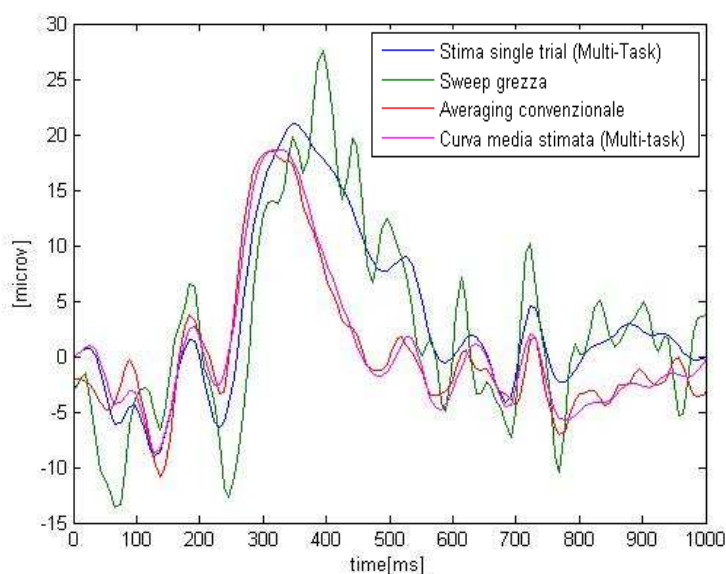


Figura 4.5: Stima single-trial per la sweep #14 del soggetto #2: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

Per il primo soggetto si vede che l’algoritmo riesce a differenziare bene tra la singola sweep stimata e la media. In particolare, si noti che con il CA si ottiene una stima molto più irregolare che si discosta in maniera significativa dai dati.

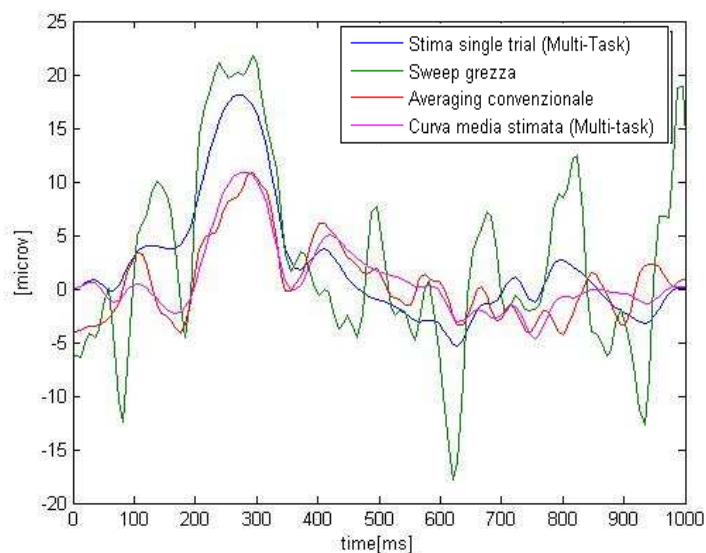


Figura 4.6: Stima single-trial per la sweep #17 del soggetto #7: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l’averaging convenzionale in rosso, sweep grezza in verde.

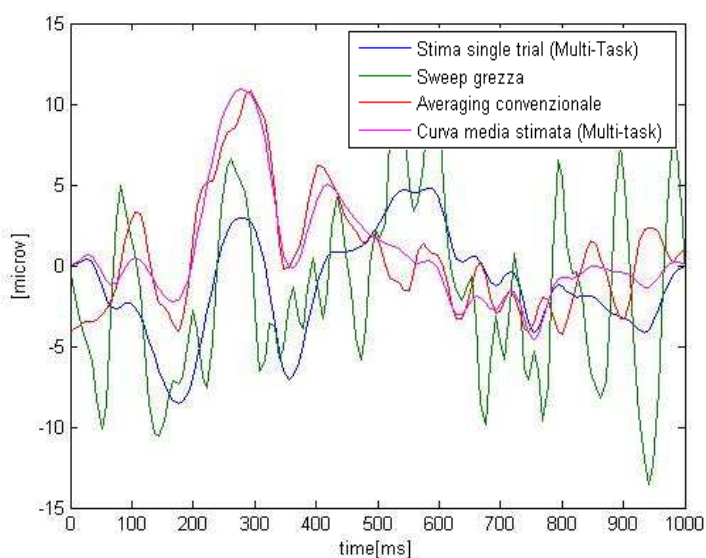


Figura 4.7: Stima single-trial per la sweep #23 del soggetto #7: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l’averaging convenzionale in rosso, sweep grezza in verde.

Anche per quanto riguarda il soggetto #7, l’algoritmo fornisce una stima regolare che segue i dati e si discosta dalla media. Si noti che, nei due soggetti rappresentativi, le stime sono ben regolarizzate e ci permettono di individuare bene il picco principale di questo tipo di segnali (P300). Come ulteriore verifica, riportiamo in fig 4.8 e 4.9 tutte le sweeps stimate per i due soggetti rappresentativi in modo da valutarne la variabilità.

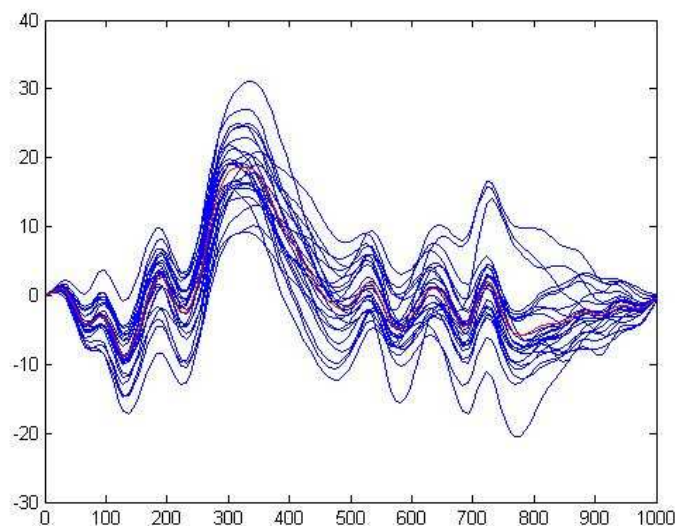


Figura 4.8: Variabilità tra le singole sweeps stimate, soggetto giovane e sano #2 (in rosso è rappresentata la curva media stimata).

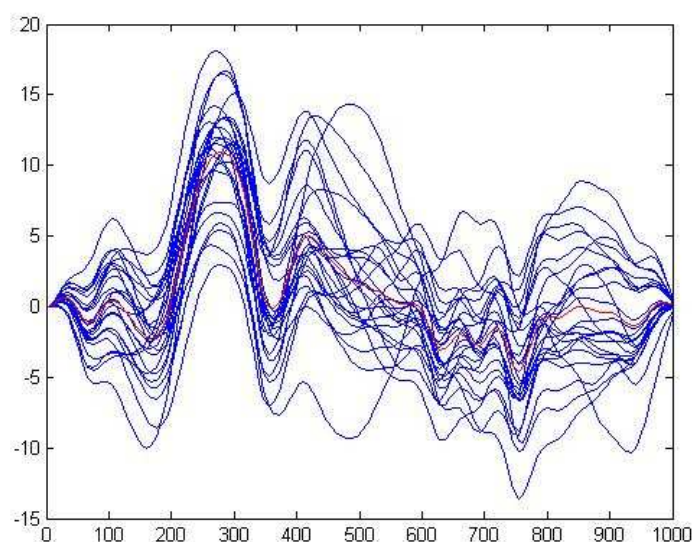


Figura 4.9: Variabilità tra le singole sweeps stimate, soggetto giovane e sano #7 (in rosso è rappresentata la curva media stimata).

Dai grafici, si vede che le stime single-trial non sono troppo vicine tra loro. Sembra, quindi, che l'algoritmo riesca a stimare bene la variabilità che c'è tra le varie risposte.

Riportiamo nella tabella 4.1 i valori medi di ampiezza e latenza della componente P300 di ogni soggetto.

	Latenza media [ms]	Sd latenza [ms]	Ampiezza media [μv]	Sd ampiezza [μv]
g1	377.3438	78.4171	9.445	2.9103
g2	356.4063	111.364	19.5224	4.7045
g3	423.9063	150.8217	9.2046	3.1331
g4	464.8438	192.2018	8.1859	1.6113
g5	339.2188	136.046	9.4331	3.1986
g6	302.3438	3.9063	15.9923	2.8351
g7	367.6563	156.0559	11.7454	3.3993
g8	389.2188	199.9924	11.9259	4.8512
g9	440.4688	220.1559	9.129	5.3223
g10	398.5938	166.4344	13.6175	4.4818
g11	467.0313	220.1582	12.3245	5.3247
media	393.3665	148.6867	11.86596	3.797473

Tabella 4.1: Valori medi di ampiezza e latenza della P300 e relativa standard deviation, soggetti giovani e sani.

Non avendo a disposizione il reale andamento del potenziale evocato, non possiamo affermare che i valori siano accurati, ma possiamo confrontarli con quelli ottenuti tramite altri metodi single-trial. La media tra tutti i soggetti può avere poi una valenza significativa se confrontata con le medie degli altri gruppi.

4.2.2 Soggetti anziani e sani

Passiamo ad analizzare il secondo gruppo, quello dei soggetti anziani e sani, riportando due soggetti rappresentativi.

In fig. 4.10 e 4.11 sono riportate le curve medie stimate (in blu) e l'averaging convenzionale (in verde), rispettivamente del soggetto #1 e del soggetto #2.

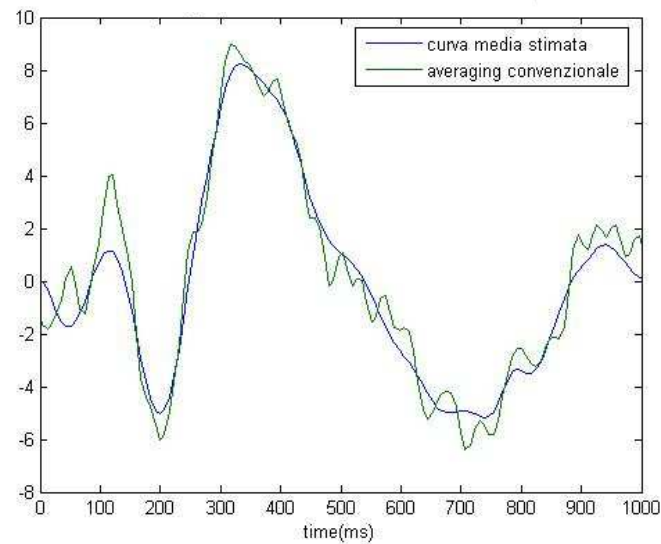


Figura 4.10: Curva stimata del potenziale medio del soggetto #1: curva stimata con il metodo multi-task learning in blu, curva stimata con l'averaging convenzionale in verde.

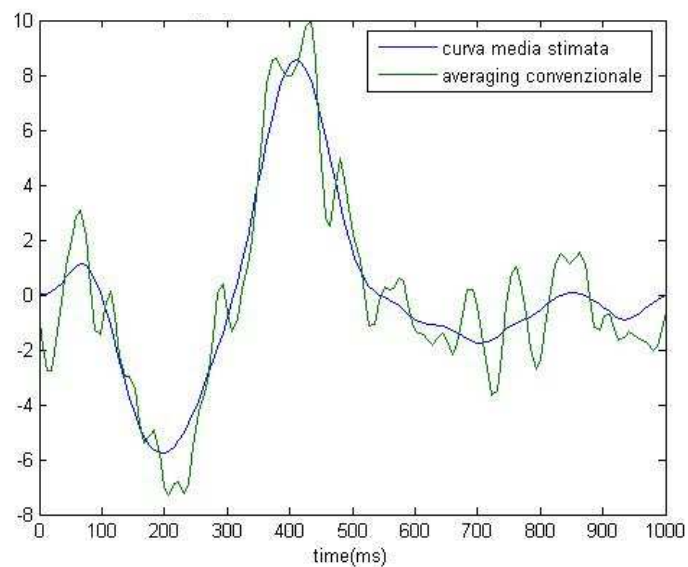


Figura 4.11: Curva stimata del potenziale medio del soggetto #2: curva stimata con il metodo multi-task learning in blu, curva stimata con l'averaging convenzionale in verde.

Anche in questo caso, la stima della curva media è molto buona e segue l'andamento dell'averaging convenzionale (CA) risultando, come previsto, più regolare.

Analizziamo ora le curve individuali stimate (fig. 4.12-4.15). Nello stesso grafico ci sono, oltre alla singola sweep grezza (in verde) e alla sweep stimata (in blu), la media stimata (in magenta) e l'averaging convenzionale (in rosso).

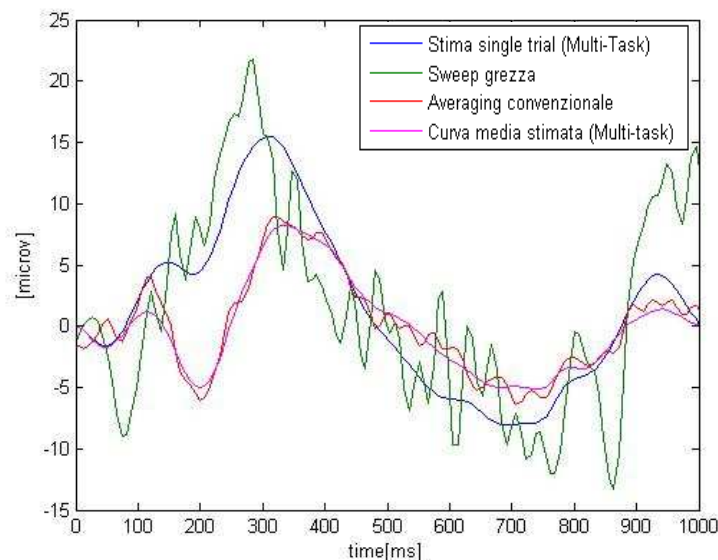


Figura 4.12 Stima single-trial per la sweep #1 del soggetto #1: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

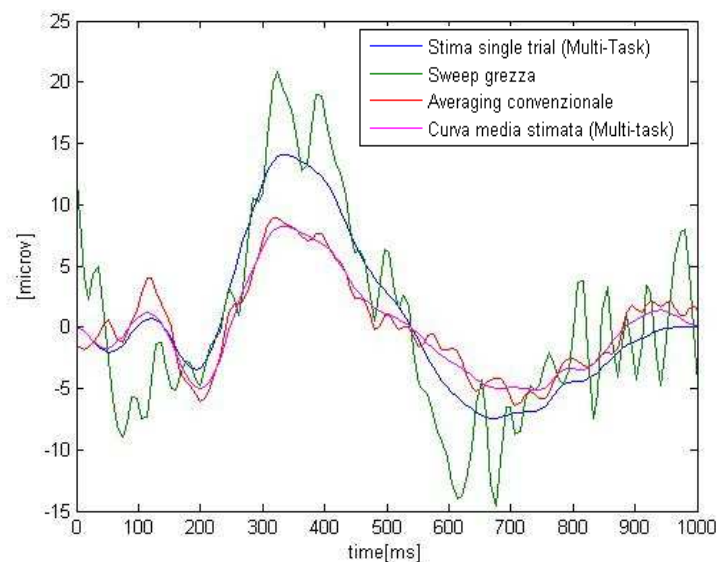


Figura 4.13: Stima single-trial per la sweep #13 del soggetto #1: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

Le figure 4.12 e 4.13 mostrano come, per un primo soggetto rappresentativo, siano state ottenute delle buone stime che si discostano dalla curva media e in cui le componenti principali del segnale sono ben individuabili. Passiamo ora ad un secondo soggetto rappresentativo.

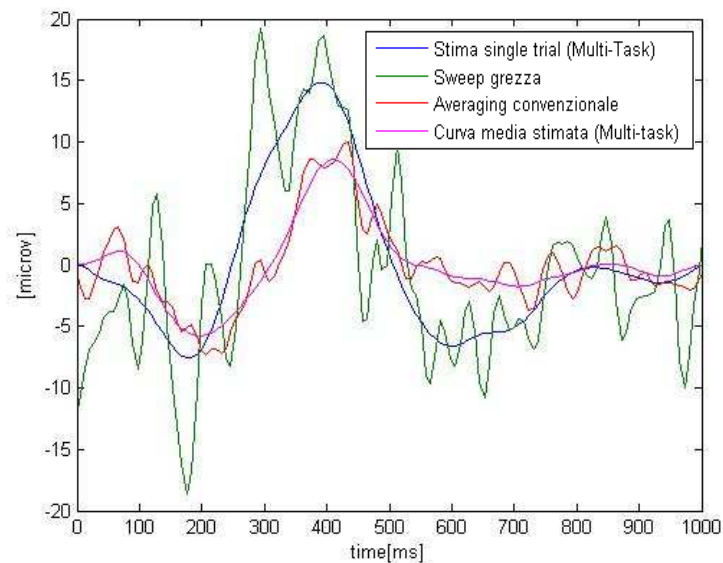


Figura 4.14: Stima single-trial per la sweep #8 del soggetto #2: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

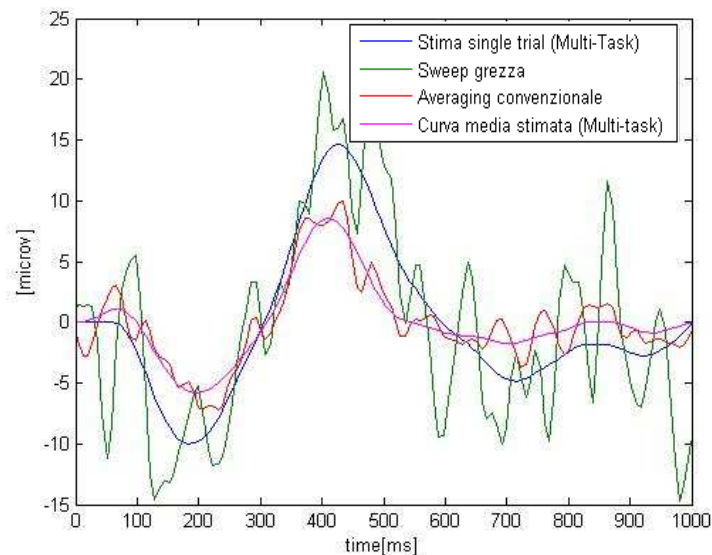


Figura 4.15: Stima single-trial per la sweep #18 del soggetto #2: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

Anche in questo caso si possono fare le stesse considerazioni che sono state fatte per il soggetto precedente. In particolare, si noti ancora come le stime date dal nostro algoritmo siano decisamente migliori di quelle fornite dal CA.

Come ulteriore prova, riportiamo in fig 4.16 e 4.17 tutte le sweeps stimate in modo da valutarne la variabilità (in rosso è rappresentata la curva media stimata).

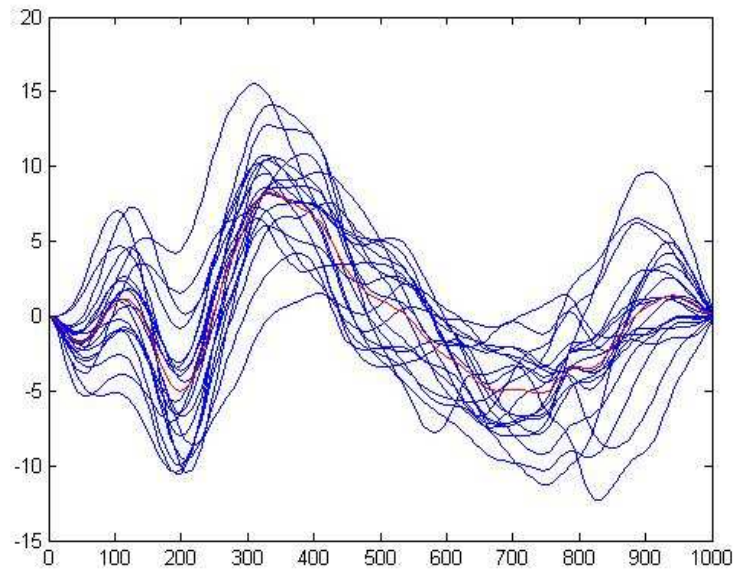


Figura 4.16: variabilità tra le singole sweeps stimate, soggetto giovane e sano #1 (in rosso è rappresentata la curva media stimata).

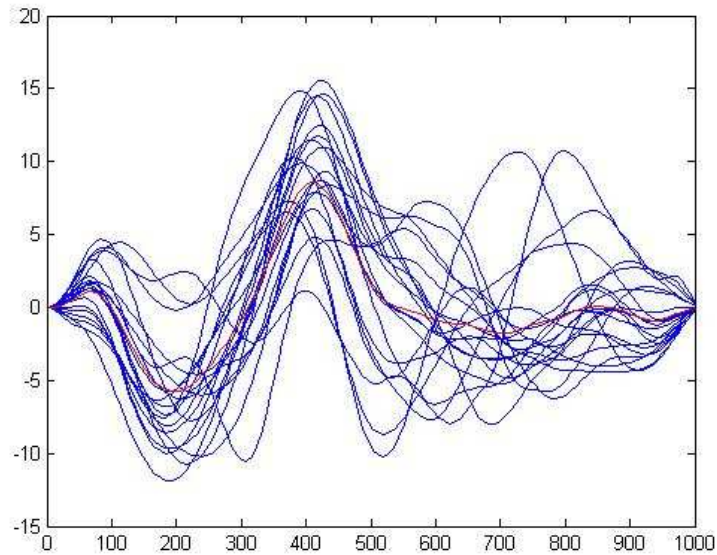


Figura 4.17: variabilità tra le singole sweeps stimate, soggetto giovane e sano #2 (in rosso è rappresentata la curva media stimata).

La variabilità tra le stime è evidente sia nel primo che nel secondo soggetto. Si noti poi che le stime partono tutte da zero e finiscono in zero. Questa forzatura è stata aggiunta nell'algoritmo perché è stato assunto che il potenziale fosse nullo prima dello stimolo e che si fosse esaurito dopo non più di un secondo.

Per concludere, vengono riportati nella tabella 4.2 i valori medi di latenza e ampiezza della P300 di ogni soggetto, compresa la media del gruppo.

	Latenza media [ms]	Sd latenza [ms]	Ampiezza media [μv]	Sd ampiezza [μv]
a1	404.2969	178.6723	9.1768	3.0122
a2	453.5156	111.8285	10.2755	3.1503
a3	361.7188	13.5554	16.6574	4.1976
a4	477.3438	167.8927	8.2095	2.8286
a5	340.2344	32.9975	14.9159	3.1633
a6	368.75	147.2119	9.3702	2.9238
a7	507.4219	153.5221	6.3714	2.1457
a8	378.125	172.8423	8.5166	3.1999
a9	300.3906	130.1981	6.7029	2.1114
a10	337.5	154.0365	9.5996	2.3463
media	399.0886	123.1912	10.0218	2.970311

Tabella 4.2: Valori medi di ampiezza e latenza della P300 e relativa standard deviation, soggetti anziani e sani.

4.2.3 *Soggetti cirrotici senza encefalopatia epatica (HE)*

Passiamo adesso al terzo gruppo, quello dei soggetti cirrotici senza encefalopatia epatica, riportando ancora gli esempi di due soggetti rappresentativi.

In fig. 4.18 e 4.19 sono riportate le curve medie stimate (in blu) e l'averaging convenzionale (in verde), rispettivamente del soggetto #4 e del soggetto #9.

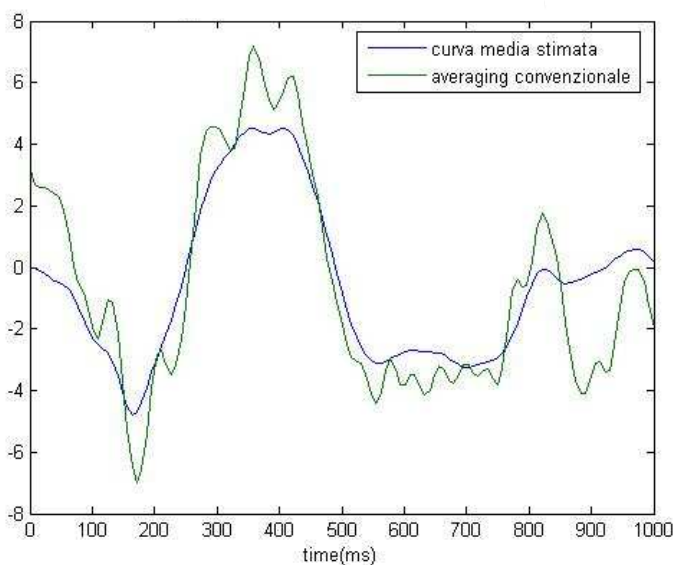


Figura 4.18: Curva stimata del potenziale medio del soggetto #4: curva stimata con il metodo multi-task learning in blu, curva stimata con l'averaging convenzionale in verde.

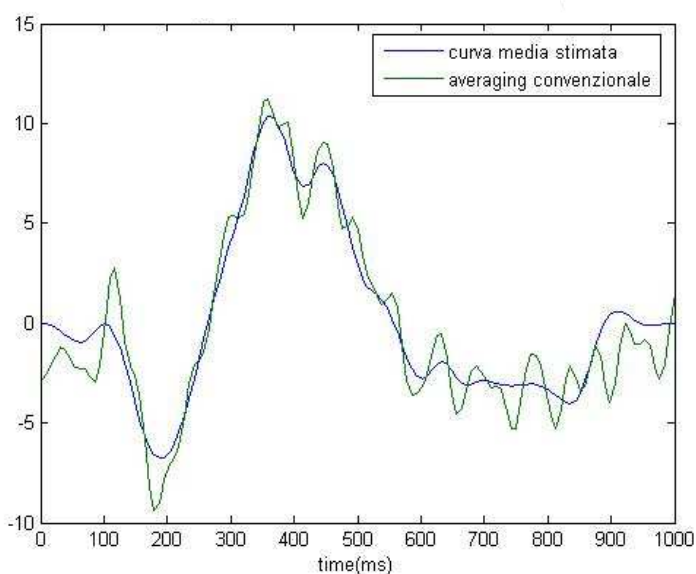


Figura 4.19: Curva stimata del potenziale medio del soggetto #9: curva stimata con il metodo multi-task learning in blu, curva stimata con l'averaging convenzionale in verde.

L'andamento della curva media stimata è ancora molto buono e migliore rispetto a quello del CA.

Qui di seguito sono riportate le curve individuali stimate. Nello stesso grafico sono state plottate, oltre alla singola sweep grezza (in verde) e alla sweep stimata (in blu), la media stimata (in magenta) e l'averaging convenzionale (in rosso).

I risultati della stima per due sweeps rappresentative del primo soggetto sono mostrati in fig. 4.20 e 4.21, del secondo soggetto in fig. 4.22 e 4.23.

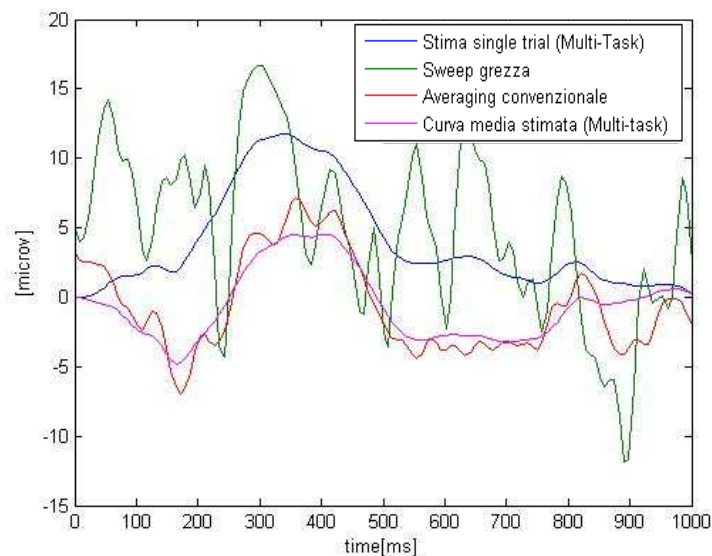


Figura 4.20: Stima single-trial per la sweep #10 del soggetto #4: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

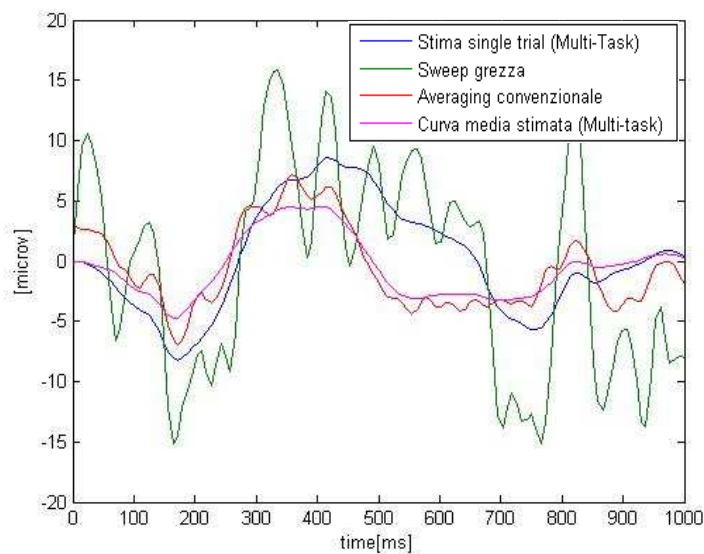


Figura 4.21: Stima single-trial per la sweep #18 del soggetto #4: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

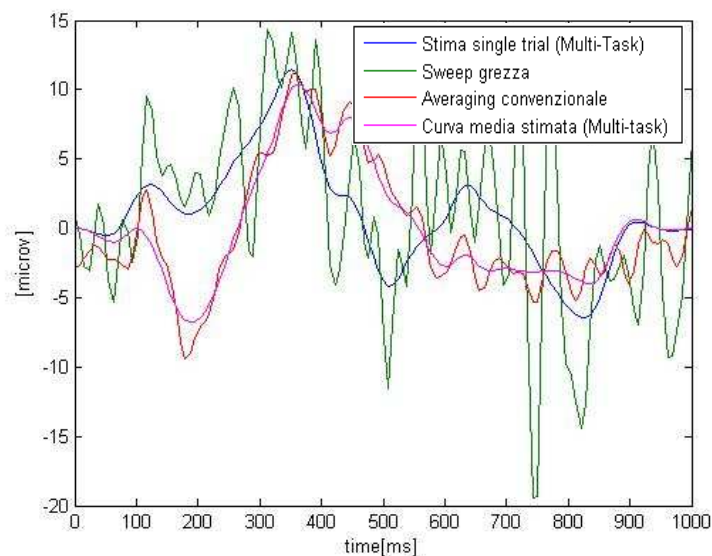


Figura 4.22: Stima single-trial per la sweep #9 del soggetto #9: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

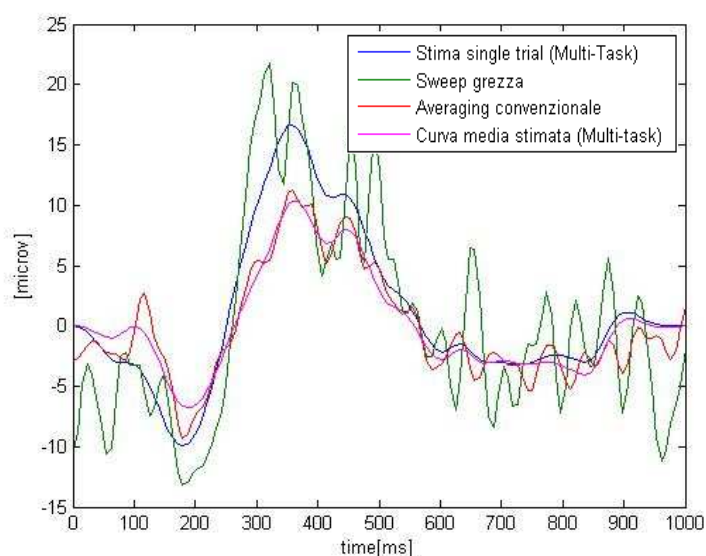


Figura 4.23: Stima single-trial per la sweep #18 del soggetto #9: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

Come si può vedere dai grafici dei due soggetti rappresentativi, il nuovo metodo riesce ad ottenere un profilo regolare che segue i dati. Inoltre, confrontando questo profilo con la stima fornita dal CA, si può notare che l'algoritmo tiene conto della variabilità tra sweep e sweep. Come ulteriore verifica, riportiamo in fig 4.24 e 4.25 tutti i potenziali stimati in modo da valutarne la variabilità (in rosso è rappresentata la curva media stimata).

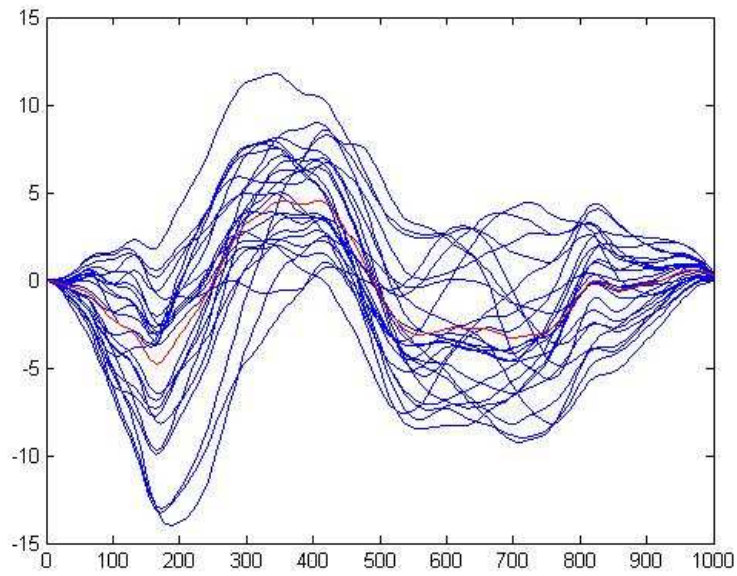


Figura 4.24: Variabilità tra le singole sweeps stimate, soggetto malato senza HE #4 (in rosso è rappresentata la curva media stimata).

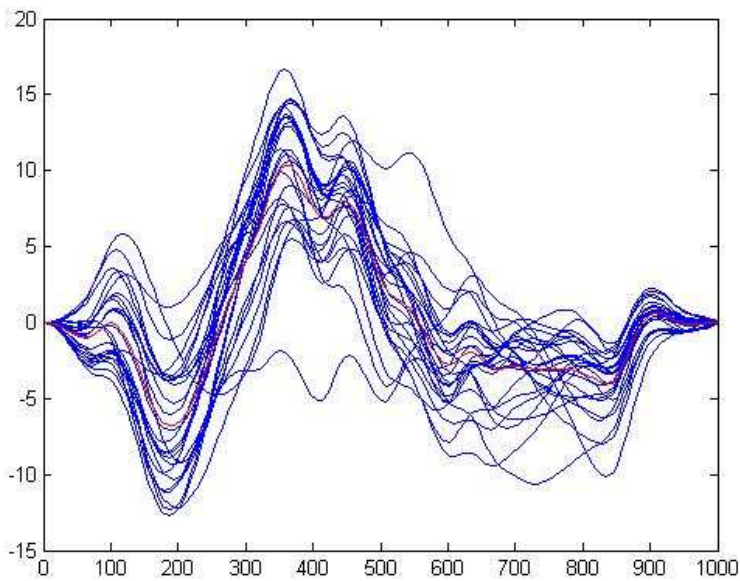


Figura 4.25: Variabilità tra le singole sweeps stimate, soggetto malato senza HE #9 (in rosso è rappresentata la curva media stimata).

Dai precedenti grafici, si vede che le stime single-trial non “collassano” sul potenziale medio. Sembra quindi che anche in questo caso l’algoritmo riesca a stimare bene la variabilità che c’è tra le varie registrazioni.

Infine, riportiamo nella tabella 4.3 i valori medi di ampiezza e latenza della componente P300 per ogni soggetto di questo gruppo.

	Latenza media [ms]	Sd latenza [ms]	Ampiezza media [μv]	Sd ampiezza [μv]
m1	401.875	101.5395	7.4039	2.1577
m2	408.125	68.9347	5.8275	2.3399
m3	344.375	217.2442	6.3669	2.7416
m4	398.125	92.2868	5.6399	2.626
m5	452.5	135.7653	7.7193	2.6566
m6	400.3125	163.6789	4.5103	1.6583
m7	374.6875	189.758	5.9548	2.7432
m8	580.9375	142.1875	7.8291	3.0324
m9	371.5625	88.7639	11.1819	3.5164
m10	394.6875	66.7927	8.4299	2.4943
m11	397.5	199.1079	10.5344	3.8439
m12	540	162.0242	5.6018	2.635
m13	341.875	200.4187	6.5392	2.3674
m14	205.3125	104.5124	10.8987	2.6719
media	400.8482	138.0725	7.459829	2.677471

Tabella 4.3: Valori medi di ampiezza e latenza della P300 e relativa standard deviation, soggetti cirrotici senza HE.

4.2.4 Soggetti cirrotici con encefalopatia epatica (HE)

Concludiamo con l'ultimo gruppo, quello dei soggetti cirrotici con HE, riportando come esempio due soggetti rappresentativi.

In fig.4.26 e 4.27 sono riportate le curve medie stimate (in blu) e l'averaging convenzionale (in verde), rispettivamente del soggetto #7 e del soggetto #9.

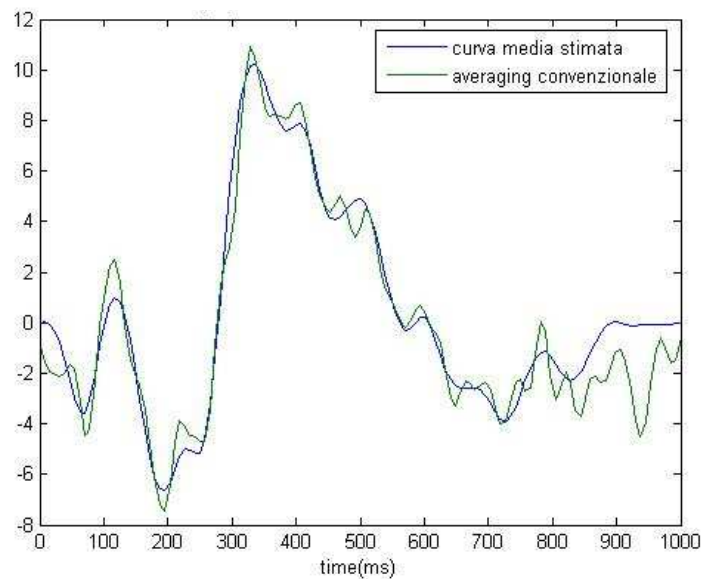


Figura 4.26: Curva stimata del potenziale medio del soggetto #7: curva stimata con il metodo multi-task learning in blu, curva stimata con l'averaging convenzionale in verde.

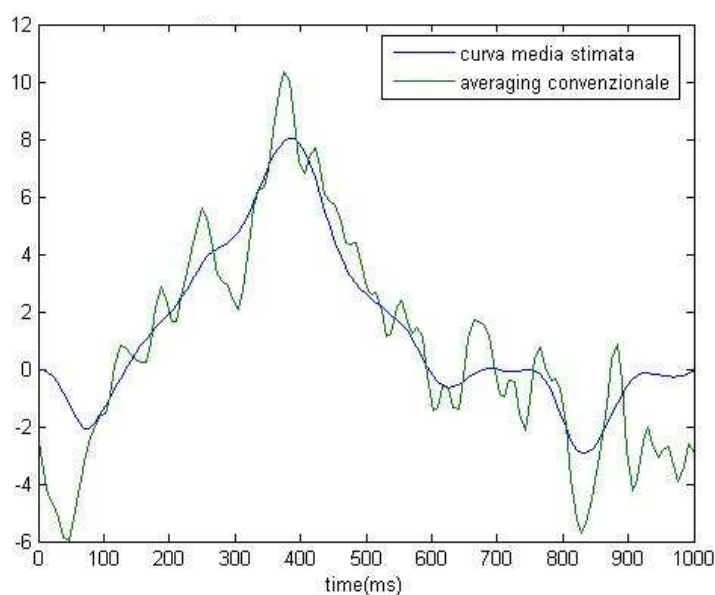


Figura 4.27: Curva stimata del potenziale medio del soggetto #9: curva stimata con il metodo multi-task learning in blu, curva stimata con l'averaging convenzionale in verde.

La stima della curva media è molto buona e segue l'andamento dell'averaging convenzionale (CA) risultando, come previsto, più regolare.

Passiamo per l'ultima volta ad analizzare le curve individuali stimate. Nello stesso grafico sono state plottate, oltre alla singola sweep grezza (in verde) e alla sweep stimata (in blu), la media stimata (in magenta) e l'averaging convenzionale (in rosso). Per il primo soggetto i risultati sono mostrati in fig. 4.28 e 4.29, per il secondo in fig. 4.30 e 4.31.

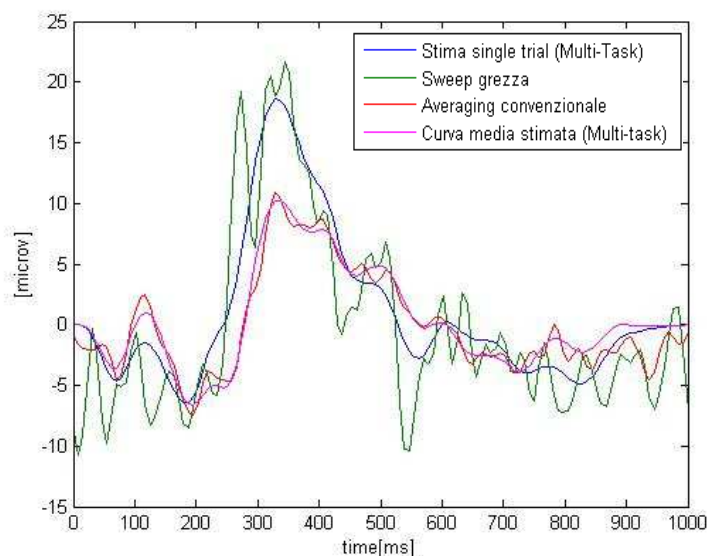


Figura 4.28: Stima single-trial per la sweep #2 del soggetto #7: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

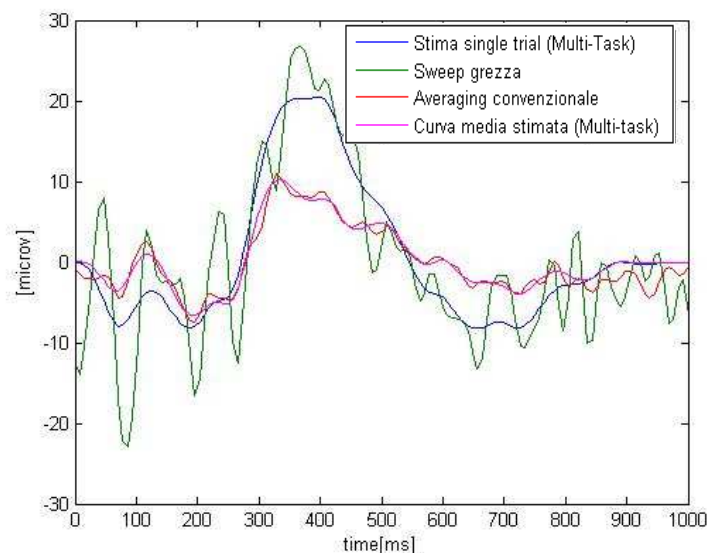


Figura 4.29: Stima single-trial per la sweep #25 del soggetto #7: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

Nel primo soggetto si vede come l'algorithmo riesca a differenziare bene tra la singola sweep stimata e la media. In particolare, si noti che con il CA si ottiene una stima molto più irregolare che si discosta in maniera significativa dai dati.

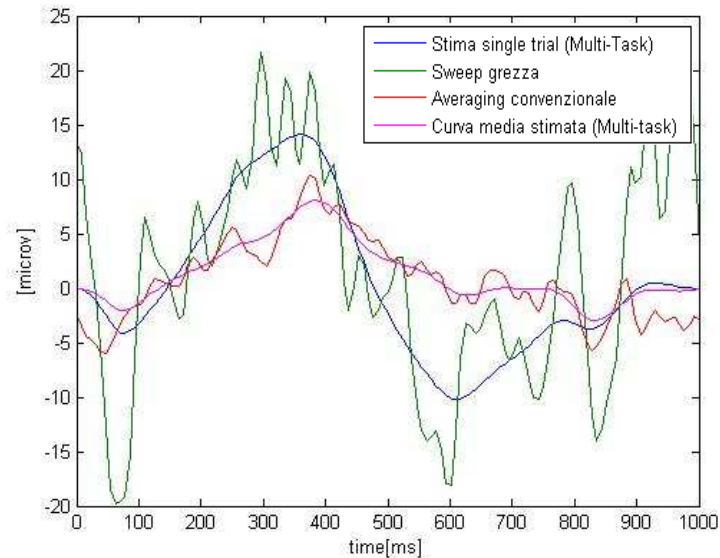


Figura 4.30: Stima single-trial per la sweep #17 del soggetto #9: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

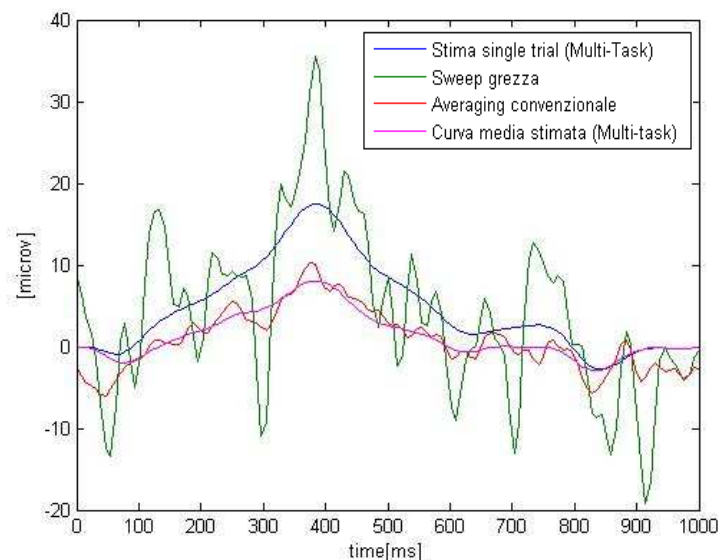


Figura 4.31: Stima single-trial per la sweep #19 del soggetto #9: curva stimata con il metodo multi-task learning in blu, curva media stimata con il metodo multi-task learning in magenta, curva media stimata con l'averaging convenzionale in rosso, sweep grezza in verde.

Anche per quest'ultimo soggetto le stime delle singole registrazioni risultano molto buone. Inoltre, si noti che le sweeps registrate sono molto rumorose ed è quindi molto difficile riuscire ad ottenere il profilo regolare del solo potenziale evocato. Dopo l'analisi di tutte le curve

individuali, si può affermare che il nuovo algoritmo fornisca una stima molto più accurata di quella che si può ottenere con il CA.

Vediamo, anche in questo caso, se il nuovo metodo riesce a stimare bene la variabilità tra le varie registrazioni (fig. 4.32 e 4.33).

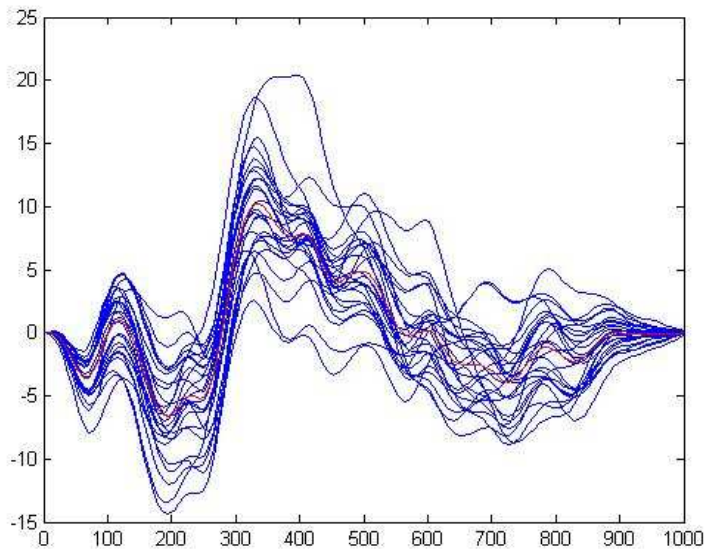


Figura 4.32: Variabilità tra le singole sweeps stimate, soggetto malato con HE #7 (in rosso è rappresentata la curva media stimata).

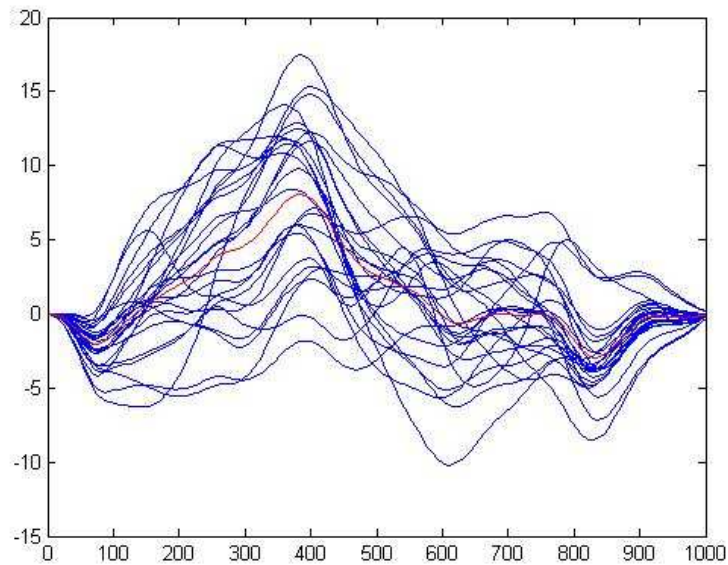


Figura 4.33: Variabilità tra le singole sweeps stimate, soggetto malato con HE #9 (in rosso è rappresentata la curva media stimata).

Anche qui si vede che le singole stime non “collassano” sulla media, ottenendo così dei profili diversi tra i vari segnali stimati (in rosso è rappresentata la curva media stimata).

Concludiamo questa sezione, relativa ai risultati dell’applicazione dell’algoritmo ai dati reali, con i valori medi della latenza e dell’ampiezza della componente P300 (tabella 4.4).

	Latenza media [ms]	Sd latenza [ms]	Ampiezza media [μv]	Sd ampiezza [μv]
mhe1	350.625	208.0644	5.2211	1.9781
mhe2	348.125	190.4483	4.4682	1.9651
mhe3	204.375	156.6999	1.9463	0.479
mhe4	323.4375	169.9555	8.2784	3.811
mhe5	424.375	186.0119	4.5258	1.8
mhe6	131.875	2.5911	12.0014	1.2587
mhe7	359.0625	78.692	10.8401	3.8764
mhe8	381.875	89.8528	5.179	1.4747
mhe9	427.1875	139.5671	9.1457	4.4516
mhe10	375.9375	177.7139	8.0417	3.1826
mhe11	460.3125	155.6943	12.3498	3.1261
media	344.2898	141.3901	7.454318	2.491209

Tabella 4.4: Valori medi di ampiezza e latenza della P300 e relativa standard deviation, soggetti cirrotici con HE.

Non si può ancora affermare che i valori ottenuti siano vicini a quelli veri perché non siamo in possesso dei profili reali, però si può dire che risultano abbastanza vicini a quelli ottenuti nelle simulazioni. Infine, questi valori possono essere confrontati con quelli ottenuti da altri metodi single-trial applicati allo stesso data set reale.

4.3 Conclusioni

In questo capitolo sono stati illustrati i risultati dell’applicazione del metodo multi-task learning della stima single-trial di potenziali evocati cognitivi registrati durante un compito di Simon. Dai risultati ottenuti, si può asserire che l’algoritmo fornisce delle stime molto buone, sia per quanto riguarda il potenziale medio sia per quanto riguarda i singoli potenziali evocati.

È importante notare che la stima del potenziale medio fornita dall'approccio di popolazione risulta ben più regolare di quella ottenuta tramite CA, per il quale sarebbe servito un numero maggiore di sweeps per ottenere gli stessi risultati.

Inoltre, la possibilità di stimare i PE single-trial ha consentito la descrizione della variabilità delle risposte di ogni soggetto analizzato.

Rispetto al lavoro documentato in [1] sono stati fatti i seguenti miglioramenti.

Prima di tutto è stato ridotto il tempo di attesa per avere le stime d'interesse. Nella versione precedente bisognava aspettare almeno 40 minuti per ottenere le curve individuali di un soggetto. Nel nostro caso, dopo aver diminuito il numero di sweeps impiegate e aver sottocampionato i segnali nella parte più onerosa computazionalmente, l'algoritmo forniva le stime di 25 trials in circa un minuto.

Per quanto riguarda i valori iniziali e finali dei segnali da stimare, essi sono stati posti uguali a zero perché si considera che il potenziale sia nullo prima dello stimolo e che si sia esaurito in meno di un minuto. In questo modo, cosa che non succedeva in [1], tutti i segnali stimati partono da zero e si esauriscono dopo un minuto.

Infine, in questo lavoro è stata estesa l'applicazione ai dati reali: alla categoria dei soggetti giovani e sani, già analizzata in [1], sono stati aggiunti gruppi di soggetti anziani e di soggetti cirrotici con e senza encefalopatia epatica.

4.4 Problemi aperti

In questa sezione si riporta un problema che è stato riscontrato nella stima degli iper-parametri. In circa la metà dei pazienti utilizzati come data set reale, la stima dell'iper-parametro $\tilde{\lambda}^2$, che descrive la regolarità del modello delle curve individuali, risultava molto piccola. Di conseguenza, le stime individuali non si discostavano dalla stima del potenziale medio e, come si vede dalla figura 4.34, la variabilità tra le singole sweeps risultava quasi nulla. Probabilmente, in questa situazione l'algoritmo tendeva a confondere la variabilità tra le curve individuali con il rumore da eliminare.

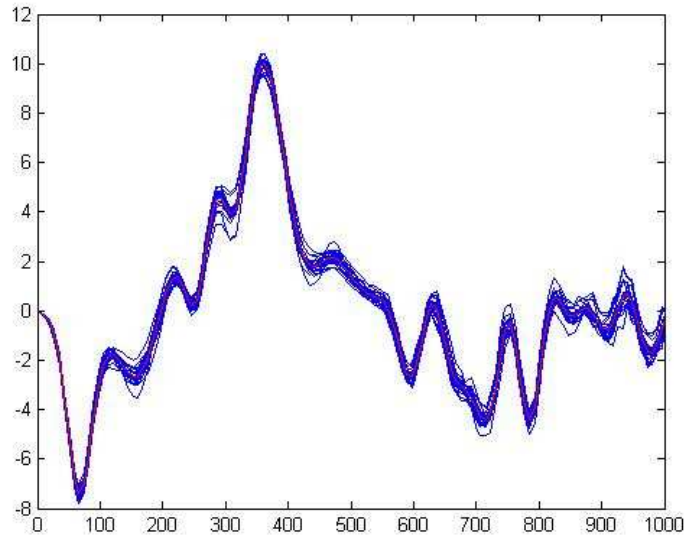


Figura 4.34: Variabilità tra le singole sweeps stimate con $\tilde{\lambda}^2 = 2.36 \times 10^{19}$.

Al fine di capire come mai in questi soggetti la minimizzazione restituisca tali valori dei parametri, la funzione obiettivo è stata valutata in funzione di $\tilde{\lambda}^2$ e $\tilde{\lambda}^2$. Il relativo grafico è riportato in fig. 4.35.

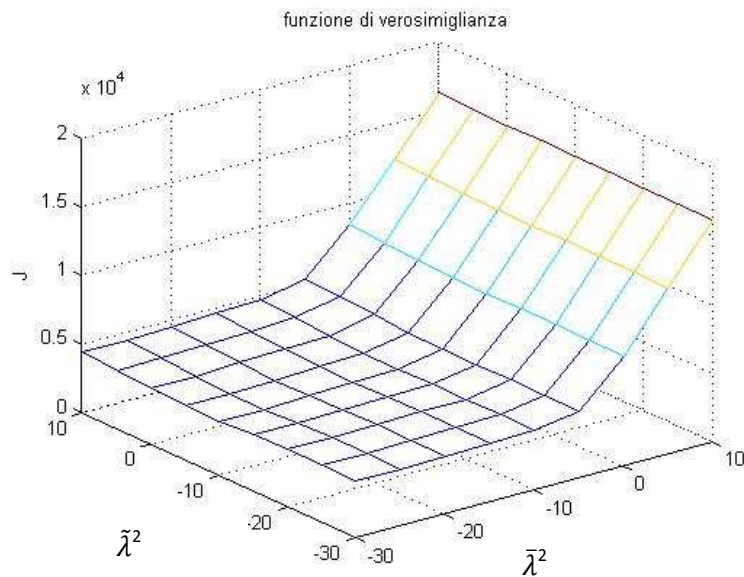


Figura 4.35: Funz. obiettivo al variare degli iper-parametri.

Come si può vedere, la funzione non presenta un punto di minimo ben definito, ma un plateau che rende viziata l'individuazione del minimo. Tale comportamento si potrebbe attribuire ad una eccessiva variabilità delle sweeps.

Per il momento, allo scopo di risolvere questo problema, si è pensato di ricorrere ad una forzatura dei parametri. In particolare, un valore di $\tilde{\lambda}^2$ che risultava molto piccolo (dell'ordine di

$10^{-18}/10^{-20}$), veniva sostituito con uno medio, prendendo in considerazione i casi in cui la stima era andata a buon fine.

In fig. 4.36, si riporta la distribuzione di $\tilde{\lambda}^2$ nei casi in cui la stima si poteva considerare soddisfacente.

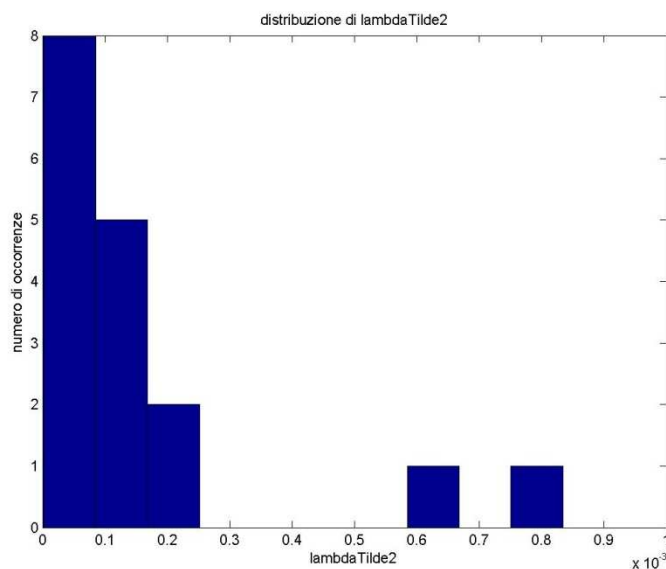


Figura 4.36: distribuzione di $\tilde{\lambda}^2$ nei casi in cui si otteneva una buona stima.

Come si può vedere dall'istogramma, il valore medio che si può utilizzare per $\tilde{\lambda}^2$ è circa 10^{-4} .

Pur avendo attribuito un valore medio agli iper-parametri, le stime fornite dal nuovo algoritmo sembrano ancora buone e simili ai risultati delle simulazioni. In particolare, nei primi due gruppi, sono stati usati come soggetti rappresentativi un caso in cui la stima di $\tilde{\lambda}^2$ era andata a buon fine (soggetto #2 per il primo gruppo e soggetto #2 per il secondo) e un caso in cui si è dovuti ricorrere ad un valore medio (soggetto #7 per il primo gruppo e soggetto #1 per il secondo).

6. Conclusioni e sviluppi futuri

In questo lavoro di tesi è stata presentata una nuova tecnica di stima dei potenziali evocati che utilizza un approccio di tipo multi-task learning. La tecnica è stata già utilizzata in [1], ottenendo risultati preliminari di interesse ma evidenziando anche alcuni problemi aperti. In particolare, i tempi di attesa per avere le stime erano troppo lunghi, le condizioni iniziali e finali per i segnali stimati non erano state ancora ben definite e le situazioni in cui il rumore sovrapposto risultava troppo elevato portavano ad una stima single-trial che non si discostava dal potenziale medio.

L'andamento del potenziale evocato medio e dei potenziali evocati single-trial vengono stimati sfruttando l'informazione derivante dall'intero dataset a disposizione, ricorrendo all'ipotesi che le N risposte evocate di un partecipante ad un esperimento possano essere considerate come soggetti appartenenti ad una popolazione omogenea. Per quanto sappiamo, il multi-task learning è sempre stato applicato, di norma, a casi in cui si dispone di pochi campioni per soggetto. Per la prima volta si è cercato di applicare questo tipo di approccio a segnali che sono stati campionati frequentemente, ma che contengono un rumore sovrapposto (l'EEG di fondo) molto elevato. In particolare, si è scelto di sviluppare ed implementare il metodo recentemente proposto in (Pillonetto et al.,2007) [4] che risulta ben applicabile al caso in esame, in quanto fa solo deboli assunzioni sulla regolarità dei segnali da stimare e perciò non pone vincoli alla forma dei PE incogniti.

Per valutare la bontà del metodo, è stato costruito un dataset simulato che ricrei una situazione reale. I risultati hanno mostrato che il metodo riesce a stimare in modo soddisfacente il PE medio, anche con un numero ridotto di sweeps a disposizione (in questo caso 25). Anche le stime dell'andamento dei PE single-trial e i risultati sui parametri di interesse sembrano buoni. In particolare le curve ottenute sono risultate molto simili ai singoli potenziali veri e il coefficiente di correlazione tra latenze vere e stimate della P300 è risultato molto alto (0.93). Infine, l'applicazione ad un caso con rumore elevato ha dato ancora dei buoni risultati sia per la curva media che per le stime single-trial. Questo aspetto è importante perché, nel precedente lavoro [1], la stima single-trial nel caso con rumore elevato non era andata a buon fine, fornendo delle stime individuali che "collassavano" nella curva media.

Dopo le simulazioni, si è passati all'applicazione ai dati reali e, in particolare, a segnali EEG acquisiti durante un compito di Simon su alcuni soggetti appartenenti a categorie diverse. I tempi di attesa sono stati notevolmente ridotti in quanto, in [1], si ottenevano le stime single-trial di un soggetto in 40 minuti mentre in questo caso l'algoritmo calcola le curve individuali di un soggetto in circa 1 minuto. I risultati si possono considerare molto promettenti poiché la variabilità delle stime single trial ottenute e il loro discostarsi dal PE medio sembra essere descrittivo di ciò che si trova tipicamente nei casi reali. Il metodo qui presentato può essere quindi un valido strumento di analisi dei potenziali evocati sia perché permette di stimare un PE medio con un numero di sweep relativamente basso (riducendo i tempi di registrazione molto spesso troppo lunghi) sia perché fornisce le stime single-trial dalle quali è possibile ricavare importanti informazioni cliniche sullo stato di salute del sistema nervoso centrale e periferico, nonché utili conoscenze per lo studio dei processi cognitivi. Inoltre, poiché le assunzioni sulla

regolarità dei segnali da stimare sono molto blande, il metodo risulta anche estremamente flessibile e facilmente estendibile ad altri tipi di risposte evocate.

In conclusione è stato fatto uno studio sui valori medi di ampiezza e latenza della P300 per i soggetti di ogni categoria. Da questa analisi si è visto che l'ampiezza della P300 nel primo gruppo (giovani e sani) ha un valore che si riduce man mano che si passa ai gruppi successivi. Ciò porta a concludere che in soggetti sani e giovani si riscontrano potenziali evocati di ampiezza maggiore rispetto ai soggetti anziani e malati. Per quanto riguarda la latenza media, si è visto che essa aumenta nel secondo e terzo gruppo (anziani e cirrotici senza HE) mentre diminuisce nell'ultimo (cirrotici con HE).

Successivi sviluppi riguarderanno in particolare lo studio approfondito dei casi in cui si sono riscontrati dei problemi nella stima degli iper-parametri che descrivono la regolarità dei segnali incogniti. Inoltre, un confronto, peraltro già avviato, con altri metodi presenti in letteratura, si renderà necessario per collocare la tecnica nel vasto ambito della stima dei potenziali evocati. Sarebbe inoltre interessante provare ad applicare alla stima dei potenziali evocati l'algoritmo di multi-task learning recentemente presentato in (Dinuzzo et al.,2008) [5], che permette di ottenere delle stime on-line.

Appendice A

A.1 Definizioni

Definiamo i seguenti vettori:

$$\begin{aligned}
 \bar{\mathbf{z}} &= [\bar{z}(t_1) \dots \bar{z}(t_n)]^T && \text{di dimensione } n \times 1 \\
 \tilde{\mathbf{z}}^j &= [\tilde{z}^j(t_1) \dots \tilde{z}^j(t_n)]^T && \text{di dimensione } n \times 1 \quad \text{con } j=1, \dots, m \\
 \bar{\mathbf{z}}_\tau &= [\bar{z}(\tau) \bar{z}(t_1) \dots \bar{z}(t_n)]^T && \text{di dimensione } (n+1) \times 1
 \end{aligned} \tag{A.1}$$

I primi due vettori della (A.1) contengono rispettivamente i valori della curva media e della j -esima curva individuale negli istanti di campionamento t_1, \dots, t_n . Il terzo vettore è simile al primo ma contiene un termine in più nella prima riga che corrisponde al valore della componente media ad un generico istante di tempo τ .

Definiamo ora, come è stato fatto da Pillonetto et al. (2007), le seguenti matrici:

$$\bar{\mathbf{R}} = \text{Var}[\bar{\mathbf{z}}] := \begin{bmatrix} \bar{R}(t_1^1, t_1^1) & \dots & \bar{R}(t_1^1, t_{nN}^N) \\ \vdots & \ddots & \vdots \\ \bar{R}(t_{nN}^N, t_1^1) & \dots & \bar{R}(t_{nN}^N, t_{nN}^N) \end{bmatrix} \tag{A.2}$$

matrice di covarianza della curva media (dimensione $n \times n$)

$$\tilde{\mathbf{R}} = \text{Var}[\tilde{\mathbf{z}}^j] := \begin{bmatrix} \tilde{R}(t_1^j, t_1^j) & \dots & \tilde{R}(t_1^j, t_{n_j}^j) \\ \vdots & \ddots & \vdots \\ \tilde{R}(t_{n_j}^j, t_1^j) & \dots & \tilde{R}(t_{n_j}^j, t_{n_j}^j) \end{bmatrix} \tag{A.3}$$

matrice di covarianza degli shift individuali (dimensione $n \times n$)

$$\bar{\mathbf{R}}_\tau = \text{cov}[\bar{\mathbf{z}}_\tau, \bar{\mathbf{z}}] \tag{A.4}$$

matrice di covarianza tra il vettore $\bar{\mathbf{z}}_\tau$ e il vettore $\bar{\mathbf{z}}$ (dimensione $(n+1) \times n$)

$$\bar{\mathbf{R}}_{\tau\tau} = \text{Var}[\bar{\mathbf{z}}_{\tau}] \quad (\text{A.5})$$

matrice di covarianza della componente media dove sono state aggiunte nella prima riga e nella prima colonna le correlazioni tra gli istanti di campionamento e il generico istante di tempo τ (dimensione $(n+1) \times (n+1)$)

$$\hat{\mathbf{R}}_{\tau\tau} = \text{Var}[\bar{\mathbf{z}}_{\tau} | \mathbf{y}] \quad (\text{A.6})$$

matrice di covarianza del vettore $\bar{\mathbf{z}}_{\tau}$ condizionata dal vettore dei dati \mathbf{y} a disposizione (dimensione $(n+1) \times (n+1)$)

$$\bar{\mathbf{r}}_{\tau} = \text{cov}[\bar{z}(\tau), \bar{\mathbf{z}}] \quad (\text{A.7})$$

covarianza tra lo scalare $\bar{z}(\tau)$ (componente media all'istante generico τ) e il vettore $\bar{\mathbf{z}}$ (componente media negli istanti di campionamento) (dimensione $1 \times n$).

Si noti che la (A.3) è indipendente da j per l'assunzione di campionamento standardizzato e che le auto-covarianze dei singoli soggetti possono essere raggruppate in una matrice a blocchi con elementi nulli fuori dalla diagonale ($\text{blockdiag}\{\tilde{\mathbf{R}}^1, \tilde{\mathbf{R}}^2, \dots, \tilde{\mathbf{R}}^N\}$) perchè i processi $\tilde{z}^j(t)$ sono indipendenti tra di loro.

Sulla base di queste relazioni e delle conoscenze di stima bayesiana, si possono ottenere la stima gaussiana a minima varianza d'errore e la matrice dell'errore di stima:

$$E[\bar{z}(\tau) | \mathbf{y}] = \text{cov}[\bar{z}(\tau), \mathbf{y}] \text{var}[\mathbf{y}]^{-1} \mathbf{y} \quad (\text{A.8})$$

$$\hat{\mathbf{R}}_{\tau\tau} = \bar{\mathbf{R}}_{\tau\tau} - \text{cov}[\bar{z}_{\tau}, \mathbf{y}] \text{var}[\mathbf{y}]^{-1} \text{cov}[\bar{z}_{\tau}, \mathbf{y}]^T \quad (\text{A.9})$$

A.2 Stima della curva media ($E[\bar{z}(\tau) | \mathbf{y}]$ e $\text{Var}[\bar{z}(\tau) | \mathbf{y}]$)

Definiamo le seguenti matrici:

$$\tilde{\mathbf{R}}_{\nu}^i = \tilde{\mathbf{R}} + \Sigma_{\nu}^j \quad (n \times n) \quad (\text{A.10})$$

$$\mathbf{F} = \sum_{i=1}^m (\tilde{\mathbf{R}}_{\nu}^i)^{-1} \quad (n \times n) \quad (\text{A.11})$$

$$\mathbf{f} = \sum_{i=1}^m (\tilde{\mathbf{R}}_{\nu}^i)^{-1} \mathbf{y}^i \quad (n \times 1) \quad (\text{A.12})$$

sfruttando queste notazioni si ha che le stime finali sono:

$$\text{Var}[\bar{z}_\tau | \mathbf{y}] = \hat{\mathbf{R}}_{\tau\tau} = \bar{\mathbf{R}}_{\tau\tau} - \bar{\mathbf{R}}_\tau \mathbf{F} \bar{\mathbf{R}}_\tau^T + \bar{\mathbf{R}}_\tau \mathbf{F} (\bar{\mathbf{R}}^{-1} + \mathbf{F})^{-1} \mathbf{F} \bar{\mathbf{R}}_\tau^T \quad (\text{A.13})$$

$$E[\bar{z}(\tau) | \mathbf{y}] = \bar{\mathbf{r}}_\tau \mathbf{c} \quad (\text{A.14})$$

Dove $\mathbf{c} \in \mathbb{R}^n$ è dato da:

$$\mathbf{c} = [\mathbf{I}_n - \mathbf{F}(\bar{\mathbf{R}}^{-1} + \mathbf{F})^{-1}] \mathbf{f} \quad (\text{A.15})$$

La dimostrazione delle (A.13) e (A.14) sono trattate in dettaglio in Pillonetto et al., 2007.

A.3 Stima della curva individuale ($E[z^{m+1} | \mathbf{y}^*]$ e $\text{Var}[z^{m+1} | \mathbf{y}^*]$)

Assumiamo ora di avere a disposizione i dati di m sweeps e che in più si abbia l'arrivo di nuovi dati appartenenti ad un $(m+1)$ -esima registrazione (campionata agli stessi istanti di tempo). Si cerca ora di stimare la nuova curva individuale sfruttando l'informazione proveniente da tutti i dati a disposizione (approccio di popolazione).

Definiamo altre due relazioni:

$$\tilde{\mathbf{r}}_\tau^{m+1} = \text{cov}[\tilde{z}^{m+1}(\tau), \tilde{z}^{m+1}] \quad (\text{A.16})$$

$$\tilde{\mathbf{r}}_\tau = \text{Var}[\tilde{z}^{m+1}(\tau)] \quad (\text{A.17})$$

Per affrontare meglio il problema conviene partizionare $E[\bar{z}_\tau | \mathbf{y}]$ e $\hat{\mathbf{R}}_{\tau\tau}$ come segue:

$$\hat{\mathbf{R}}_{\tau\tau} = \begin{pmatrix} \text{Var}[\bar{z}(\tau) | \mathbf{y}] & \text{cov}[\bar{z}(\tau), \bar{\mathbf{z}} | \mathbf{y}] \\ \text{cov}[\bar{z}(\tau), \bar{\mathbf{z}} | \mathbf{y}]^T & \text{Var}[\bar{\mathbf{z}} | \mathbf{y}] \end{pmatrix} = \begin{pmatrix} \mathbf{p} & \bar{\mathbf{p}} \\ \bar{\mathbf{p}}^T & \mathbf{P} \end{pmatrix} \quad (\text{A.18})$$

$$E[\bar{z}_\tau | \mathbf{y}] = \begin{bmatrix} E[\bar{z}(\tau) | \mathbf{y}] \\ E[\bar{\mathbf{z}} | \mathbf{y}] \end{bmatrix} = \begin{bmatrix} \xi_\tau \\ \xi \end{bmatrix} \quad (\text{A.19})$$

dove

- $\text{Var}[\bar{z}(\tau) | \mathbf{y}]$ è lo scalare dell'errore di stima all'istante τ
- $\text{cov}[\bar{z}(\tau), \bar{\mathbf{z}} | \mathbf{y}]$ ha dimensione $1 \times n$
- $\text{Var}[\bar{\mathbf{z}} | \mathbf{y}]$ è la matrice di covarianza della curva media ($n \times n$)

Tenendo conto della (1.2) e del fatto che \bar{z} , \tilde{z}^{m+1} e \mathbf{v}^{m+1} sono indipendenti tra di loro e che \tilde{z}^{m+1} e \mathbf{v}^{m+1} sono indipendenti da \mathbf{y} , si ha che:

$$E[\mathbf{y}^{m+1} | \mathbf{y}] = E[\bar{z} + \tilde{z}^{m+1} + \mathbf{v}^{m+1} | \mathbf{y}] = E[\bar{z} | \mathbf{y}] \quad (\text{A.20})$$

$$\text{Var}[\mathbf{y}^{m+1} | \mathbf{y}] = \text{Var}[\bar{z} | \mathbf{y}] + \text{Var}[\tilde{z}^{m+1} | \mathbf{y}] + \text{Var}[\mathbf{v}^{m+1} | \mathbf{y}] = \text{Var}[\bar{z} | \mathbf{y}] + \tilde{\mathbf{R}} + \Sigma_v^{m+1} \quad (\text{A.21})$$

$$E[z^{m+1}(\tau) | \mathbf{y}] = E[\bar{z}(\tau) + \tilde{z}^{m+1}(\tau) | \mathbf{y}] = E[\bar{z}(\tau) | \mathbf{y}] \quad (\text{A.22})$$

$$\text{Var}[z^{m+1}(\tau) | \mathbf{y}] = \text{Var}[\bar{z}(\tau) + \tilde{z}^{m+1}(\tau) | \mathbf{y}] = \text{Var}[\bar{z}(\tau) | \mathbf{y}] + \text{Var}[\tilde{z}^{m+1}(\tau)] \quad (\text{A.23})$$

$$\text{cov}[z^{m+1}(\tau), \mathbf{y}^{m+1} | \mathbf{y}] = \text{cov}[\bar{z}(\tau), \bar{z} | \mathbf{y}] + \text{cov}[\tilde{z}^{m+1}(\tau), \tilde{z}^{m+1}]. \quad (\text{A.24})$$

Prendiamo ora le formule per la stima gaussiana e adattiamole al caso in esame:

$$E[z^{m+1} | \mathbf{y}^+] = E[z^{m+1} | \mathbf{y}] + \text{cov}[z^{m+1}(\tau), \mathbf{y}^{m+1} | \mathbf{y}] \cdot \text{var}[\mathbf{y}^{m+1} | \mathbf{y}]^{-1} \cdot (\mathbf{y}^{m+1} - E[\mathbf{y}^{m+1} | \mathbf{y}]) \quad (\text{A.25})$$

$$\text{Var}[z^{m+1}(\tau) | \mathbf{y}^+] = \text{Var}[z^{m+1}(\tau) | \mathbf{y}] - \text{cov}[z^{m+1}(\tau), \mathbf{y}^{m+1} | \mathbf{y}] \cdot \text{Var}[\mathbf{y}^{m+1} | \mathbf{y}]^{-1} \cdot \text{cov}[z^{m+1}(\tau), \mathbf{y}^{m+1} | \mathbf{y}]^T \quad (\text{A.26})$$

Sfruttando le relazioni precedenti si ha che:

$$E[z^{m+1} | \mathbf{y}^+] = E[\bar{z}(\tau) | \mathbf{y}] + (\text{cov}[\bar{z}(\tau), \bar{z} | \mathbf{y}] + \text{cov}[\tilde{z}^{m+1}(\tau), \tilde{z}^{m+1}]) \cdot (\text{Var}[\bar{z} | \mathbf{y}] + \tilde{\mathbf{R}} + \Sigma_v^{m+1})^{-1} \cdot (\mathbf{y}^{m+1} - E[\bar{z} | \mathbf{y}]) \quad (\text{A.27})$$

e

$$\text{Var}[z^{m+1}(\tau) | \mathbf{y}^+] = \text{Var}[\bar{z}(\tau) | \mathbf{y}] + \text{Var}[\tilde{z}^{m+1}(\tau)] - (\text{cov}[\bar{z}(\tau), \bar{z} | \mathbf{y}] + \text{cov}[\tilde{z}^{m+1}(\tau), \tilde{z}^{m+1}]) \cdot (\text{Var}[\bar{z} | \mathbf{y}] + \tilde{\mathbf{R}} + \Sigma_v^{m+1}) \cdot (\text{cov}[\bar{z}(\tau), \bar{z} | \mathbf{y}] + \text{cov}[\tilde{z}^{m+1}(\tau), \tilde{z}^{m+1}])^T \quad (\text{A.28})$$

Ora, applicando la partizione vista nelle (A.18) e (A.19), si possono ottenere le stime d'interesse per la curva individuale del soggetto (m+1)-esimo che sono state implementate nel codice:

$$E[z^{m+1} | \mathbf{y}^+] = \xi_\tau + [\bar{\mathbf{p}} + \tilde{\mathbf{r}}_\tau^{m+1}] \cdot (\mathbf{P} + \tilde{\mathbf{R}} + \Sigma_v^{m+1})^{-1} \cdot (\mathbf{y}^{m+1} - \xi) \quad (\text{A.29})$$

$$\text{Var}[z^{m+1}(\tau) | \mathbf{y}^+] = \rho + \tilde{\mathbf{r}}_\tau - [\bar{\mathbf{p}} + \tilde{\mathbf{r}}_\tau^{m+1}] \cdot (\mathbf{P} + \tilde{\mathbf{R}} + \Sigma_v^{m+1})^{-1} \cdot [\bar{\mathbf{p}} + \tilde{\mathbf{r}}_\tau^{m+1}]^T \quad (\text{A.30})$$

A.4 Calcolo della funzione di verosimiglianza

Il primo termine della (1.11) è dato da:

$$\mathbf{y}^T \text{Var}[\mathbf{y}]^{-1} \mathbf{y} = \mathbf{y}^T \mathbf{c} \quad (\text{A.31})$$

Per quanto riguarda il secondo termine della funzione costo $J(\mathbf{y};\theta)$, il determinante della $\text{Var}[\mathbf{y}]$ si può calcolare con il seguente algoritmo:

$$\det(\text{Var}[\mathbf{y}]) = \left(\prod_{i=1}^m \det(\mathbf{C}_{ii}) \right)^2 \quad (\text{A.32})$$

dove le \mathbf{C}_{ii} sono matrici $n \times n$ costruite in modo che:

$$\mathbf{C}_{ii} = \text{chol}[\mathbf{A}_{ii} - \mathbf{D}_i] \quad (\text{A.33-a})$$

$$\mathbf{C}_{(i+1)i} = (\bar{\mathbf{R}} - \mathbf{D}_i)(\mathbf{C}_{ii}^T)^{-1} \quad (\text{A.33-b})$$

$$\mathbf{C}_{ki} = \mathbf{C}_{(i+1)i} \quad \text{per } k > i+1 \quad (\text{A.33-c})$$

$$\mathbf{D}_{i+1} = \mathbf{D}_i + \mathbf{C}_{(i+1)i} \mathbf{C}_{(i+1)i}^T \quad (\text{A.33-d})$$

con $\text{chol}[\mathbf{K}]$ che denota la fattorizzazione inferiore di Cholesky della matrice \mathbf{K} e $\mathbf{A}_{kk} = \bar{\mathbf{R}} + \tilde{\mathbf{R}} + \Sigma_k^v$ che denota il k -esimo blocco $n \times n$ nella diagonale della matrice $\text{Var}[\mathbf{y}]$.

Introduciamo infine le seguenti condizioni iniziali per l'algoritmo:

$$\mathbf{D}_1 = \mathbf{0}$$

$$\mathbf{C}_{21} = \bar{\mathbf{R}} (\mathbf{C}_{11}^T)^{-1} \quad (\text{A.34})$$

Le dimostrazioni della (A.32) e delle (A.33) sono riportate nell'articolo di Pillonetto et al.2007.

Per quanto visto precedentemente, la funzione da minimizzare per rendere massima la verosimiglianza risulta essere:

$$J(\mathbf{y};\boldsymbol{\theta}) = \mathbf{y}^T \mathbf{c} + \log(\prod_{i=1}^m \det(\mathbf{C}_{ii}))^2 \quad (\text{A.35})$$

Si ricercano quindi i valori del vettore $\boldsymbol{\theta} = [\bar{\lambda}^2 \tilde{\lambda}^2]^T$ che rendono minima la (A.35), per poi sostituirli nelle (1.8) e (1.9) in modo da ottenere le funzioni di auto-covarianza della curva media e delle curve individuali.

Appendice B

Vengono qui riportati il programma main e le functions scritte in linguaggio di programmazione Matlab che realizzano l'algoritmo di multi-task learning applicato a potenziali evocati acquisiti durante un compito di Simon.

Prima di riportare il codice descriviamo passo dopo passo come esso è stato ricavato.

Lo script *main.m* carica i dati di un soggetto e, dopo averli sottocampionati da 512 a 128 Hz, li raccoglie in una matrice (in ogni riga ci saranno i campioni dell'i-esima sweep). A questi segnali viene poi tolta la media del pre-stimolo in modo da avere che il valore atteso sia nullo.

A questo punto si usa un ciclo for per creare il vettore \mathbf{y} dei dati: per ogni sweep si toglie il pre-stimolo (800 ms) e si prende un campione ogni 4. In questo modo si riduce di molto il costo computazionale che passa da decine di minuti a pochi minuti.

All'interno dello stesso ciclo si calcola la matrice \mathbf{A} che contiene la varianza e i coefficienti del modello auto regressivo del rumore calcolati dal pre-stimolo di ogni sweep: per trovare l'ordine p ottimo del modello si è fatto riferimento all'*fpevalue* (valore assunto dal *final prediction error*), mentre per il calcolo dei coefficienti sono state usate le equazioni di *Yule Walker*.

L'ultima riga e l'ultima colonna di Σ_v sono state forzate a valori molto bassi in modo da considerare l'ultimo campione deterministicamente noto (non sono state considerate nulle per problemi di singolarità).

Si passa ora all'applicazione dell'algoritmo di multi-task learning.

Il primo passo è il calcolo degli iper-parametri $\bar{\lambda}^2$ e $\tilde{\lambda}^2$: si sfrutta una function già presente in Matlab (*fminsearch*) per calcolare il minimo della funzione di verosimiglianza data dalla (1.11). Come dati iniziali sono stati messi due valori presumibilmente vicini a quelli ottimi in modo da velocizzare la convergenza e, per il calcolo della funzione costo da minimizzare, è stata implementata la function *mxlik.m*.

Questa function calcola inizialmente il primo termine della funzione di verosimiglianza, dato dalla (A.31), utilizzando la function *stimac.m* che stima il vettore \mathbf{c} dei pesi dell'i-esima sweep.

Il secondo termine della (1.11) è dato dall'algoritmo descritto nell'appendice A dove viene sfruttata la fattorizzazione di Cholesky per calcolare il determinante della $\text{Var}[\mathbf{y}]$. Vengono inoltre sfruttate le function *Rmean.m* e *Rtilde.m* per calcolare rispettivamente le funzioni di auto covarianza della curva media e delle curve individuali.

Per questo step di minimizzazione si è scelto di usare come parametri $\exp(\bar{\lambda}^2)$ invece di $\bar{\lambda}^2$ per avere il vincolo di non negatività e per fare in modo che l'algoritmo di minimizzazione arrivi prima a convergenza.

Dopo questo passo, c'è un controllo dei valori ottenuti dalla funzione *fminsearch*: se i valori degli iper-parametri sono troppo bassi (come è successo in alcuni casi), ad essi vengono assegnati dei valori empirici (presi da casi in cui venivano stimati bene).

Per la stima della curva media, vengono calcolati \mathbf{F} ed \mathbf{f} tramite le (A.11) e (A.12) per poi essere utilizzati nella stima del vettore \mathbf{c} (A.15). Ora, con un ciclo for, si può calcolare $\bar{\mathbf{r}}_\tau$ per ogni istante di tempo tramite la function *rmeantau.m* e stimare i valori, istante per istante, della curva media utilizzando la (A.14).

Successivamente, si applica la partizione di $E[\bar{\mathbf{z}}_\tau|\mathbf{y}]$ ed $\hat{\mathbf{R}}_\tau$ come visto nelle (A.18) e (A.19) in modo da ottenere $\mathbf{p}, \bar{\mathbf{p}}, \mathbf{P}, \xi_\tau$ e ξ .

Per stimare le curve individuali, si applica un doppio ciclo for dove, per ogni sweep, viene calcolato il valore, istante per istante, della stima della curva individuale utilizzando la (1.46): si noti che al posto di $\tilde{\mathbf{r}}_\tau$ viene usata ancora la function *rmeanTau.m* perché si è deciso di adottare per le curve individuali lo stesso modello a priori della curva media.

CODICE:

Lo script *main.m* contiene l'algoritmo descritto nel capitolo 1: il programma carica i dati in una matrice, calcola il modello del rumore dal prestimolo di ogni sweep ed applica la function *mxlik* per trovare gli iperparametri che verranno usati per stimare la curva media e le curve individuali.

```
%programma main che carica N sweep (già filtrate) di un paziente giovane
(compito di Simon)
```

```
clear all
close all
clc
```

```
load soggetto_condition1.mat
```

```
N=25;           %numero di sweep
el=26;         %numero dell'elettrodo Pz
```

```
Fs=512; %frequenza originale di campionamento
Fc=128; %nuova frequenza di campionamento
```

```
segnali=double(EEG.data(el,1:Fs/Fc:926,:));
segnali=permute(segnali,[3 2 1]);
tempi=EEG.times(1:Fs/Fc:926);
```

```
clear el EEG
```



```

%tolgo la media del pre-stimolo
ps=mean(segnali(:,1:103),2);
segnali=segnali-repmat(ps,1,size(segnali,2));

step=4;

ordmax=10; %ordine max del modello AR sul pre-stimolo
y=[];
myAR=zeros(N,ordmax+2);

for i=1:N
    temp=segnali(i,:);
    pre=temp(1:step:103); %pre-stimolo
    temp2=temp(103:step:232);
    temp2(1)=0;
    temp2(end)=0;
    y=[y;temp2']; %vettore dei dati sottocampionati

    fpevalue=zeros(1,ordmax);
    for j=1:ordmax
        th=ar(pre,j,'yw');
        fpevalue(j)=fpe(th);
    end
    [Y p]=min(fpevalue); %ordine ottimo del modello
    th=ar(pre,p,'yw');
    myAR(i,1)=th.noisevariance; %varianza del modello AR del rumore
    myAR(i,2:p+2)=th.a; %coefficienti del modello AR del rumore
end

ts=tempi(103:step:232)'; %griglia temporale sottocampionata
n=length(ts); %numero di campioni per sweep

%matrici di covarianza dell'EEG
sigmaV=[];
for i=1:N
    col=zeros(n,1);
    for j=1:ordmax+1
        col(j)=myAR(i,j+1);
    end
    rig=zeros(1,n);
    rig(1)=1;
    A=toeplitz(col,rig);
    sigmav=myAR(i,1)*inv(A'*A);
    sigmav(n,:)=0.01;
    sigmav(:,n)=0.01;
    sigmaV=[sigmaV;sigmav];
end

%averaging convenzionale
sum=0;
for i=1:N
    temp=segnali(i,:);
    sum=sum+temp(103:232);
end
aver=sum/N;

clear temp temp2 pre th Y p ordmax m sum var

%-----algoritmo di multi-task learning non parametrico-----

```

```

%calcolo degli iper-parametri

OPTIONS=optimset('TolFun',1e-
3,'MaxIter',1e2,'MaxFunEvals',1e3,'TolX',1e-3,'FunValCheck','on');
[X Fval] = fminsearch(@(X) mxlik(X,ts,y,N,n,sigmaV),[-5 -5],OPTIONS);
toc

lambda=exp(X(1))
lambdaTilde=exp(X(2))

%forzatura parametri

if lambda<1e-6
    lambda=1e-3
end

if lambdaTilde<1e-6
    lambdaTilde=1e-4
end

%----- stima della curva media -----

t=tempi(103:232)';

%calcolo F e f
F=zeros(n,n); %inizializzazione di F
f=zeros(n,1); %inizializzazione di f
for i=1:N
    yi=y(i*n-n+1:n*i); %vettore dei dati del soggetto i-esimo
    sigmav=sigmaV(i*n-n+1:n*i,:); %varianza del soggetto i-esimo
    Rtildavi=Rtilde(ts,lambdaTilde)+sigmav;
    F=F+inv(Rtildavi);
    f=f+inv(Rtildavi)*yi;
end

%stima dei coefficienti c
I=eye(n);
c=(I-F*inv(inv(Rmean(ts,lambda))+F))*f;

%stima della curva media
zmean=zeros(length(t),1);

for i=1:length(t)
    rmeantau=rmeanTau(ts,t(i),lambda);
    zmean(i)=rmeantau*c; %E[zmean(tau)|y]
end

%intervalli di confidenza
p=zeros(length(t),1);
phat=[];
P=[];
for i=1:length(t)
    Rmeantautau=RmeanTautau(ts,t(i),lambda);
    Rmeantau=RavgTau(ts,t(i),lambda);
    RhatTau=Rmeantautau-Rmeantau*F*Rmeantau'
+Rmeantau*F*inv(inv(Rmean(ts,lambda))+F)*F*Rmeantau';
    p(i,1)=RhatTau(1,1); %var[zmean(tau)|y]
    phat=[phat;RhatTau(1,2:n+1)]; %cov[(zmean(tau),zmean)|y]
end

```

```

        P=[P;RhatTau(2:n+1,2:n+1)];      %var[zmean|y]
end

%----- stima della curva individuale -----

%calcolo di xi e di xiTau
xiTau=zmean;
temp=Rmeantau*c;
xi=temp(2:n+1);

%calcolo delle curve individuali (uso rmeanTau perchè il modello è il
%medesimo)
temp=zeros(length(t),1);
individual=zeros(N,n);
for i=1:N
    for j=1:length(t)

individual(i,j)=xiTau(j)+(phat(j,:)+rmeanTau(ts,t(j),lambdaTilde))
*inv(P((j*n)-n+1:j*n,:)+Rtilde(ts,lambdaTilde)
+sigmaV(i*n-n+1:n*i,:))*(y(i*n-n+1:n*i)-xi);
        end
    end
end

```

La function *mxlik.m* calcola la funzione di verosimiglianza (si veda il paragrafo A.4): riceve in ingresso il vettore dei **parametri**, la griglia di campionamento **ts**, il vettore dei dati **y**, il numero **N** di sweeps, il numero **n** di campioni per sweep e la matrice del rumore **sigmaV**.

```

%function che calcola la funzione costo da minimizzare (maximum
%Likelihood)

function J=mxlik(parametri,ts,y,N,n,sigmaV)

lambda=exp(parametri(1));
lambdaTilde=exp(parametri(2));

%primo termine
J1=0;
for k=1:N
    yk=y(k*n-n+1:n*k);
    c=stimac(ts,y,lambda,lambdaTilde,N,k,sigmaV);
    J1=J1+yk'*c;
end

%secondo termine
J2=0;
R=Rmean(ts,lambda);
D=zeros(n,n);
for i=1:N
    sigmav=sigmaV(i*n-n+1:n*i,:); %varianza del soggetto i-esimo
    A=R+Rtilde(ts,lambdaTilde)+sigmav;
    [C1 p]=chol(A-D);
    C1=C1'; %per avere C1 triangolare inferiore
    if p==0 %se A-D è definita positiva

```

```

        C2=(R-D)*inv(C1');
        D=D+C2*C2';
        J2=J2+log(det(C1));
    else
        J2=inf;
    end
end
J2=2*J2;

%funzione costo totale
J=J1+J2

```

La function *stimac.m* implementa la (A.15) per ottenere la stima del vettore dei pesi \mathbf{c} .

In ingresso abbiamo il vettore della griglia di campionamento \mathbf{ts} , il vettore dei dati \mathbf{y} , gli iperparametri λ e λ Tilde, il numero N di sweeps, il numero K della sweep di cui si vuole stimare il peso e la matrice del rumore $\sigma\mathbf{v}$.

Questa funzione restituisce il vettore che rappresenta i pesi della k -esima sweep.

```

%funzione che restituisce i pesi c della k-esima sweep

function c=stimac(ts,y,lambda,lambdaTilde,N,k,sigmaV)

n=length(ts); %numero di campioni per soggetto
F=zeros(n,n); %inizializzazione di F
f=zeros(n,1); %inizializzazione di f
for i=1:N
    yi=y(i*n-n+1:n*i); %vettore dei dati del soggetto i-esimo
    sigmav=sigmaV(i*n-n+1:n*i,:); %varianza del soggetto i-esimo
    Rtildevi=Rtilde(ts,lambdaTilde)+sigmav;
    f=f+inv(Rtildevi)*yi;
    F=F+inv(Rtildevi);
    if i==k
        Ftemp=inv(Rtildevi);
        ftemp=inv(Rtildevi)*yi;
    end
end

c=ftemp-Ftemp*(inv(inv(Rmean(ts,lambda))+F))*f;

```

Vengono infine riportate le functions che implementano rispettivamente le matrici di auto-covarianza della curva media e degli shift individuali, $\bar{\mathbf{R}}_v$, $\bar{\mathbf{R}}_{\tau}$ ed $\bar{\mathbf{r}}_{\tau}$.

```

%funzione che restituisce la matrice di auto-covarianza della curva
media
%ingressi: tempi di campionamento e iperparametro

```

```

function R = Rmean(ts,lambda)

for i=1:length(ts)
    ti=ts(i);
    for j=1:length(ts)
        tau=ts(j);
        if ti<=tau
            R(i,j)=lambda*0.5*(ti^2)*(tau-(ti/3));
        else
            R(i,j)=lambda*0.5*(tau^2)*(ti-(tau/3));
        end
    end
end
end

```

```

%funzione che restituisce la matrice di auto-covarianza della curva
individuale
%ingressi: tempi di campionamento e iperparametro

```

```

function Rind = Rtilde(ts,lambdaTilde)

for i=1:length(ts)
    ti=ts(i);
    for j=1:length(ts)
        tau=ts(j);
        if ti<=tau
            Rind(i,j)=lambdaTilde*0.5*(ti^2)*(tau-(ti/3));
        else
            Rind(i,j)=lambdaTilde*0.5*(tau^2)*(ti-(tau/3));
        end
    end
end
end

```

```

%funzione che restituisce cov[zmeanTau,zmean]

```

```

function Rmeantau=RavgTau(ts,tau,lambda)

t2=[tau; ts];
for i=1:length(ts)
    ti=ts(i);
    for j=1:length(t2)
        tau=t2(j);
        if ti<=tau
            Rmeantau(j,i)=lambda*0.5*(ti^2)*(tau-(ti/3));
        else
            Rmeantau(j,i)=lambda*0.5*(tau^2)*(ti-(tau/3));
        end
    end
end
end
end

```

```

%funzione che restituisce la var[ZmeanTau]

function Rmeantautau = RmeanTautau(ts,tau,lambda)

ty=[tau;ts];
for i=1:length(ty)
    ti=ty(i);
    for j=1:length(ty)
        tau=ty(j);
        if ti<=tau
            Rmeantautau(i,j)=lambda*0.5*(ti^2)*(tau-(ti/3));
        else
            Rmeantautau(i,j)=lambda*0.5*(tau^2)*(ti-(tau/3));
        end
    end
end
end

```

```

%funzione che restituisce cov[zmean(tau),zmean]

function rmeantau = rmeanTau(ts,tau,lambda)

for i=1:length(ts)
    ti=ts(i);
    if ti<=tau
        rmeantau(1,i)=lambda*0.5*(ti^2)*(tau-(ti/3));
    else
        rmeantau(1,i)=lambda*0.5*(tau^2)*(ti-(tau/3));
    end
end
end
end

```

Bibliografia

[1] Suman F.

Metodi di multi-task learning per la stima di segnali fisiologici: algoritmi ed applicazioni per la misura di potenziali evocati single-trial

Tesi di Laurea,DEI, Università di Padova,A.A. 2008/2009

[2] Pasqualotto E.

Misura single trial di potenziali evocati: nuove metodiche e applicazione allo studio di processi cognitivi

Tesi di Laurea,DEI, Università di Padova,A.A. 2008/2009

[3] Neve M.

Bayesian learning techniques for non parametric identification

Tesi di Dottorato, Dept. Of Computer Engineering and System Science, University of Pavia 2005

[4] Pillonetto G.,De Nicolao G.,Chierici M.,Cobelli C.

Fast Algorithms for Nonparametric Population Modeling of Large Data Sets

Automatica 45(1): 173-179 (2007)

[5] Pillonetto G.,Dinuzzo F.,De Nicolao G.

Bayesian online multi-task learning

IEEE Transactions on Pattern Analysis and Machine Intelligence, 32(2):193-205 (2010)

[6] Caruana R.

Multitask learning

In: "Pratt L.,Thrun S. (Editors), Machine Learning, 28:41-75(1997)"

[7] Liu Q., Liao X., Li H., Stack J. and Carin L.

Semisupervised Multitask Learning

IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(6) (2009)

[8] Verrotta D., Park k., T.F.Blaschke, L.B.Sheiner

A Semiparametric Method for Describing Noisy Population Pharmacokinetic Data

Journal of Pharmacokinetics and Biopharmaceutics, 25(5) (1997)

[9] Bansal P., Sun M. and Sciabassi R.

Simulation and Extraction of Single-trial Evoked Potentials

Proceedings of the 26th Annual International Conference of the IEEE EMBS, 1:200-3 (2004)

[10] Sparacino G., Milani S., Arslan E., Cobelli C.

A Bayesian approach to estimate evoked potentials

Computer Methods and Programs in Biomed., 68: 223-248, 2002

[11] Leonowicz Z., Karvanen J., Shishkin S.L.

Trimmed estimators for robust averaging of event-related potentials

Journal of Neuroscience Methods, 142: 17–26 (2005)

[12] Rahnea T., Spechtb H., Muhler R.

Sorted averaging—application to auditory event-related responses

Journal of Neuroscience Methods, 172: 74–78 (2008)

[13] Zouridakis G., Iyer D., Diaz J. and Patidar U.

Estimation of individual evoked potential components using iterative independent component analysis

Phys. Med. Biol. 52: 5353–5368 (2007)

[14] Mouraux A., Iannetti G.D.

Across-trial averaging of event-related EEG responses and beyond

Magnetic Resonance Imaging, 26: 1041–1054 (2008)

[15] Merzagora A.C., Bracchi F., Cerutti S., Rossi L., Gaggiani A., and Bianchi A.M.

Evaluation and Application of a RBF Neural Network for Online Single-Sweep Extraction of SEPs During Scoliosis Surgery

IEEE Transactions on Biomedical Engineering, 54(7) (2007)

[16] Li R., Keil A., Principe J.C.

Single-trial P300 estimation with a spatiotemporal filtering method

Journal of Neuroscience Methods, 177: 488–496 (2009)

[17] Karjalainen P.A.

Regularization and Bayesian Methods for Evoked Potential Estimation

PhD Thesis, Department of Applied Physics, University of Kuopio (1997)

[18] Stefanos D. Georgiadis, Perttu O. Ranta-aho, Mika P. Tarvainen, and Pasi A. Karjalainen

A Subspace Method for Dynamical Estimation of Evoked Potentials

Comput Intell Neurosci.; vol. 2007: Article ID 61916 (2007)

[19] Guterman H., Nehmadi Y., Chistyakov A., Soustiel J., Feinsod M.

A comparison of neural network and Bayes recognition approaches in the evaluation of the brainstem trigeminal evoked potentials in multiple sclerosis

International Journal of Bio-Medical Computing, 43: 203-213 (1996)

[20] Nait-Ali, A.M. Adam, O. Motsch, J.-F.

The brainstem auditory evoked potentials estimation using a Bayesian deconvolution method

Engineering in Medicine and Biology Society, 4: 1516 - 1517 (1996)

[21] Friston K., Henson R., Phillips C., Mattout J.

Bayesian estimation of evoked and induced responses.

Hum Brain Mapp.; 27(9):722-35 (2006)

Ringraziamenti

Prima di tutto ringrazio il prof. Giovanni Sparacino e gli ing. Costanza D'Avanzo e Anahita Goljahani per avermi aiutato nello svolgimento di questa tesi dimostrando disponibilità e interesse nel lavoro svolto.

Poi ringrazio i miei genitori per avermi sempre supportato in ogni mia decisione, lasciandomi libero di fare ciò che più volevo.

Grazie ai miei amici e ai miei compagni di università, senza i quali questi cinque anni sarebbero stati molto più difficili.

A Giulia, ultima ma più importante, per essermi sempre stata vicina. Che questo anno e mezzo sia solo l'inizio di una bellissima vita insieme.