

**POLITECNICO DI MILANO**  
**Facoltà di Ingegneria dei Sistemi**  
**Corso di Laurea in Ingegneria Biomedica**  
**Laurea Magistrale**



**UN ALGORITMO GENETICO PER LA  
VALUTAZIONE DI DIFFERENTI PROTOCOLLI  
DI STIMOLAZIONE VISIVA IN P300-BCI**

**Relatori:**

Prof. Luca Mainardi

Prof. Matteo Matteucci

**Tesi di Laurea di:**

Francesco Onorati 724696

Anno accademico 2009/2010

# Indice

---

Indice.....	1
Sommario.....	3
Abstract.....	7
Introduzione.....	10
1. Brain-Computer Interface.....	12
1.1 Il Sistema Nervoso Centrale.....	12
1.2 Segnale EEG.....	14
1.3 Potenziali evento-relati: P300.....	16
1.4 Il <i>rebound</i> .....	20
1.5 Meccanismi attentivi.....	22
1.6 Classificazione delle BCI.....	24
1.6 Le componenti di una BCI.....	28
1.7 BCI non invasive: alcuni approcci applicativi.....	30
1.7.1 MEG.....	31
1.7.2 fMRI.....	31
1.7.3 NIRS.....	31
1.7.4 Potenziali corticali lenti.....	32
1.7.5 Potenziali motori.....	32
1.8 P300-BCI.....	33
2. Metodi.....	36
2.1 Classificazione.....	36
2.1.1 Classificatore logistico.....	38
2.2 Computazione evolutiva.....	40
2.2.1 Algoritmo genetico.....	41
2.2.2. La funzione di <i>fitness</i> .....	48
3. Protocollo sperimentale.....	54
3.1 Interfacce di stimolazione visiva.....	55
3.1.1 Ipotesi sperimentale.....	58
3.2 Costruzione del <i>dataset</i> .....	59
3.3 Protocollo di analisi.....	61
3.3.1 Ampiezza della finestra.....	61

3.3.2 Struttura dell'algoritmo genetico.....	62
4. Risultati.....	70
4.1 Interfaccia "Arrows" .....	72
4.1.1 GA vs ICA .....	72
4.1.2 GA senza <i>rebound</i> vs GA con <i>rebound</i> .....	81
4.1.3 Confronto <i>dataset</i> .....	88
4.1.4 <i>Template</i> .....	89
4.2 Interfaccia "Auto" .....	93
4.2.1 GA vs ICA .....	93
4.2.2 GA senza <i>rebound</i> vs GA con <i>rebound</i> .....	100
4.2.3 Confronto <i>dataset</i> .....	107
4.2.4 <i>Template</i> .....	108
4.3 Interfaccia "Vol" .....	111
4.3.1 GA vs ICA .....	111
4.3.2 GA senza <i>rebound</i> vs GA con <i>rebound</i> .....	115
4.3.3 Confronto <i>dataset</i> .....	119
4.3.4 <i>Template</i> .....	120
4.4 Confronto tra le diverse interfacce.....	122
5. Conclusioni e sviluppi futuri.....	125
Bibliografia.....	129

# Sommario

---

Un'interfaccia cervello-computer (in inglese, *Brain-Computer Interface*, abbreviato in BCI) è un'interfaccia che connette il computer direttamente al cervello, attraverso i segnali generati durante la sua attività. Per far ciò è necessaria un'analisi numerica che interpreti l'attività cerebrale, letta attraverso tale interfaccia.

La recente disponibilità di hardware digitali sempre più potenti e lo sviluppo di nuovi algoritmi di analisi, ha portato l'EEG a divenire un potenziale mezzo di comunicazione diretto tra cervello e computer.

Sono diversi i tipi di attività cerebrale usati in BCI, secondo il pattern temporale o in frequenza che un evento interno o esterno fa emergere. In questo lavoro di tesi, ci siamo focalizzati su un particolare potenziale, la P300, includendo in alcune analisi la successiva onda lenta, chiamata in questo lavoro *rebound* e riconducibile alla componente *slow negative wave 2* (SNW2).

L'esistenza di un sistema d'attenzione implicito è particolarmente rilevante per lo sviluppo delle BCI, poiché l'attenzione del soggetto è in grado di influenzare le caratteristiche della risposta allo stimolo: ciò consente di adottare un approccio cognitivo ai sistemi BCI, per sviluppare interfacce diverse e verificare se e quali di queste risultino maggiormente efficaci, con lo scopo finale di essere implementate in neuroprotesi.

Lo scopo del lavoro qui presentato è verificare che il sistema da noi sviluppato abbia *performance* confrontabili con un sistema recentemente illustrato in letteratura, basato sulla decomposizione ICA; confrontare le prestazioni su diverse interfacce, verificando se le differenti prestazioni ottenute siano significative; stabilire se l'uso di registrazioni di durata maggiore, che possano includere la SNW2, apportino effettivi e significativi vantaggi in termini di prestazioni.

## Materiali e Metodi

Il nostro sistema di BCI prevede, nella fase di selezione ed estrazione di caratteristiche dal segnale, la mappatura dei segnali EEG in uno spazio delle *feature*, in cui le singole *sweep*, *target* o non *target*, tendono a separarsi: la selezione delle *feature* è affidata a un algoritmo genetico progettato *ad hoc* per l'applicazione.

Nella seconda fase si procede alla classificazione di ogni *sweep* attraverso l'addestramento, per mezzo delle *feature* stesse, di un classificatore logistico.

Infine si procede a quantificare le prestazioni del classificatore rispetto a una funzione di merito, scelta tra le numerose presenti in letteratura, in modo da garantire una misura efficace della classificazione, confrontabile con altri sistemi.

Dodici soggetti sani hanno preso parte allo studio; ogni registrazione inizia 500 ms prima dello stimolo e termina 1000 ms dopo lo stimolo, per una durata totale di 1500 ms; la finestra temporale di riferimento, detta "senza *rebound*", comprende 200 ms precedenti allo stimolo, fino agli 800 ms successivi, per un intervallo totale di 1 s; per includere il *rebound*, abbiamo aumentato la finestra temporale, includendo anche gli ultimi 200 ms di registrazione, in modo da confrontare analisi diverse del nostro sistema.

Ai partecipanti sono state presentate tre interfacce, denominate "Arrows", "Auto" e "Vol"; per ogni interfaccia i partecipanti hanno eseguito otto sessioni di *learning* (LS), sedici sessioni di *testing* (TS), divise in quattro sedute (le prime tre vanno a accrescere il *training set* del classificatore, l'ultima seduta, detta *last testing* (LTS), costituisce uno dei due *testing set* del classificatore), quattro sessioni di *follow-up* (FS).

Sono state eseguite due analisi distinte, una in cui il *testing set* del classificatore è la sessione LTS, una in cui il *testing set* del classificatore è costituito dalle sessioni di FS: ciò in modo da testare la robustezza e la capacità di generalizzazione del classificatore.

Per ogni analisi, sono stati costruiti due tipi di *dataset*: uno fa riferimento alle *sweep* utilizzate dal sistema dell'équipe del S.Camillo per l'analisi e la creazione del proprio classificatore, scelte tra le sessioni LS+TS; uno fa riferimento a tutte le *sweep* delle fasi di *training* (LS+TS).

Pertanto, tutte le analisi precedentemente descritte costituiscono un totale di 20 analisi ( $4 \text{ dataset} \times 2 \text{ finestre temporali}$  per le interfacce "Arrows" e "Auto";  $2 \text{ dataset} \times 2 \text{ finestre temporali}$  per l'interfaccia "Vol").

## Risultati e conclusioni

I risultati ottenuti dal nostro sistema risultano superiori alle prestazioni ottenute dal sistema ICA: per tutti i *dataset* di ogni interfaccia il nostro sistema risulta superiore, sia nel valor medio della misura F-bilanciata, sia nel confronto inter-soggettivo, mediamente per 10 soggetti su 12; la robustezza del metodo è assicurata da una bassa varianza (dalle 4 alle 5 volte inferiore rispetto al metodo ICA), da risultati medi e inter-soggettivi confrontabili al variare di *dataset* e di interfacce, e da differenze statisticamente non significative al variare del *set* di *testing*, che indica anche una buona generalizzazione dell'algoritmo, condizione prevista in fase di progettazione, includendo una cross-validazione a 4 parti.

Includere il *rebound* sembra dare buoni risultati, permettendo un miglioramento fino al 10% rispetto alle prestazioni iniziali: per tutti i *dataset* di ogni interfaccia la classificazione con il *rebound* risulta essere migliore della classificazione senza *rebound*; per le interfacce “Arrows” e “Auto” le differenze sono statisticamente significative, al contrario di quanto accade per l'interfaccia “Vol”, probabilmente a causa del fatto che la classificazione per questa interfaccia raggiunge già soluzioni ottime senza *rebound*, difficilmente migliorabili da un contributo informativo comunque limitato.

Per quanto concerne le differenze tra le interfacce, le interfacce “Arrows” ed “Auto” hanno *performance* quasi sovrapponibili; il miglioramento delle prestazioni per l'interfaccia “Vol” conferma invece le ipotesi sperimentali: per questa interfaccia il nostro sistema registra le migliori *performance* in valor medio e in valor assoluto; per quanto riguarda il sistema ICA, non ci sono differenze tra le interfacce: una possibile spiegazione è che il nostro sistema riesce a cogliere con maggiore efficacia le diverse risposte al nuovo protocollo di stimolazione visiva proposto dall'équipe del S. Camillo.

## Sviluppi futuri

Gli sviluppi futuri in questo contesto sono molteplici e trasversali, abbracciando le competenze e gli ambiti più diversi.

Alcuni sviluppi che potrebbero essere apportati al sistema da noi esaminato sono: l'uso di una finestra temporale più ampia, che completerebbe lo studio relativo al *rebound*; l'uso di canali EEG in cui le differenze *target* e non *target* risultino più accentuate; l'introduzione di un'analisi tempo-frequenza, per integrare ai *pattern*

temporali, *pattern* in frequenza: a tal proposito, andrebbe modificato l'algoritmo genetico in modo da sfruttare i nuovi contributi informativi.

Infine, uno sviluppo immediato potrebbe essere implementare una versione in tempo reale del sistema: ciò porterebbe alla realizzazione di un *tool* applicativo in ambito clinico, che potrebbe coadiuvare lo sviluppo di nuovi paradigmi riabilitati e potenzialità comunicative.

# Abstract

---

A Brain-Computer Interface connects directly a computer to the brain, through signals generated during brain activity. For this purpose, numerical analysis is necessary to interpret brain activity recorded through the interface.

The recent availability of increasingly powerful digital hardware and the development of new analysis algorithms has led the EEG to become a potential means of direct communication between computer and brain.

There are different brain activities used in BCI, according to the temporal or frequency pattern elicited by an internal or external event. In this thesis, we focused on particular brain potentials: the P300 and one of the subsequent slow negative waves, the “rebound” wave or SNW2.

The existence of an implicit attentive system is important for the development of BCI, because the subject's attention can influence the quality of his/her response to a stimulus: a cognitive approach to BCI systems can develop different interfaces and evaluate which of them is more effective for implementing in neuroprostheses.

The aim of this work is: to verify whether our system has performances comparable with an ICA decomposition-based system, recently illustrated in the literature; to compare performances on different interfaces, checking whether the obtained differences are statistically significant; and to determine whether longer recordings including SNW2 can bring real and significant benefits in terms of performance.

## Materials and Methods

As first step of feature selection and extraction from the signal, our BCI system maps EEG signals in a feature space, where sweeps tend to separate from each other; feature selection is provided by a genetic algorithm specifically designed for the application.



As second step, a label is assigned to each element in the feature space, and the sweeps' classification is achieved by a logistic classifier trained with the features obtained from the GA.

Finally, we proceed to quantify the performance of the classifier using a fitness function chosen between the ones illustrated in the literature to ensure an effective and comparable measure.

Twelve healthy subjects participated in the study. Every recording starts 500 ms before and ends 1000 ms after the stimulus onset, for a duration of 1500 ms; the reference 1-s window starts 200 ms before and ends 800 ms after the stimulus onset; in order to include the SNW2, we enlarged the time window up to the end of the recording, so that we could compare performances from different analyses of our system.

Three interfaces, called "Arrows", "Auto" and "Vol", were shown to the subjects; for each interface, participants performed eight learning sessions (LS), sixteen testing sessions, divided into four sessions (the first three are used to increase the training set of the classifier, the last testing session (LTS) is one of the two testing sets of the classifier), and four follow-up sessions (FS).

Two separate analyses were made, using alternatively LTS or FS as testing set for the classifier, in order to test its robustness and generalization ability; for each analysis, there are two types of data sets: the first one is a selection of sweeps provided by S. Camillo Hospital's researchers for the analysis and creation of their classifier; the second one is made of all training set sweeps.

Twenty analyses have been performed: 4 data sets  $\times$  2 time windows, for both the "Arrows" and the "Auto" interfaces; 2 data sets  $\times$  2 time windows for the "Vol" interface.

## **Results and conclusions**

The results obtained with our system are better than those obtained with the ICA system: our system has higher average performances in every data set of every interface, proving more effective on an average of 10 of 12 subjects.

The reliability of the method is ensured by: a low variance (4-5 times lower than with the ICA method); comparable average and inter-subjective performances on different data sets and interfaces; and statistically non-significant differences on

different testing sets. Such a result also indicates a good generalization of the algorithm, as expected during its designing, when a 4-folds cross-validation was included.

The integration of the SNW2 seems to give good results (an improvement of up to 10% compared to the initial performance): for every data set of every interface the classification including SNW2 is better than the one without it; for both the “Arrows” and the “Auto” interfaces the differences are statistically significant; this doesn’t happen with the “Vol” interface, whose classification probably already reaches optimal solutions without SNW2; therefore such a result is difficult to improve by adding a limited information rate.

Regarding the differences between the interfaces, “Arrows” and “Auto” show almost overlapping performances; performances improvement for the “Vol” interface confirms the experimental hypothesis: for this interface our system offers the best performance, in both average and absolute values. There is no improvement for the ICA system: a possible explanation is that our system is able to catch the different responses to the new visual stimulation protocol by the S. Camillo team more effectively.

### **Future developments**

Future developments in this context are manifold and transversal, as they affect different skills and backgrounds.

Some improvements that could be implemented on the system we examined are: the use of a wider time window, which would complete the study on the SNW2; the use of EEG channels in which differences between target and non-target are more pronounced; and the introduction of a time-frequency analysis, to integrate frequency patterns: for this purpose, the genetic algorithm should be modified.

Finally, an immediate outcome could be a real-time version of the system: it would lead to the creation of a clinical tool that could help develop new rehabilitation paradigms and communication potentials.

# Introduzione

---

Un'interfaccia cervello-computer (in inglese, *Brain-Computer Interface*, abbreviato in BCI) è un sistema che consente di inviare comandi ad un computer attraverso l'analisi di segnali elettroencefalografici.

In assenza di metodi per riparare funzionalità motorie perse a seguito di malattie o incidenti, una delle opzioni per sostituire in parte le funzioni motorie compromesse è fornire al cervello nuovi canali di comunicazione e controllo, ovvero creare un'interfaccia che connetta direttamente il cervello ad un dispositivo esecutore del movimento e/o mezzo d'espressione del messaggio del soggetto al mondo esterno.

Il segnale più noto dell'attività encefalica è l'elettroencefalogramma (EEG), ovvero la registrazione dei potenziali encefalici attraverso elettrodi applicati sullo scalpo. In Neurologia, i segnali EEG sono usati soprattutto per valutare i disturbi/malattie neurologiche o per studiare il funzionamento del cervello; la recente disponibilità di hardware digitali sempre più potenti e lo sviluppo di nuovi algoritmi di analisi, ha portato l'EEG a divenire un potenziale mezzo di comunicazione diretto tra cervello e computer.

Vi Sono diversi tipi di attività cerebrale che possono essere usati per la realizzazione di un sistema BCI. In questo lavoro di tesi, ci siamo focalizzati su un particolare potenziale, la P300, includendo in alcune analisi la successiva onda lenta, chiamata in questo lavoro *rebound* e riconducibile alla componente *slow negative wave 2* (SNW2).

Attualmente sono presenti tre diversi orientamenti nello sviluppo di BCI:

1. BCI viste come un paradigma di apprendimento (1), secondo cui il processo riguarda principalmente l'utente, mentre la macchina adegua l'uscita in funzione del *feedback* ricevuto dal soggetto;
2. BCI viste come un problema di *machine learning* (2), secondo cui l'utente è visto semplicemente come la sorgente di un segnale predicibile, e l'unica

necessità è individuare questo segnale e codificarlo, senza l'esigenza di una fase di apprendimento da parte del soggetto.

3. BCI e soggetto visti come due sistemi dinamici interagenti (3), secondo cui il sistema di BCI ha l'obiettivo di selezionare le caratteristiche che il soggetto può controllare più efficacemente, e di ottimizzare la traduzione di questi segnali nel controllo di dispositivi a valle. L'ottimizzazione faciliterebbe l'apprendimento da parte dell'utente, che di contro migliorerebbe le *performance* del sistema di BCI.

Come vedremo, il nostro approccio (4) è una sintesi di questi tre orientamenti: vengono usate diverse interfacce grafiche secondo il presupposto che una combinazione di simboli ben precisa possa aiutare l'utente nell'apprendimento e nel processo di attenzione; uno degli strumenti usati è un classificatore logistico, che deve essere addestrato a distinguere i segnali in ingresso; un algoritmo genetico è utilizzato per selezionare le caratteristiche migliori per descrivere i segnali *target*, in modo che i comandi eseguiti dal computer seguano nel modo più fedele possibile le intenzioni del soggetto, aumentandone così l'apprendimento.

A tale scopo, l'esistenza di un sistema d'attenzione implicito è particolarmente rilevante per lo sviluppo delle BCI, poiché l'attenzione del soggetto è in grado di influenzare le caratteristiche della risposta allo stimolo: ciò consente di adottare un approccio cognitivo ai sistemi BCI, per sviluppare interfacce diverse e verificare se e quali di queste risultino maggiormente efficaci, con lo scopo finale di essere implementate in neuroprotesi.

Lo scopo del lavoro qui presentato è: verificare che il sistema da noi sviluppato abbia *performance* confrontabili con un sistema recentemente illustrato in letteratura; confrontare le prestazioni su diverse interfacce, verificando se le differenti prestazioni ottenute siano significative; stabilire se l'uso di registrazioni di durata maggiore, che possano includere altre componenti del segnale EEG, apportino effettivi e significativi vantaggi in termini di prestazioni.

# 1. Brain-Computer Interface

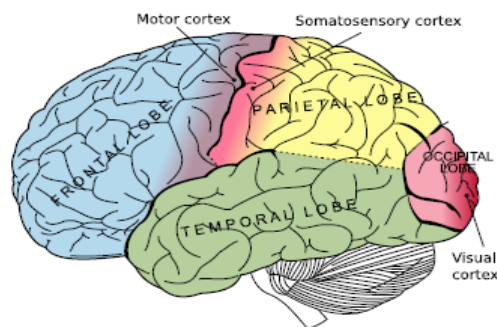
---

## 1.1 Il Sistema Nervoso Centrale

Il cervello, o ciò che comunemente viene definito tale, ovvero l'encefalo, è diviso in telencefalo e diencefalo; il telencefalo è responsabile delle funzioni di più alto livello, come pensare, pianificare e memorizzare; il diencefalo coadiuva alle funzioni di percezione cosciente delle sensazioni, dell'omeostasi e del ritmo circadiano.

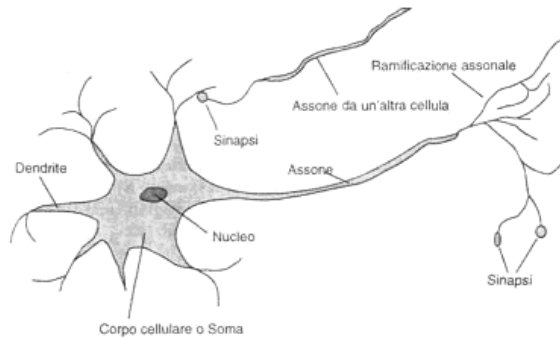
La corteccia cerebrale, lo strato esterno del telencefalo, è adibita alle funzioni più complesse: appare ripiegata a causa dei numerosi solchi, tra cui spiccano delle fenditure più profonde e ampie, dette scissure, che segnano il confine tra i diversi lobi.

La corteccia cerebrale è divisibile a grandi linee in aree di competenze specifiche [Figura 1]: ad esempio, la corteccia motoria (situata nella parte posteriore del lobo parietale, appena prima della scissura centrale) è responsabile dei movimenti volontari; la corteccia somato-sensoriale (sulla parte anteriore del lobo parietale) riceve informazioni sensoriali da parte del corpo; la corteccia visiva (nel lobo occipitale) riceve input dagli occhi.



**Figura 1** - Corteccia cerebrale divisa in lobi, su cui sono evidenziate le alcune aree di competenza specifiche.

L'elemento base del tessuto neuronale è il neurone [Figura 2]. I neuroni sono connessi tra loro da lunghe fibre che si protendono dai loro corpi cellulari: gli assoni, che costituiscono i terminali di uscita; i dendriti, che invece costituiscono i terminali d'ingresso.



**Figura 2** - Rappresentazione sintetica del neurone con le sue componenti principali

Tra un assone e i dendriti è presente una giunzione, detta sinapsi, tramite la quale un neurone comunica con i neuroni a esso connessi; il segnale attraversa questa giunzione per mezzo del rilascio di sostanze chimiche, i neurotrasmettitori, che, legandosi ai recettori sulla superficie dell'altro neurone, modulano la risposta del neurone pre-sinaptico verso il neurone post-sinaptico. Senza scendere in ulteriori dettagli, l'effetto finale è una variazione del potenziale di membrana: se il potenziale supera una soglia, si verifica l'impulso (potenziale d'azione<sup>1</sup>), che viaggia lungo l'assone verso le sinapsi successive.

L'attività complessiva e globale del cervello, che poi è quella che solitamente viene registrata ed analizzata, è una combinazione non lineare, spaziale e temporale dei potenziali d'azione dei singoli neuroni.

---

<sup>1</sup> Un potenziale d'azione è una rapida inversione del potenziale elettrico misurabile ai due lati di una membrana cellulare, cui fa seguito il ritorno all'iniziale condizione di riposo.

## 1.2 Segnale EEG

Sono diversi i metodi per osservare l'attività cerebrale. Possono essere divisi in due grandi classi: metodi che registrano direttamente l'attività dei neuroni, di questi fa parte l'EEG; metodi che misurano l'attività neuronale indirettamente, ad esempio attraverso la misura del flusso sanguigno.

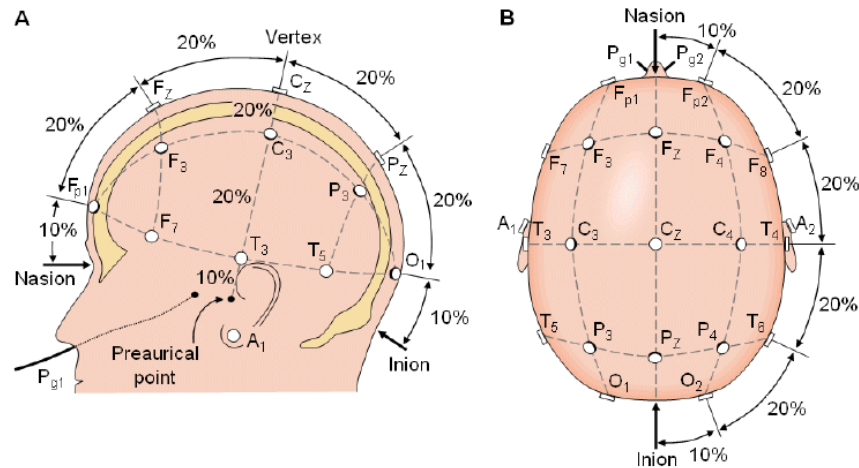
Nell'elettroencefalogramma si fa uso di elettrodi sullo scalpo che percepiscono le variazioni di potenziale elettrico, causate dai potenziali d'azione e dall'attività basale dei neuroni (5).

Sin dal primo articolo scientifico sui segnali EEG (6) sono state formulate ipotesi e sviluppate teorie sulla corrispondenza tra le registrazioni EEG e pensieri, intenti e decisioni, e soprattutto sulla possibilità, grazie a strumenti ed algoritmi, di tradurre e decifrare i segnali EEG in modo da poter controllare dispositivi elettronici attraverso un canale di comunicazione cerebrale diretto, bypassando i normali canali periferici e muscolari.

Il potenziale elettrico registrato dall'EEG riguarda per la maggior parte l'attività corticale appena sottostante, quindi è possibile mappare, anche se con risoluzione spaziale e precisione limite, l'intera corteccia attraverso l'uso di più elettrodi.

In passato uno dei problemi maggiori riguardava l'amplificazione del segnale, di ampiezza dell'ordine delle decine di microvolt; l'elettronica moderna permette di ottenere dispositivi compatti di alta qualità, non solo nell'amplificazione, ma anche nel *pre-processing*, con filtri passa-banda per le componenti non volute, o filtri arresta-banda per eliminare disturbi di rete (3).

Gli elettrodi sono piccoli dischi metallici, solitamente in oro o argento, a bassa impedenza, posti sullo scalpo. La posizione è di solito in accordo con le convenzioni internazionali: in questo modo, gli elettrodi sono posti sempre negli stessi punti per ogni soggetto, facilitando le operazioni di confronto tra studi differenti. Gli standard più usati sono il sistema 10-20 [Figura 3] e il sistema 10-10. Entrambi definiscono la posizione degli elettrodi in base alle misure antropometriche prese da quattro punti fissi, il nasione e l'inion lungo la linea mediana antero-posteriore ed i punti preauricolari lungo la linea coronale latero-laterale: mentre nel sistema 10-20 le distanze tra gli elettrodi sono del 10% o del 20% la dimensione di riferimento, nel sistema 10-10 le distanze tra un elettrodo e l'altro sono tutte il 10% della dimensione di riferimento.



**Figura 3** - Disposizione degli elettrodi nel sistema 10-20.

Affinché sia misurata una variazione di potenziale elettrico, è necessario definire un livello zero. Per quanto riguarda gli EEG, esistono diversi modi di definire un elettrodo di riferimento: può essere scelto un elettrodo di riferimento fisso, comune a tutti gli altri elettrodi, di solito posto sul mastoide o sul lobo dell'orecchio, e in tal caso si parla di riferimento comune; può essere computata la media di tutti o alcuni segnali e sottratta a ogni segnale, e in questo caso si parla di riferimento medio; il riferimento può essere un elettrodo o un segnale rispetto agli altri, e si parla di riferimento monopolare; alternativamente, si può definire un riferimento per ogni coppia di elettrodi, così che ogni canale registrato abbia il proprio riferimento, e si parla di configurazione bipolare.

BCI basate su EEG sono state per lungo tempo tralasciate per almeno tre problematiche (7):

1. La risoluzione e il grado di affidabilità di un sistema di BCI basato su EEG erano critici a causa della vastità di elementi neuronali attivi, da cui deriva la complessità e l'alta variabilità del segnale da *trial* a *trial*; la possibilità di riconoscere un singolo messaggio o comando da questa complessità e variabilità appariva remota se non impossibile;
2. Essendo la finalità una comunicazione *on-line*, la tecnologia e la strumentazione per ottenere comunicazioni basate su EEG risultavano costose ed eccessivamente sofisticate;
3. I primi esempi di sistemi di BCI basati su EEG avevano offerto delle prestazioni limitate, e avevano spinto ad abbandonare questi segnali in favore di metodi d'indagine più recenti.



Tuttavia, recentemente, grazie allo sviluppo di nuove tecnologie, l'approdo di nuove soluzioni hardware e software e la creazione di algoritmi sempre più efficienti, l'uso di EEG come base per la comunicazione computer-cervello è tornato in auge.

Uno dei motivi principali è stato la comprensione dei meccanismi di origine dei segnali EEG e delle relazioni tra specifiche funzioni e specifici pattern: numerosi studi hanno dimostrato la relazione tra EEG e movimenti attuati e/o pianificati e tra EEG e operazioni mentali (8), (9), (10). In tal modo è stato possibile considerare quali segnali EEG possano essere usati per comunicazione e controllo; un ulteriore impulso è stato dato, come detto, dallo sviluppo di hardware e software per l'analisi multicanale sempre meno costosi e sempre più sofisticati; infine, il riconoscimento sociale delle potenzialità inesprese di persone con malattie neurodegenerative o con alle spalle gravi incidenti a livello spinale, ha dato impulso ma anche giustificazione alla ricerca in ambito BCI, fornendo sistemi più efficienti e affidabili, tali da soddisfare le esigenze di quei disabili o in genere di quegli utenti che venivano frustrati dall'utilizzo dei primi esemplari di BCI.

Non meno importanti gli sviluppi nella tecnologia in uso in elettrofisiologia: oggi gli elettrodi epidurali possono fornire una maggiore risoluzione spaziale (11), benché tale caratteristica non sia critica nella maggior parte degli usi delle BCI.

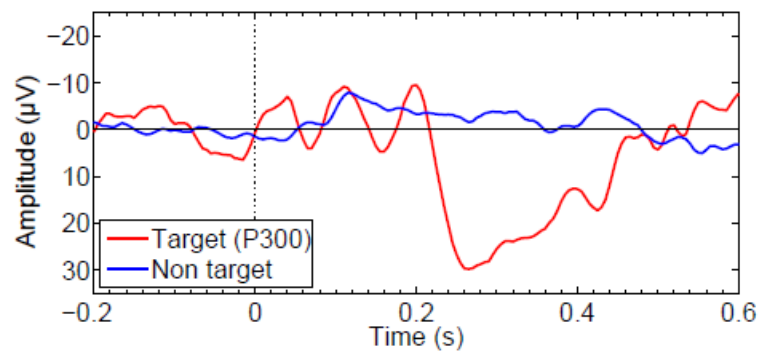
## 1.3 Potenziali evento-relati: P300

I potenziali evento-relati (in inglese, *Event-Related Potential*, ERP) sono forme d'onda evocate, successive a una stimolazione sensoriale o a un evento particolarmente significativo; essi si distinguono da quello che viene definito EEG spontaneo, costituito dall'attività elettrica di fondo, che può assumere specifici *pattern* in frequenza (bande elettroencefalografiche).

Sia le diverse bande elettroencefalografiche che gli ERP si caratterizzano per una propria morfologia ben distinguibile, con delle caratteristiche ben precise e misurabili. Il fatto che le caratteristiche varino, in relazione a un evento esterno piuttosto che a un'intenzione dell'individuo stesso, rappresenta il punto chiave per lo sviluppo di una BCI.

I potenziali evocati possono essere classificati in due grandi categorie, in funzione dei fattori che maggiormente li influenzano: i potenziali determinati dalla natura fisica degli stimoli impiegati per evocarli sono detti esogeni, mentre i potenziali correlati al significato dello stimolo sono detti endogeni.

La P300, o P3 [Figura 4], è comunemente definita come il correlato elettrofisiologico di un evento significativo per l'individuo, rilevante per il compito che sta svolgendo: stimoli uditivi, visivi o somato-sensoriali, rari o particolarmente significativi, se intervallati a stimoli frequenti o poco importanti, inducono nell'EEG l'emergere di un picco positivo a circa 300 ms, prevalentemente nella corteccia parietale (7), (12).



**Figura 4** - Grafico di una P300, ottenuta come media su molti trial in un paradigma oddball; l'onset è all'istante  $t=0$ .

Sebbene non siano ancora chiariti del tutto quali processi sono legati alla P300, c'è una posizione che più di altre ha saputo spiegare i risultati finora ottenuti ed ha saputo confutare le obiezioni sollevate: la *Context-Updating Theory* (13). Secondo questa teoria, se durante l'analisi dello stimolo il soggetto non rileva cambiamenti negli attributi dello stimolo, il modello mentale dello stimolo viene mantenuto e si registrerebbero solo componenti sensoriali (N100, P200, N200); viceversa, se viene percepito uno stimolo nuovo, i processi attentivi effettueranno un *updating* della rappresentazione mentale dello stimolo, e ciò porterebbe alla comparsa della P300. Questo semplice modello spiega come entrino in gioco processi cognitivi quali l'attenzione, intesa come energie cognitive impiegate per svolgere il compito, e la memoria a breve termine, che rappresenta il luogo dove è mantenuta la rappresentazione mentale dello stimolo, che viene di volta in volta confrontata con lo stimolo esterno.

Nel tracciato EEG la P300 si manifesta come un picco di ampiezza positiva (per questo P, *positive*, polarità che in genere nei tracciati viene rappresentata nella parte

inferiore rispetto alla *baseline*) con una latenza di 300-700 ms (da cui deriva il numero 300 che segue la lettera P), elicitato dalla comparsa di uno stimolo raro (14); l'altro modo in cui viene chiamata è P3, poichè nel tracciato EEG risulta essere il terzo picco rilevante di orientamento positivo, dalla comparsa dell'evento-stimolo (15).

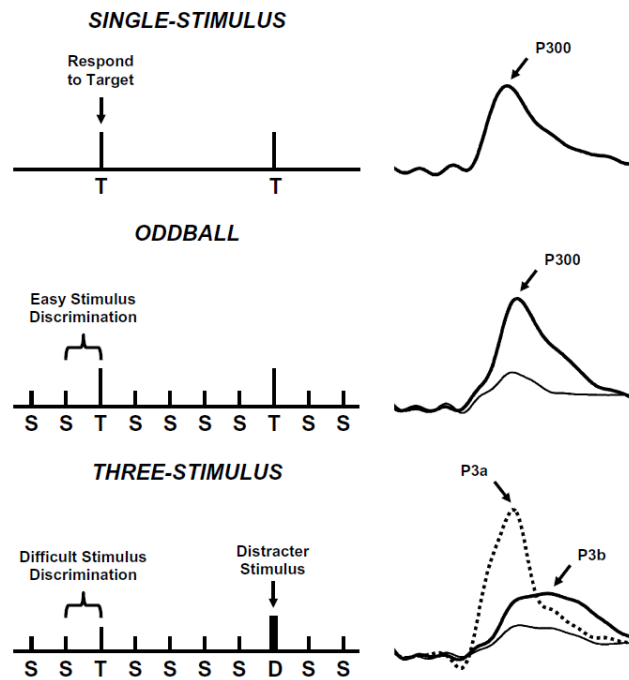
La P300 è un potenziale positivo endogeno che compare solamente in seguito a stimoli *target* (dall'inglese, "bersaglio"), ed ha distribuzione caratteristica sulle derivazioni posteriori (centro-parieto-occipitali) mediane (16); non riflette una specifica funzione cognitiva, ma è espressione globale dei molteplici processi cerebrali implicati nel mantenimento della memoria di lavoro (13).

L'ampiezza del picco della P300 è inversamente proporzionale alla frequenza di occorrenza dello stimolo *target*: man mano che la frequenza di comparsa dello stimolo *target* aumenta, l'ampiezza della P300 tende a diminuire, una chiara dimostrazione di come la frequenza d'occorrenza dello stimolo *target* sia un elemento che influenza fortemente soprattutto l'ampiezza della P300 e la forma che assume (17).

La latenza della P300 è invece direttamente correlata con il tempo impiegato dal soggetto per completare il pieno riconoscimento dello stimolo, e dipende quindi dalla complessità e dalla quantità dell'informazione veicolata dallo stimolo (18).

Il paradigma sperimentale che solitamente si usa per elicitarne una P300 è il classico *task oddball*: ad esempio, nell'*oddball* uditivo vengono fatti ascoltare due suoni di frequenza diversa, uno acuto con un'alta probabilità di occorrenza e uno di diversa tonalità, con bassa probabilità di occorrenza; viene chiesto di contare il numero di volte che viene presentato il suono con bassa probabilità d'occorrenza e di ignorare l'altro suono; in questo modo il suono *target*, atteso e raro, elicitava la P300, che non viene elicitata invece dal suono ad alta occorrenza.

Una variazione del *task* ha permesso di identificare le due varianti della P300; rispetto al tradizionale *oddball* con due stimoli, nell'*oddball* a tre stimoli (19), viene presentato sempre uno stimolo infrequente e significativo, ma intervallato da due diverse tipologie di stimolo, uno frequente e di routine, e un altro di distrazione [Figura 5].



**Figura 5** - Paradigmi di stimolazione diversi, in cui vengono evidenziate le differenze nelle forme d'onda a seconda del task eseguito; nella parte inferiore della figura, si evidenzia il task oddball a tre stimoli, in cui sono presenti due P300 di forma e latenza diverse (20).

In questo caso, il soggetto risponde diversamente ai due stimoli (20):

1. Nel caso di stimolo di distrazione, si ha come risposta la P3a (in questo caso, chiamata P3 *no-go*, poiché secondo il protocollo, il soggetto non deve rispondervi); la P3a è uno stimolo correlato a un generico orientamento spaziale e all'attenzione involontaria automatica; rappresenterebbe l'inizio del processo di raffronto tra l'*input* sensoriale e il contenuto della memoria a breve termine (21), in particolare in presenza di un segnale di distrazione atipico (ad esempio l'abbaiare di un cane tra segnali costituiti da toni o forme colorate tra segnali costituiti da lettere): per questo è chiamato anche *novelty* P3 (dall'inglese, "novità"); è una P300 frontale/centrale, con una latenza relativamente bassa (250-280 ms) e con un alto adattamento (22);
2. La classica P300, detta P3b, è invece la risposta a uno stimolo *target*, cui il soggetto deve rispondere con un *task* fisico (premere un pulsante) o mentale (un'operazione di conteggio): è un fenomeno relativo all'attenzione volontaria, e segnerebbe la chiusura post-decisionale, una funzione specifica dei processi cognitivi in riferimento all'interruzione delle elaborazioni (quindi un fenomeno inibitorio), essenziale nell'organizzare azioni cognitive

seriali, in modo da permettere di re-inizializzare i processi di raffronto quando una parte di operazioni cognitive è stata completata e un'altra parte deve essere iniziata (21). La chiusura post-decisionale sarebbe essenziale per l'aggiornamento degli stimoli *target* nella memoria a breve termine.

La P3b ha una latenza maggiore e molto variabile da *task* a *task* (dai 300 ai 700 ms), con ampiezza massima in zona parietale (20).

La P300 è stata quindi usata come risposta particolare o rara in ingresso a una BCI (23), (24). La P300 è particolarmente evidente solo nelle risposte evocate dal messaggio/simbolo scelto dall'utente, o indicatogli, e quindi può essere usata dalla BCI per determinare l'intento dell'utente. A prescindere dall'algoritmo usato per riconoscere la P300, resta valida la possibilità per le BCI (con maggiore o minore accuratezza ed affidabilità) di usare la P300 per il controllo dell'ambiente da parte del soggetto (25).

Il vantaggio immediato dell'uso della P300 come base della comunicazione nella BCI è che non necessita di una fase di *training* iniziale da parte dell'utente, in quanto è una tipica risposta nativa; allo stesso tempo, tuttavia, è stato dimostrato che la P300 cambia in relazione ai protocolli di condizionamento (26), (27) e durante le diverse sessioni di BCI, soprattutto a lungo termine (28): le *performance* della BCI possono quindi deteriorarsi nel tempo se avviene un fenomeno d'abitudine allo stimolo, o possono migliorare, se avviene un fenomeno di rafforzamento della risposta.

## 1.4 Il *rebound*

È stata universalmente accettata la classificazione tra le possibili onde lente negative, sulla base delle diverse latenze e sui siti dello scalpo dov'è possibile osservarle: *contingent negative variation* (CNV)<sup>2</sup>, correlata con un processo d'attesa; potenziale anticipatorio, che segnali l'intenzione all'azione; naturale deflessione della P300 in determinati *task* (29).

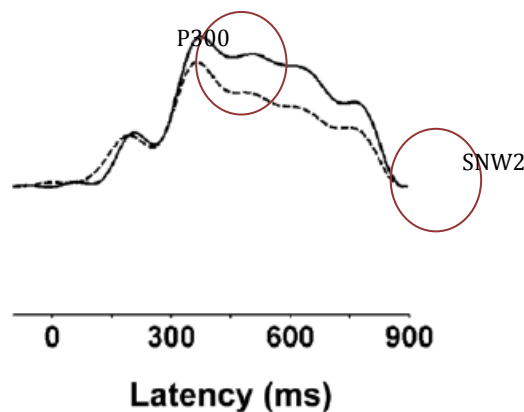
Tra queste, d'interesse nel nostro studio è l'onda lenta post-P300, solitamente chiamata onda lenta negativa 2 (in ingl., *slow negative wave 2*, SNW2) o risposta

---

<sup>2</sup> È una deflessione negativa che si verifica quando l'individuo attende il verificarsi di un segnale o è stimolato da un segnale inaspettato e nuovo (7).

d'orientamento (OR) [Figura 6], predominante in siti frontali o centrali dello scalpo, anche se presente su tutto lo scalpo (30).

Usando un'analisi alle componenti principali (18), è stato mostrato che l'ERP può essere scomposto in due componenti sovrapposte: la P300 [Figura 4], di cui abbiamo già discusso [1.3 Potenziali evento-relati: P300]; e un'onda lenta, con latenza 400-800 ms, negativa in area frontale e centrale, positiva in area parietale. Mentre la P300 cresce in funzione della detettabilità del segnale, l'onda lenta mostra una relazione inversa: ciò ha portato all'ipotesi che l'onda lenta dipende da quanto uno stimolo sia inaspettato o difficile da ricondurre a uno stimolo noto; dal momento che la P300 è coinvolta in un inizio di valutazione cognitiva del segnale, una crescita in ampiezza dell'onda lenta potrebbe dipendere da un processo cognitivo prolungato, richiesto nei casi in cui la decisione è più difficile (18).



**Figura 6** - Componenti di un segnale EEG target, in cui si evidenziano le componenti oggetto di questo lavoro di tesi.

Negli anni '80 ha raccolto consensi l'interpretazione di quest'onda (in particolare se frontale e centrale) come una componente anticipatoria della CNV (31); nello stesso contesto, è stata proposta la P300 come l'inizio di un processo di aggiornamento detto fase di orientamento, in cui il cervello modifica la propria rappresentazione mentale del contesto ambientale in cui si trova, rispondendo a una sequenza di compiti cognitivi significativi (13): benchè si rifletta essenzialmente nella P300, questa fase continuerebbe poi con l'emergere di una negatività tardiva (400-600 ms); tale negatività è stata spesso correlata al fenomeno di *readiness* (dall'inglese, preparazione), il cui corrispettivo fenomeno elettro-fisiologico è appunto la CNV (32).

La *readiness* è l'abilità del cervello di rimanere concentrato su uno stimolo senza distrarsi a causa di stimoli d'interruzione, transitori e non significativi.

Più recenti studi hanno confermato che la SNW2 è distintiva di stimoli *target* rispetto a stimoli di routine, in particolare stimoli significativi, inaspettati e nuovi (33). Tuttavia, non è stata ancora trovata una relazione incontrovertibile tra questa attività tardiva e la fase di orientamento, anche se l'adattamento osservato a seguito di numerose ripetizioni può corroborare quest'ipotesi (34).

## 1.5 Meccanismi attentivi

L'attenzione è una funzione cognitiva che può influenzare la forma della P300; in effetti, la localizzazione degli stimoli ambientali e l'organizzazione di azioni appropriate in risposta ad essi sono strettamente legati al funzionamento dei sistemi attenzionali. In particolare, alcuni processi, specifici dell'attenzione visiva spaziale, hanno profondi effetti sulla prestazione e sui potenziali evento-relati (35).

L'orientamento dell'attenzione può essere di due tipi: implicito o esplicito. L'orientamento esplicito è caratterizzato dal movimento degli occhi e/o del capo e viene attivato in corrispondenza della rilevazione di un oggetto di interesse nella periferia del campo visivo; grazie all'orientamento esplicito, la fovea viene allineata con l'oggetto da esplorare.

L'orientamento implicito, invece, è caratterizzato dallo spostamento dell'attenzione in assenza di movimenti corporei finalizzati all'allineamento della fovea con l'oggetto da esplorare.

Uno studio sperimentale, poi divenuto paradigma classico nella ricerca sull'attenzione visiva, ha indagato sugli effetti dell'orientamento implicito dell'attenzione, sulla velocità dei tempi di reazione a stimoli visivi, sia endogeni<sup>3</sup>, dovuti a processi mentali, sia esogeni<sup>3</sup>, dovuto a stimoli anticipatori esterni (36); il compito dei partecipanti era rilevare la presenza di uno stimolo visivo che potesse apparire solo in uno di due lati di uno schermo, diametralmente opposti rispetto a un punto di fissazione centrale; quando lo stimolo era presente, i partecipanti dovevano

---

<sup>3</sup> In questo caso, i termini endogeno ed esogeno sono riferiti allo stimolo, come sinonimi di evento interno (endogeno) ed esterno (esogeno).

premere un tasto il più rapidamente possibile; durante l'esecuzione della risposta venivano registrati i tempi di reazione; l'orientamento dell'attenzione era implicito esogeno, in quanto il partecipante doveva rispondere a uno stimolo *target* posto alla periferia del campo visivo mentre manteneva lo sguardo sul punto di fissazione centrale, senza alcun *cue* (dall'inglese, avvertimento) anticipatorio.

Nella condizione di orientamento implicito endogeno, venivano presentati *cue* centrali, che anticipavano correttamente il segnale nell'80% dei casi: si è registrato un beneficio nelle prestazioni, ossia una riduzione dei tempi di reazione e una maggior accuratezza nelle risposte nei *trial* in cui il *cue* anticipava correttamente lo stimolo. Dal momento che i partecipanti non potevano spostare lo sguardo dal punto di fissazione, si suppone che gli effetti delle diverse condizioni sui tempi di reazione indichino l'entrata in azione di meccanismi attentivi. Tali meccanismi aumenterebbero o diminuirebbero la responsività al segnale nelle posizioni attese: ci saranno dei benefici nel caso di *cue* corretti e, viceversa, dei costi nei *cue* non corretti, in quanto il partecipante deve distogliere l'attenzione dalla posizione indicata dal *cue* e spostarla nella posizione di effettiva comparsa dello stimolo. Riassumendo, sono emersi due sistemi dell'orientamento dell'attenzione:

1. Uno endogeno, controllato intenzionalmente ed evocato da *cue* anticipatori centrali;
2. Uno esogeno, in cui l'attenzione si sposta automaticamente nello spazio, senza processi cognitivi o attentivi, ed è attivato da stimoli periferici.

Di conseguenza possiamo riassumere dicendo che l'orientamento dell'attenzione può essere elicitato passivamente da stimoli periferici (orientamento esogeno), o essere diretto volontariamente dall'individuo nelle diverse posizioni spaziali (orientamento endogeno).

Due sono le conclusioni che risultano interessanti ai fini del nostro progetto:

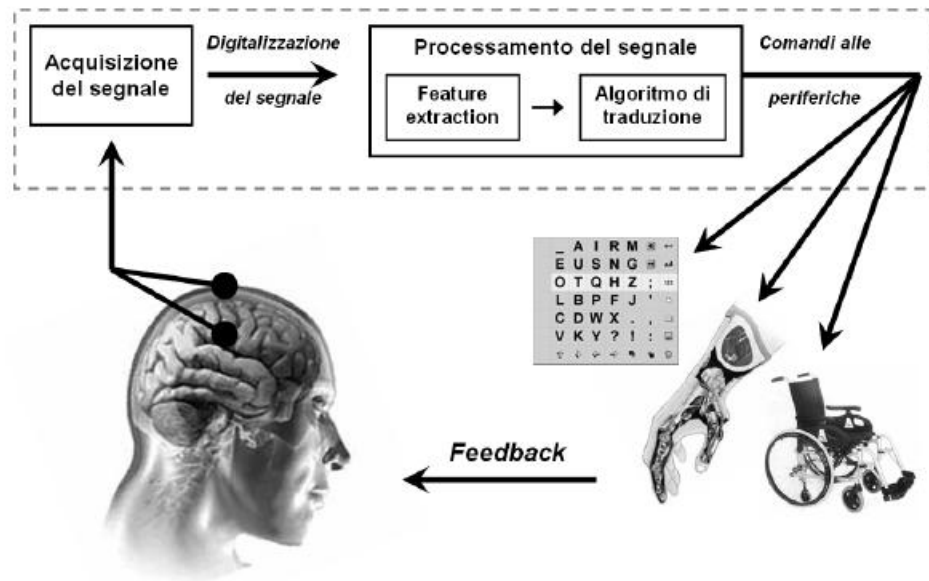
1. La prima riguarda l'esistenza di un sistema d'attenzione implicito. Questo dato è particolarmente rilevante nel campo delle BCI, in quanto sono tecnologie che nel prossimo futuro si possono evolvere in neuroprotesi, soprattutto per individui con gravi disabilità motorie. Il fatto che sia dimostrata l'esistenza di un sistema attentivo che non debba essere supportato da alcun apparato muscolare (come i movimenti oculari) rappresenta un vantaggio per questo nuovo tipo di tecnologia;



2. La seconda riguarda il fatto che l'attesa del soggetto è in grado di influenzare le *performance* della risposta allo stimolo (riduzione dei tempi di reazione e maggior accuratezza). Questo grazie all'esistenza di diversi processi attentivi che possono accompagnare e facilitare (ma anche svantaggiare, come abbiamo visto) l'azione. L'esistenza di questi diversi meccanismi attentivi ci consente di adottare un approccio cognitivo ai sistemi BCI. Si possono quindi sviluppare interfacce diverse e verificare quale di queste risultino maggiormente efficaci, con lo scopo finale di essere implementate in neuroprotesi.

## 1.6 Classificazione delle BCI

Un'interfaccia cervello-computer (in inglese, *Brain-Computer Interface*, abbreviato in BCI, o anche *Brain-Machine Interface*, BMI) è un sistema di comunicazione in cui i messaggi o i comandi che l'utente normalmente invia per agire nel mondo esterno non passano attraverso la canonica via costituita dai nervi periferici e dai muscoli, ma usa come codificatore ed attuatore il computer ed i dispositivi ad esso collegati. Per far ciò, è necessaria un'analisi numerica che interpreti l'attività cerebrale letta all'interfaccia: ad esempio, nelle BCI basate su EEG [Figura 7] il messaggio è codificato in forma di segnali EEG.



**Figura 7** - Componenti e funzionamento di base di una BCI. Il successo del processo dipende dall'interazione e dal reciproco adattamento dell'utente e del sistema: il primo deve mantenere una buona correlazione tra le intenzioni e l'attività cerebrale registrata dalla BCI; il sistema BCI deve permettere la corretta selezione delle caratteristiche del segnale che il soggetto può manipolare, tradurre tali caratteristiche in comandi corretti ed efficaci per la periferica.

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

1. Le BCI dipendenti non usano la normale via neuromuscolare, ma utilizzano vie neuromuscolari alternative, senza la necessità quindi di bypassare completamente il tratto muscolare periferico; essenzialmente sostituiscono una via interrotta o deteriorata con un'altra pre-esistente, che serva per generare l'attività cerebrale che codifica il messaggio da inviare al computer. È il caso, ad esempio, dei sistemi di BCI caratterizzati da una matrice di lettere che s'illuminano una alla volta: il soggetto seleziona la lettera scelta allineandovi lo sguardo, così da elicitarne un potenziale evocato visivo (VEP), che risulta essere più ampio dei potenziali evocati dalle altre lettere (37).
2. Le BCI indipendenti non dipendono in alcun modo dalle normali vie d'uscita neuromuscolari. Il messaggio in questi sistemi non è convogliato in vie neuromuscolari o sensorie, e non è necessario l'intervento di queste vie per far emergere l'attività cerebrale necessaria alla codifica del messaggio. È il caso, ad esempio, dei sistemi di BCI in cui tramite la solita matrice di lettere che si illuminano in modo consequenziale viene dettato il potenziale evocato

P300, correlato e sincronizzato con la scelta del soggetto (23), (24). In questo caso, l'*output* registrato, utile alla codifica del messaggio, è il segnale EEG, che non è veicolato da nessuna via neuromuscolare periferica, ma dipende solo dalle intenzioni dell'utente, e non dalla direzione dello sguardo (14), (13), (38). Bypassando l'attività periferica, i sistemi di BCI indipendenti risultano di maggior interesse teorico sulle funzioni del cervello rispetto ai BCI dipendenti, e di maggiore utilità, specialmente per pazienti con le più gravi disabilità neuromuscolari.

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

1. Le BCI invasive consistono essenzialmente in matrici di elettrodi impiantate nella corteccia motoria o premotoria di soggetti paralizzati (39). Ereditando l'approccio robotico, ovvero l'uso di esecutori meccatronici direttamente interfacciati al soggetto, si basano sull'ipotesi che le abilità motorie possano essere ricostruite a partire dalla codifica diretta dei segnali neuronali indirizzati ai tratti periferici neuromuscolari (40); sviluppate nella pratica clinica, sono state un tentativo di sostituzione totale o parziale di funzioni motorie, principalmente a livello di arti superiori (41).

Le BCI invasive richiedono l'impianto di microelettrodi tramite intervento chirurgico; se l'attività che si vuole registrare comprende un'area cerebrale ampia, possono essere utilizzate delle vere e proprie griglie applicate in porzioni d'interesse sulla corteccia (42); l'utilizzo di una matrice di elettrodi consente la registrazione del segnale su una superficie di corteccia maggiore: questo permette di distinguere diversi *pattern* d'attivazione corticale e di utilizzare quest'informazione come punto di partenza per una BCI.

Un recente studio ha dimostrato come le BCI basate su segnale ECoG<sup>4</sup> implementate su individui con gravi disabilità motorie e di comunicazione, abbiano maggiori potenzialità in termini di efficacia rispetto a una BCI basata su segnale EEG; tali vantaggi sono riassumibili in: un miglior segnale registrato (più stabile e di maggior ampiezza); una riduzione dei tempi di training; un numero maggiore di gradi di libertà da poter controllare (in

---

<sup>4</sup>L'elettrocorticogramma (ECoG) è la registrazione dell'attività cerebrale, sia spontanea sia evocata, sulla corteccia.

riferimento alle direzioni spaziali, che siano su cursore o a monitor); migliori prestazioni, sia per quanto riguarda la precisione di classificazione del segnale, sia per quanto riguarda la velocità di trasmissione dell'informazione (43).

Nonostante questi vantaggi, in un questionario a più di 100 ricercatori alla *BCI-conference 2005*, organizzata da Wolpaw e Vaughan a New York, la maggioranza ha indicato le tecniche non invasive preferibili per il prossimo futuro, dal momento che le BCI invasive hanno uno svantaggio che supera tutti i potenziali vantaggi, ovvero la necessità di un rischioso e complesso intervento chirurgico, che attualmente rende auspicabile la scelta di una BCI non invasiva, se non per soggetti con disabilità estremamente gravi (3).

2. Le BCI non invasive riprendono e fanno proprie le tradizioni dell'elaborazione dei segnali, dell'intelligenza artificiale e della neuropsicologia come basi teoriche e strumenti tecnologici per il controllo e la comunicazione tra cervello e macchina, non più semplice dispositivo esecutore del movimento, ma interprete delle intenzioni del soggetto, fino alle più alte funzioni cognitive e di pianificazione (44).

L'intento è usare i segnali definiti di *neurofeedback*, segnali di condizionamento del sistema nervoso centrale (CNS), tramite cui è possibile apprendere e/o modificare funzioni motorie, senza alcuna mediazione del sistema somatico-muscolare (45). L'impulso più importante lo hanno dato: la scoperta dell'autoregolazione delle onde cerebrali (46), ovvero la modifica di caratteristiche del segnale EEG a seconda del comportamento, delle condizioni ambientali e dell'addestramento del soggetto, senza la mediazione delle vie periferiche somatiche e neuromuscolari; la scoperta dell'apprendimento e di una forma di controllo volontario nei segnali cerebrali dopo una fase di *training* (47).

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

totale scomparsa di risposta muscolare (includendo anche muscoli oculari), condizione definita come *completely locked-in state* (CLIS) (1).

Con il tempo, le differenze tra i filoni BCI invasive e non invasive e tra le BCI dipendenti ed indipendenti ha lasciato spazio a una concezione più globale della BCI, fino all'ipotesi estremamente recente di una coesione dei diversi approcci in un unico sistema ibrido (48).

## 1.6 Le componenti di una BCI

Nella costruzione di un sistema di BCI è necessario fare delle scelte preliminari, a cominciare dal segnale biologico da acquisire, passando per la tecnica di acquisizione, la modalità con cui il segnale biologico viene elicitato, per finire con la registrazione ed elaborazione del segnale e il metodo di analisi e traduzione del segnale.

Due sono le componenti essenziali di cui necessita ogni BCI (3), (49):

1. Un sistema hardware dedicato alla presentazione degli stimoli sensoriali, all'acquisizione del segnale EEG (elettrodi, amplificatore, filtri, ecc.) alla registrazione di risposte comportamentali e alla presentazione del *feedback* (visivo, acustico, tattile);
2. Un sistema per l'elaborazione e la classificazione del segnale.

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

I blocchi fondamentali sono i seguenti [Figura 7]:

1. *Acquisizione del segnale*: in questo caso vanno sottolineate le differenze tra segnali di risposte evocate ed attività spontanee; le risposte evocate (come la P300) sono risposte del cervello a stimoli sensori stereotipati provenienti in questo caso dal sistema di BCI, quindi l'*onset* è a carico del sistema di acquisizione; l'attività spontanea (come i ritmi EEG) non dipende dalla generazione di stimoli da parte del sistema di BCI, quindi l'*onset* è dato dall'utente; nella parte di acquisizione del segnale, l'*input* scelto è acquisito dagli elettrodi, amplificato e digitalizzato.

2. *Processing del segnale*: le BCI possono usare caratteristiche del segnale nel dominio del tempo o della frequenza, oppure usare entrambi i domini, soluzione che potrebbe portare a un miglioramento delle *performance* (50); in genere, possiamo dire che gli attuali BCI estraggono e riconoscono caratteristiche che riflettono eventi cerebrali ben definiti ed identificabili (particolari onde o ritmi); è possibile per una BCI usare caratteristiche che non siano strettamente o direttamente correlate a uno specifico evento cerebrale, ma che comunque risultino di rilevanza per stabilire il momento e il tipo di intento dell'utente; in questo caso, è particolarmente importante garantire che la caratteristica scelta non sia contaminata da EMG, EOG o altri artefatti del SNC.
3. *L'algoritmo di traduzione*: la fase successiva è interessata alla traduzione delle caratteristiche, estratte o riconosciute, in comandi ai dispositivi a valle, in modo da poter esprimere in maniera corretta l'intento dell'utente. Questo algoritmo può usare metodi lineari (la classica analisi statistica), o non lineari (ad esempio le reti neurali); qualunque sia la sua natura, ogni algoritmo ha come obiettivo la traduzione dei segnali elettrofisiologici in segnali di controllo. L'efficienza dell'algoritmo ad adattarsi a ogni singolo utente si può valutare su tre livelli:
  - a. Al primo accesso del nuovo utente, l'algoritmo si adatta alle particolari caratteristiche del segnale; come esempio concreto, parlando di BCI basate su P300, l'algoritmo si adatta alle caratteristiche di ampiezza e latenza della P300 dell'utente connesso; una BCI che ha solo questo livello di adattamento, continuerà ad avere *performance* costanti se e solo se le risposte ai segnali da parte dell'utente resteranno costanti;
  - b. Poiché i segnali EEG ed altri segnali elettrofisiologici mostrano in genere variazioni a breve e lungo termine, legati ai ritmi circadiani ed alle condizioni fisiche, ormonali o ambientali, è necessario che sia presente un secondo livello di adattamento, ovvero un periodico adeguamento per ridurre l'impatto di variazioni spontanee; un buon algoritmo di traduzione deve essere in grado di adattarsi in modo da ottenere la massima corrispondenza tra l'attuale *range* di valori delle caratteristiche di segnale dell'utente e il *range* di valori disponibili dei dispositivi da controllare.

- c. L'ultimo livello di adattamento è in realtà l'adattamento auspicabile per ogni sistema di BCI, ovvero riflettere le capacità adattative del cervello e rispondervi in modo rapido ed affidabile: il cervello così potrà modificare il segnale d'uscita in modo da migliorare le operazioni BCI, come se fosse un canale neuromuscolare, ottenendo non solo un costante miglioramento delle *performance* del sistema di BCI, ma incoraggiando l'utente a sviluppare una nuova abilità e, in ultima istanza, un nuovo metodo di comunicazione.
4. *Il dispositivo d'uscita*: per la maggior parte delle attuali BCI, il dispositivo d'uscita è il monitor di un computer e l'uscita (auspicabile) è la selezione di un obiettivo, una lettera o un'icona presente su di esso; in altre BCI l'uscita è il movimento di un cursore verso l'alternativa prescelta. L'uscita è anche un *feedback* che l'utente usa per testare la capacità del sistema di rispondere al suo intento, espresso con le caratteristiche del segnale che la BCI traduce in comandi.
5. *Il protocollo operativo*: ogni BCI ha un protocollo che guida le operazioni, pianificate in modo specifico in relazione all'interfaccia o allo stimolo presentato all'utente, al segnale registrato ed alla tipologia di BCI con cui si lavora, che definisce l'interazione tra utente e sistema e quale *feedback* è restituito all'utente, ammesso che esso sia presente.

## 1.7 BCI non invasive: alcuni approcci applicativi

Esistono numerosi metodi per monitorare l'attività cerebrale in funzione di un suo utilizzo come *input* a sistemi di BCI non invasive: dall'elettroencefalografia (EEG) a metodi d'indagine più sofisticati e di *imaging*, come la tomografia ad emissione di positroni (PET) e la risonanza magnetica funzionale (fMRI).

Ad oggi, solo l'EEG consente una risoluzione temporale sufficiente all'utilizzo in sistemi di BCI inseriti negli ambienti e con le finalità più diversi, richiedendo un equipaggiamento poco costoso e relativamente semplice (3).

Per qual che riguarda le P300-BCI, vista l'importanza dell'argomento nel contesto del nostro lavoro, verranno trattate separatamente.

### 1.7.1 MEG

La magnetoencefalografia (MEG) è una tecnica che consente di studiare la funzionalità cerebrale tramite la misura del campo magnetico generato dalle correnti intracellulari dei neuroni piramidali corticali. I vantaggi di questa tecnica risiedono nella possibilità di identificare con ottima risoluzione temporale e buona risoluzione spaziale le aree cerebrali attivate durante specifici processi.

Rispetto all'EEG, la MEG difetta per due essenziali ragioni: è maggiormente sensibile a fonti di rumore esterne; l'apparecchiatura per MEG è molto costosa oltre che molto ingombrante.

### 1.7.2 fMRI

La risonanza magnetica funzionale (fMRI) è una tecnica di *imaging* biomedico che consiste nell'uso della risonanza magnetica per valutare la funzionalità di un organo o un apparato, in maniera complementare all'*imaging* morfologico. Il segnale misurato dalla fMRI, utilizzato come base di una BCI che utilizzi tale metodologia (51), è il livello di ossigenazione del sangue (in ingl., *Blood Oxigenation Level Dependent (signal)*, BOLD), un indice indiretto dell'attivazione di una determinata area cerebrale: a un maggior consumo di ossigeno in una determinata area cerebrale, corrisponde una maggior attività cerebrale.

Purtroppo, come per la tecnologia MEG, anche la fMRI ha grossi limiti pratici per quel che riguarda costi e dimensioni delle attrezzature necessarie, e in più vi è la presenza di un ritardo (3-6 secondi) tra l'attività cerebrale e il segnale fMRI.

### 1.7.3 NIRS

La NIRS (in ingl., *Near Infrared Spectroscopy*) è una tecnica non invasiva in grado di misurare l'ossigenazione dei tessuti attraverso la radiazione nel vicino



infrarosso (700-950 nm); l'assorbimento di questa radiazione da parte del tessuto biologico è basso ed è principalmente dovuto all'emoglobina; l'emoglobina ossigenata e l'emoglobina deossigenata hanno spettri di assorbimento differenti e questa caratteristica permette di misurare la concentrazione d'ossigeno nei tessuti studiati, *input* di un sistema di BCI (52).

I vantaggi della NIRS sono: la facilità di applicazione rispetto l'EEG, costi e dimensioni notevolmente ridotti rispetto fMRI e MEG, la buona risoluzione spaziale, la capacità di localizzare l'informazione e l'alta sensibilità nel rilevare piccole concentrazioni di sostanze. D'altra parte la NIRS difetta nel tempo di latenza della risposta emodinamica, dell'ordine dei secondi; inoltre subisce più di altre tecniche l'interferenza di artefatti dovuti a piccoli movimenti o ai capelli (53).

## 1.7.4 Potenziali corticali lenti

I potenziali corticali lenti (in ingl., *Slow Cortical Potentials*, SCPs) sono potenziali a bassa frequenza (DC-2 Hz) registrati sullo scalpo e associati a vari eventi cognitivi o sensori-motori.

Le BCI basate sugli SPCs sfruttano il fenomeno del condizionamento operante, una procedura di modifica del comportamento (inteso anche come funzionamento di un sistema o di un organismo) sulla base di periodi di apprendimento. Tale apprendimento, nell'ambito cognitivo, procede a stadi successivi, al termine dei quali il soggetto apprende *task* cognitivi sempre più complessi (dal semplice puntamento di un cursore a video, alla selezione di lettere da una matrice di simboli) (54).

## 1.7.5 Potenziali motori

I movimenti sono accompagnati da potenziali transienti detettabili sulla superficie dello scalpo: una deflessione negativa, detta potenziale di *readiness*, precede movimenti auto-indotti; in aggiunta, movimenti immaginati elicitano potenziali simili a quelli indotti dall'esecuzione di movimenti (55); la regione dello scalpo dov'è possibile registrare questi movimenti varia con la natura del movimento stesso.

L'uso di potenziali motori nei sistemi di BCI (56) ha modificato l'approccio basato su stimoli esterni, evidenziando la possibilità di paradigmi in cui è l'utente che

fornisce l'*onset* al sistema: per questo motivo, questi sistemi sono stati chiamati BCI asincroni, in quanto non è dato dal sistema un segnale di *onset*, ma è tutto deciso dal soggetto; addestrare un sistema di BCI asincrono senza la verifica che le intenzioni dell'utente corrispondano al movimento eseguito non rende affidabili questi sistemi e richiede tempi molto lunghi da parte dell'utente.

## 1.8 P300-BCI

La P300 è un potenziale positivo, ampio, studiato in forma estensiva nel contesto del paradigma *oddball*. La prima comunicazione BCI che abbia utilizzato la P300, chiamata *P300-speller*, prevedeva una matrice 6×6 di simboli (26 lettere dell'alfabeto e alcuni comandi funzionali), di colore grigio su sfondo nero, le cui colonne e righe erano alternativamente e in modo casuale rese più luminose [Figura 8]: una P300 veniva prodotta quando la colonna o la riga del simbolo scelto erano illuminate; il simbolo scelto era selezionato tramite la combinazione delle risposte su righe e colonne (23).



**Figura 8** - Matrice 6x6 usata nel *P300-speller* come stimolo visivo; righe e colonne sono illuminate una alla volta in modo casuale.

Molti lavori hanno ripreso l'idea della *P300-speller*, ossia di una matrice di simboli, in cui righe e colonne lampeggiano in sequenza casuale. Successivamente, è stata usata la solita matrice 6×6 per condurre uno studio comparativo tra partecipanti normali e pazienti con SLA, al fine di verificare l'eventuale efficacia di un sistema di comunicazione per pazienti con gravi disabilità motorie (24); tale lavoro, inoltre, ha

dimostrato come lo sviluppo di un sistema P300-BCI funzionale sia un problema multidisciplinare: oltre a sviluppare paradigmi sperimentali più efficaci, è necessaria l'evoluzione di algoritmi di classificazione più accurati e sistemi di acquisizione del segnale più efficienti (24); infatti, studi recenti hanno dimostrato che i risultati migliorano se vengono utilizzati sistemi d'analisi più potenti (49), (57), (58).

Un problema che si è subito presentato è stata la bassa velocità di trasmissione, che risulta essere uno degli svantaggi più limitanti delle BCI non invasive, soprattutto se confrontate alle *performance* delle BCI invasive: sebbene la P300 sia il segnale d'origine più rapido tra i segnali EEG, la velocità di comunicazione (in ingl., *bit transfert*) tra cervello e computer raramente supera i 25 bit/min.

Nel tempo sono stati testati quindi diversi intervalli inter-stimolo (ISI), per verificare se la velocità di presentazione dello stimolo risultasse significativa nelle prestazioni della BCI, e si potesse pertanto ridurre in modo da aumentare la velocità di comunicazione (59), (60); benché la questione sia ancora aperta, la maggior parte degli studi più recenti riporta un significativo vantaggio in termini temporali con ISI più brevi a parità di accuratezza (61), (60).

Anche se il paradigma che utilizza una matrice di lettere e simboli è stato quello più esplorato ed indagato nel campo delle P300-BCI, sono stati proposti anche altri tipi di interfaccia. Alcuni esempi interessanti sono quelli riportati nei lavori con un ambiente virtuale (un appartamento con oggetti): le prestazioni ottenute sono in linea con i precedenti lavori sulla P300-BCI (25).

Un metodo alternativo (62) prevede un'interfaccia a quattro frecce direzionali o quattro scelte (nel caso specifico: "YES", "NO", "PASS", "END") ai lati dello schermo, con il classico lampeggiamento casuale come stimolo; per la P300-BCI con questa interfaccia, sono state confrontate le prestazioni tra una modalità visiva, una modalità acustica e una terza modalità che prevedeva la simultanea presentazione di stimoli acustici e visivi<sup>5</sup> (62). Le conclusioni principali di questo lavoro sono state le seguenti:

1. Una BCI basata sulla P300 può essere effettivamente utilizzata come protesi comunicativa per pazienti con SLA;

---

<sup>5</sup> Da notare come la modalità con presentazione acustica potrebbe essere una buona risposta alla critica sollevata verso i sistemi di BCI basati sulla P300, ovvero di porre troppa fiducia sulle interfacce visive, senza tener conto dell'impatto che la SLA potrebbe avere sia sul sistema visivo, sia sul sistema attentivo dei pazienti.

2. Sia la modalità visiva che la modalità uditiva di presentazione dello stimolo possono essere utilizzate per elicitare la P300.

Arrivati a questo punto della ricerca, si cominciano a muovere i primi passi per rendere queste tecnologie oltre che efficaci, anche fruibili nel quotidiano da soggetti patologici. Per far questo non bisogna solo migliorare le interfacce o perfezionare i classificatori ma bisogna anche rendere a misura d'utente la tecnologia EEG, i software utilizzati, etc.

Un primo esempio è un prototipo di BCI installato nelle case dei pazienti e connesso ai centri di studio-ricerca-analisi tramite Web (63).

## 2. Metodi

---

Un sistema di BCI prevede:

1. Una fase di selezione delle caratteristiche del segnale; vengono analizzati i segnali EEG e si tenta di estrarre le caratteristiche peculiari del segnale;
2. Una fase in cui le caratteristiche sono utilizzate per classificare i segnali;
3. Il calcolo delle *performance* del classificatore.

Ognuna di queste fasi è implementata con diversi strumenti matematici; nel nostro caso avremo:

1. Per la prima fase, i segnali EEG sono legati a uno spazio delle *feature*, in cui le singole *sweep*, appartenenti a una delle due possibili classi, tendono a separarsi; la ricerca della soluzione migliore è affidata a un algoritmo genetico progettato *ad hoc* per l'applicazione;
2. Nella seconda fase, vengono assegnate delle etichette (in ingl., *label* o *class*) agli elementi nello spazio delle *feature*, e si procede alla classificazione di ogni *sweep* attraverso l'addestramento, per mezzo delle *feature* stesse, di un classificatore logistico;
3. Nella terza fase si procede a quantificare, rispetto a una funzione di merito, le prestazioni della BCI, anche in relazione ad altri sistemi.

### 2.1 Classificazione

Un classificatore è una funzione che assegna un'etichetta agli oggetti, intesi come reali o come loro rappresentazione; questa corrispondenza può essere ottenuta tramite una funzione prefissata, che può essere aggiornata nel corso della

classificazione, o tramite un algoritmo di apprendimento, che rientra nel campo del *machine learning* (apprendimento automatico).

Un classificatore con algoritmo di apprendimento, trova una classificazione sulla base di un *set* di esempi già etichettati, secondo il processo che viene detto metodo induttivo.

In termini più formali, il problema può essere espresso come segue: un classificatore è dato da un'opportuna funzione  $f: X \rightarrow Y$ , dove  $X \subseteq \mathbb{R}^n$ ,  $Y$  è un *set* di possibili etichette, tendenzialmente uno spazio finito e discreto (nel nostro caso, un problema di classificazione binaria, ovvero  $Y = \{-1, +1\}$  cui corrispondono rispettivamente a *target* e non *target*), ed è definita su  $X \times Y$  una distribuzione di probabilità condizionata  $p(\cdot, \cdot)$ , appresa tramite un *training set* di coppie  $\langle \mathbf{x}_i, y_i \rangle$  dove  $\mathbf{x}_i \in X, y_i \in Y$  con  $i=1 \dots N$ .

Spesso  $f$  è compresa in una famiglia  $F$  di classificatori, ovvero  $f(\mathbf{x}, \alpha)$ , dove  $\alpha$  è un parametro (o un vettore di parametri) che identifica un particolare classificatore in  $F$ . In tal modo, è possibile definire con maggiore accuratezza cosa significa “classificatore opportuno”, che risulta essere una  $f(\cdot, \alpha_0)$ , dove  $\alpha_0$  è il parametro o il vettore di parametri t.c. minimizzi la funzione dipendente dall'errore  $\varepsilon = |y - f(\mathbf{x}, \alpha)|$  (più in generale, da una funzione *loss*), detta funzione di rischio; per definizione, questa funzione è pari al valore atteso dell'errore:

$$\begin{aligned} R(\alpha) &= E[|y - f(\mathbf{x}, \alpha)|] = \int |y - f(\mathbf{x}, \alpha)| dp(\mathbf{x}, y) \\ &= \iint |y - f(\mathbf{x}, \alpha)| p(\mathbf{x}, y) d\mathbf{x} dy \end{aligned} \quad (1)$$

Questa formula dipende da  $\alpha$  ma anche dalla distribuzione  $p(\cdot, \cdot)$ , una distribuzione sconosciuta *a priori*: pertanto il valore atteso è tipicamente approssimato con una media empirica sul *training set*; l'equazione (1) diviene:

$$R_{emp}(\alpha) = \frac{1}{N} \sum_{i=1}^N |y_i - f(\mathbf{x}_i, \alpha)| \quad (2)$$

detta “rischio empirico”.

Poiché  $R_{emp}(\alpha) \rightarrow R(\alpha)$  al crescere di  $N$ , potrebbe apparire una scelta opportuna approssimare la  $R(\alpha)$  con la  $R_{emp}(\alpha)$ . Tuttavia, per un numero finito di  $N$ , la differenza tra i due fattori può essere significativa; questo problema è detto *overfitting*, e significa che il classificatore scelto  $f(\cdot, \alpha_0)$  ha delle performance molto buone sugli esempi del *training set*, ma scarse sui generici esempi  $\mathbf{x} \in X$ .

La capacità di generalizzazione di un classificatore, ovvero l'abilità del classificatore di estrarre l'informazione necessaria (e solo quella) dal *training set*, senza legarsi ai campioni presentati, è una condizione fondamentale, e per evitare l'*overfitting*, è necessario attuare particolari tecniche; nel nostro contesto, viene usata la cross-validazione a  $k$  parti (in ingl., *k-fold cross-validation*).

La cross-validazione è in genere un insieme di tecniche per stabilire se i risultati di un'analisi statistica siano generalizzabili al di là del *dataset* utilizzato; in particolare, nella cross-validazione a  $k$  parti, il *dataset* è diviso in  $k$  sotto-*set* mutuamente esclusivi di dimensioni approssimativamente uguali; uno dei  $k$  sotto-*set* viene utilizzato per la validazione, mentre gli altri  $k-1$  sotto-*set* vengono usati nell'addestramento; il processo viene quindi ripetuto  $k$  volte, ognuno con un *set* di validazione diverso; i  $k$  risultati ottenuti vengono quindi mediati per produrre una singola stima (64).

I vantaggi della cross-validazione a  $k$  parti sono che ogni osservazione è usata sia nell'addestramento sia nella validazione, e che ogni osservazione è usata nella validazione solo una volta; inoltre, è dimostrato che la varianza della stima si riduce al crescere di  $k$  (65).

## 2.1.1 Classificatore logistico

Per assegnare un attributo dicotomico a una o più variabili (statisticamente indipendenti, come le *sweep* delle nostre registrazioni EEG), una buona scelta è un classificatore basato sulla regressione logistica (66).

Nella regressione logistica, la risposta binaria  $y$  a un vettore di variabili indipendenti  $\mathbf{x}$ , è considerata come una variabile casuale binomiale (bernoulliana) di parametro  $p$  (66).

Il modello che ne deriva, detto modello logistico, si basa sulla funzione *logit*, che esprime il legame tra la risposta categorica binaria, il parametro  $p$  della distribuzione e il vettore aleatorio  $\mathbf{x}$ :

$$\text{logit} [p(\mathbf{x})] = \log \frac{p(\mathbf{x})}{1 - p(\mathbf{x})} \quad (3)$$

È possibile riscrivere la funzione (3) come un predittore lineare (67):

$$\text{logit} [p(\mathbf{x})] = \beta_0 + \beta_1 x_1 + \dots + \beta_h x_h \quad (4)$$

dove  $\beta$  è un vettore di coefficienti.

Combinando le funzioni (3) e (4), otteniamo:

$$p(\mathbf{x}) = \frac{\exp(\beta \mathbf{x})}{1 + \exp(\beta \mathbf{x})} \quad (5)$$

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

Un classificatore logistico approssima quindi la probabilità  $P(y|\mathbf{x})$  con una funzione logistica (68):

$$P(y = +1|\mathbf{x}) = \frac{1}{1 + \exp(\omega_0 + \sum_{j=1}^n \omega_j x_j)} \quad (6)$$

$$P(y = -1|\mathbf{x}) = 1 - P(y = +1|\mathbf{x}) = \frac{\exp(\omega_0 + \sum_{j=1}^n \omega_j x_j)}{1 + \exp(\omega_0 + \sum_{j=1}^n \omega_j x_j)}$$

dove con  $x_j$  si indicano le  $n$  componenti del vettore  $\mathbf{x}$ .

Per decidere a quale classe assegnare un dato campione  $\mathbf{x}$ , vengono confrontate le due probabilità  $P(y=+1|\mathbf{x})$  e  $P(y=-1|\mathbf{x})$ . Il vettore parametrico  $\omega$  viene trovato massimizzando il logaritmo della funzione di verosimiglianza:

$$L(\omega) = \sum_{i=1}^N \log P(y_i | \mathbf{x}_i, \omega) \quad (7)$$



Nella pratica non si massimizza la funzione (7) così definita, ma si tiene conto di un fattore di regolarizzazione, un termine di penalizzazione che evita valori troppo elevati di  $\omega$  in tal modo, viene data meno importanza alla direzione dello spazio di  $X$  che ha un potere discriminatorio maggiore, rendendo il classificatore più bilanciato su  $X$ .

$$L^{(\lambda)}(\omega) = \sum_{i=1}^N \log P(y_i | \mathbf{x}_i, \omega) - \lambda \|\omega\|^2 \quad (8)$$

dove  $\lambda$  è il fattore di penalizzazione.

Il ruolo della regolarizzazione è evitare la singolarità nell'inversione di  $\mathbf{X}^T \mathbf{X}$ ; nel caso specifico assume anche un ruolo diverso: un grande numero di parametri o un'alta correlazione tra loro può portare a una stima parametrica instabile (68); questo effetto può essere ridotto ponendo un limite al valore dei parametri, preferendo stime parametriche con minore varianza (66).

## 2.2 Computazione evolutiva

La computazione evolutiva è costituita da tecniche che traspongono in algoritmi d'ottimizzazione i principi dell'evoluzione naturale.

In un generico metodo per la ricerca della soluzione ottima, le possibili soluzioni a un problema possono essere totalmente note e disponibili, e quindi tali soluzioni possono essere esaminate una a una per selezionare quella ottima.

Questa ricerca esaustiva diventa ben presto impraticabile man mano che cresce la dimensione dello spazio delle soluzioni. La maggior parte degli algoritmi di ricerca campiona lo spazio delle soluzioni in modo casuale (ad esempio, nella Teoria del *random walk*) o in modo euristico (ad esempio, nella discesa del gradiente), una soluzione alla volta, nella speranza di ottenere la soluzione ottima o accontentandosi di una soluzione "buona" in tempi ragionevoli.

L'aspetto chiave che distingue gli algoritmi evolutivisti dagli algoritmi tradizionali è la presenza di una base di popolazione di possibili soluzioni al problema,

su cui l'algoritmo evolve generazione dopo generazione, eseguendo una ricerca della soluzione in modo efficiente (69).

La tecnica di calcolo evoluzionistico più popolare è l'algoritmo genetico, in cui soluzioni potenziali al problema vengono selezionate, valutate e combinate tra loro per ottenere (in teoria) migliori soluzioni potenziali.

## 2.2.1 Algoritmo genetico

Negli algoritmi genetici la rappresentazione della soluzione è solitamente una stringa di valori a lunghezza fissata, che rappresenta l'individuo nella popolazione. Ogni posizione nella stringa rappresenta una particolare caratteristica dell'individuo; di solito si suppone, seguendo in parte l'analogia biologica, che le stringhe siano una collezione di caratteristiche individuali, rappresentanti una particolare soluzione al problema, con una bassa se non alcuna interazione tra loro (69).

Gli elementi chiave in un GA sono gli individui e le popolazioni, che sono a loro volta una collezione di individui coinvolti nel processo di ricerca.

Esporremo di seguito, nel modo il più possibile esaustivo, le caratteristiche generali di ogni algoritmo genetico [**Errore. L'origine riferimento non è stata trovata.**], facendo particolare riferimento a quelle soluzioni distintive usate nel nostro approccio.



**Figura 9** - Rappresentazione schematica di un algoritmo genetico con le sue fasi fondamentali.

### 2.2.1.1 L'individuo

Un individuo rappresenta una singola soluzione al problema; esso porta con sé due rappresentazioni della soluzione:

1. I cromosomi, ovvero la soluzione sotto forma di bit o valori;
2. Il fenotipo, ovvero la soluzione secondo i termini e le forme richieste dal problema; lo spazio dei fenotipi è direttamente legato allo spazio delle soluzioni da una funzione di morfogenesi, che traduce in fenotipi le soluzioni espresse in cromosomi.

È auspicabile che l'espressione di un fenotipo possa avere un qualche significato all'interno del problema (forma d'onda o un particolare *range* di valori): ad esempio, nel nostro caso, una forma d'onda simile al segnale *target*; tuttavia, non è necessaria per l'efficacia dell'algoritmo l'espressione di fenotipi "realistici".

Un problema diverso è quando la funzione di morfogenesi non è iniettiva<sup>6</sup>, e in questo caso si dice che la rappresentazione è degenerata: sebbene sia tollerata una bassa degenerazione, tuttavia un'alta degenerazione significa che i cromosomi non esprimono specifiche caratteristiche della soluzione (69), quindi va cambiata la funzione di morfogenesi.

### 2.2.1.2 I geni

I geni sono la base nella costruzione di un GA, poiché rappresentano le caratteristiche specifiche di un individuo, per mezzo delle quali avviene la selezione delle soluzioni migliori. Un gene rappresenta in genere il valore di un singolo fattore, che solitamente deve rientrare in un *range* di controllo, necessario per far sì che l'algoritmo non esploda verso soluzioni inconsistenti.

Ogni gene deve essere espresso in un fenotipo tramite dei parametri fenotipici o in genere tramite delle funzioni morfogenetiche: si può dire che queste funzioni codificano le soluzioni in cromosomi e viceversa, i cromosomi in soluzioni; non è detto

---

<sup>6</sup> Una funzione si dice iniettiva se a un elemento del codominio corrisponde al massimo un elemento del dominio.

e non è necessario che queste funzioni siano biunivoche<sup>7</sup>, anche se devono essere almeno suriettiva<sup>8</sup>, in modo che tutto lo spazio delle soluzioni sia rappresentato (69).

### 2.2.1.3 La *fitness*

La funzione di merito, detta anche *fitness* di un individuo, valuta le soluzioni trovate dal GA rispetto ai suoi fenotipi, quindi per calcolarla è necessario decodificare il cromosoma.

### 2.2.1.4 La popolazione

Una popolazione è una collezione di individui. Gli aspetti più importanti e critici della popolazione come usata in GA sono:

1. La generazione iniziale della popolazione, quindi l'inizializzazione del GA;
2. La dimensione della popolazione.

Mentre il dimensionamento dipende dalla complessità del problema, non ci sono criteri per decidere quale inizializzazione usare: spesso, per la natura stessa dell'algoritmo, è sufficiente un'inizializzazione casuale, anche per rappresentare un *pool* genetico il più ampio possibile rispetto allo spazio delle soluzioni; altre volte, se i cromosomi sono in codifica binaria, si può optare per un'inizializzazione a zero o a uno; infine, può essere usata un'inizializzazione di tipo euristico, in modo da velocizzare la ricerca di soluzione ottima, anche se il rischio è una bassa diversità, con l'algoritmo confinato ad esplorare solo una piccola parte dello spazio delle soluzioni.

### 2.2.1.5 La codifica

La codifica è il modo di rappresentare i geni individuali e può avere diverse forme, in base al problema da risolvere. Le codifiche più usate sono:

1. La codifica binaria, secondo cui i cromosomi sono stringhe di bit, e ogni bit, o insieme di bit, rappresenta una caratteristica dell'individuo. Un'altra

---

<sup>7</sup> Una funzione si dice biunivoca quando ad ogni elemento del codominio corrisponde uno e un solo elemento del dominio.

<sup>8</sup> Una funzione si dice suriettiva quando ogni elemento del codominio è immagine di almeno un elemento del dominio.

possibilità è che l'intera stringa rappresenti un valore, e in questo caso la lunghezza della stringa è critica e dipende dall'accuratezza richiesta;

2. Codifiche ottale o esadecimale;
3. La codifica a permutazione, ovvero tramite una stringa numerica che rappresenta il valore in sequenza;
4. La codifica a valori, ovvero ogni cromosoma è una stringa di valori, non solo numerici ma anche simbolici o di caratteristiche descrittive del problema, e ogni valore è connesso in qualche modo al problema; altre volte, dove presenti numeri reali, può essere usata come codifica direttamente il valore usato nel problema, evitando una complicata codifica binaria;
5. La codifica ad albero, usata nell'evoluzione di programmi nella programmazione genetica, è caratterizzata da cromosomi costituiti da funzioni o comandi.

### 2.2.1.6 Il processo di riproduzione

Il processo di riproduzione è il cuore di ogni GA: è il processo tramite cui vengono creati individui nuovi ed eventualmente più adatti. Il processo è costituito da quattro fasi:

#### 1. Selezione

La selezione è il processo di scelta degli individui il cui corredo genetico deve essere ereditato, totalmente o in parte, dalla popolazione successiva.

L'idea alla base è far emergere i soggetti nella popolazione con più alta *fitness*, nell'ipotesi che la loro progenie abbia una *fitness* maggiore.

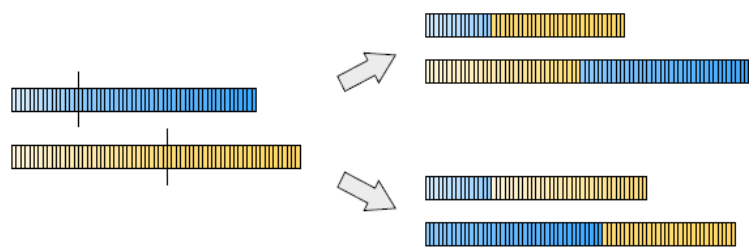
La pressione selettiva definisce quanto sono favoriti gli individui migliori nella selezione e determina il grado di convergenza del GA verso soluzioni ottime. Pressioni selettive troppo alte potrebbero tuttavia spingere prematuramente l'algoritmo verso soluzioni sub-ottimali, poiché mancherebbe la necessaria diversità alla popolazione per esplorare lo spazio delle soluzioni (69).

Si possono distinguere due tipi di schemi selettivi, uno proporzionale e uno relativo all'ordinamento; il primo è basato su valori di *fitness* individuali,

rispetto ai valori trovati nella popolazione, quindi la pressione selettiva risulta dipendente dalla particolare distribuzione di *fitness* e potrebbe essere necessaria una funzione di *scaling* per aumentare la probabilità di selezionare individui con più alta *fitness*; il secondo è basato sulla classifica (in ingl., *ranking*) degli individui, quindi la pressione selettiva non dipende dalla distribuzione della *fitness* nella popolazione.

## 2. Ricombinazione

Detta comunemente *crossover*, la ricombinazione è il processo tramite il quale da individui genitori vengono prodotti individui figli; mentre la riproduzione genera dei cloni di cromosomi con alta *fitness*, la ricombinazione genera nuovi individui nella speranza di creare una progenie migliore.



**Figura 10** - Rappresentazione della ricombinazione.

Esistono diversi metodi di ricombinazione, distinguibili a seconda del numero di individui genitori, del numero di siti d'incrocio, della presenza di maschere di ricombinazione, eccetera.

Il processo più semplice (usato anche nel seguente lavoro di tesi) è la ricombinazione a singolo punto. Esso procede secondo i seguenti passaggi: vengono scelti casualmente due individui tra quelli selezionati; viene scelto casualmente un sito d'incrocio lungo il cromosoma; vengono scambiati i valori tra le due stringhe rispetto al sito d'incrocio; si generano infine quattro nuovi individui [Figura 10], ognuno con una parte del corredo cromosomico degli individui genitori.

Se la lunghezza e l'ordinamento dei geni non è importante, possono essere utilizzati tutti e quattro i nuovi individui (anzi, risulta opportuno,

poiché aumenterebbe il mescolamento e la diversità della popolazione); se invece devono essere preservati l'ordine e/o la lunghezza dei cromosomi, possono essere utilizzati solo due dei quattro individui ottenuti.

Il parametro di base nelle tecniche di ricombinazione è la probabilità di ricombinazione,  $P_R$ . Al limite, questo parametro può valere 0%, e pertanto la nuova popolazione è fatta da individui cloni della popolazione precedente; o può valere 100%, e tutti gli individui nella nuova popolazione sono il risultato di ricombinazione; un valore intermedio permette una buona diversità, lasciando nella nuova generazione alcuni esemplari della vecchia, non sempre scelti tra gli individui a *fitness* massima, per aumentare la diversità nella popolazione.

### 3. Mutazione

La mutazione [Figura 11] gioca un ruolo fondamentale nel recupero di materiale genetico perso nella selezione e nella ricombinazione, prevenendo così che l'algoritmo si fermi in un minimo locale: mentre la ricombinazione è vista come il passaggio fondamentale nella ricerca di soluzioni migliori, la mutazione, nel suo ruolo di disturbo casuale, aiuta nell'esplorazione dell'intero spazio delle soluzioni, ovvero il processo di ricerca risulta ergodico<sup>9</sup>.



Figura 11 - Rappresentazione della mutazione.

Per ogni tipo di rappresentazione esistono diverse forme di mutazione; descriviamo brevemente i metodi usati nel seguente lavoro di tesi:

- a. per elementi discreti, consiste nel modificare con una bassissima probabilità il valore del gene, scegliendo una tra le possibili alternative;

<sup>9</sup> Un algoritmo si dice ergodico quando la probabilità di generare una qualunque soluzione è non nulla, ovvero per  $t \rightarrow \infty$  vengono esplorati tutti i punti di lavoro.

b. per una struttura dati a valori reali, la mutazione consiste nell'aggiungere una piccola perturbazione secondo una distribuzione Gaussiana: se il valore esce dall'intervallo ammissibile, la parte eccedente viene riportata all'inizio dell'intervallo (in modo analogo all'aritmetica modulare).

Il parametro di base nelle tecniche di mutazione è la probabilità di mutazione,  $P_M$ ; è un parametro che deve essere mantenuto basso, poiché renderebbe il GA un algoritmo di ricerca troppo casuale.

#### 4. Sostituzione

La sostituzione è l'ultimo passaggio di ogni ciclo di riproduzione: dal momento che la dimensione della popolazione è fissa, gli individui figli devono in qualche modo sostituire gli individui genitore, ed è necessario un metodo che determini quale membro della popolazione deve venir sostituito dalle nuove soluzioni. La tecnica usata può influenzare la convergenza stessa delle soluzioni verso l'ottimo.

Esistono due tecniche basilari: l'aggiornamento generazionale e l'aggiornamento in condizioni stazionarie.

Il concetto su cui si fonda l'aggiornamento generazionale consiste nel generare  $N$  individui figli da una popolazione di dimensione  $N$  per formare la nuova popolazione alla generazione successiva; chiamato anche aggiornamento  $(\lambda, \mu)$ , il processo parte da una popolazione di  $\mu$  individui che produce una progenie di  $\lambda$  individui, dove  $\lambda \geq \mu$ ; i  $\mu$  migliori individui della nuova generazione andranno a formare la nuova generazione.

Nell'aggiornamento a stato stazionario, dove stato stazionario indica una modifica (quasi-)infinitesima delle condizioni della popolazione, i nuovi individui sono inseriti nella popolazione non appena creati, andando a sostituire individui scelti tra i peggiori o tra i più vecchi; essendo quest'ultimo un metodo molto drastico, solitamente si segue un processo in cui i passaggi di selezione e sostituzione. Il metodo usato nella seguente tesi è il metodo *tournament selection*.

In questo metodo, ogni individuo di una nuova popolazione è selezionato in un "torneo" tra individui della generazione precedente: un numero fissato  $k$  di individui sono scelti a caso dalla vecchia popolazione, e l'individuo (o gli



individui) con la *fitness* più alta è dichiarato vincitore e passerà nella generazione successiva, di cui verrà generata la popolazione; i valori di  $k$  sono di solito piccoli, per evitare *overfitting* e mantenere un'adeguata diversità nella popolazione.

### 2.2.2. La funzione di *fitness*<sup>10</sup>

La selezione di modelli è il processo in cui si trovano e si modificano i parametri di un modello a seconda del problema di classificazione. Nella progettazione di sistemi di BCI, la selezione di modelli è un passaggio fondamentale: esso può includere la selezione e/o l'estrazione di un particolare tipo di caratteristica del segnale (un particolare canale EEG, un fenomeno neurologico, una banda di frequenza, i parametri del classificatore); ad esempio, nella *feature selection* per trovare il *set* ottimo di caratteristiche per un certo sistema di BCI, ne sono considerati diversi; per ogni *set*, viene calcolata la *performance* del sistema, e comparata alle *performance* degli altri *set*; il *set* con la *performance* migliore viene scelto.

Risulta fondamentale quindi definire una misura della prestazioni, ovvero un sistema per stabilire se un *set* è migliore degli altri, di quanto è migliore, e perché una determinata misura è migliore delle altre; inoltre, il calcolo della *performance* è effettuato in due situazioni diverse: durante la procedura di selezione per valutare la bontà delle soluzioni trovate dal sistema; nel confronto con le performance di altri sistemi;

La scelta della metrica di valutazione è di grande importanza in quanto dalla *fitness* dipende la procedura di ricerca della soluzione: un metrica di valutazione non corretta potrebbe guidare la selezione del modello lontano dall'ottimo o potrebbe indirizzare verso erronee conclusioni nel confronto tra due diversi sistemi di BCI.

La *performance* di un sistema di BCI a stati discreti è di solito riassunta nella matrice di confusione (70), contenente informazioni circa le classificazioni attese e le classificazioni ottenute dal sistema di classificazione; le *performance* sono valutate usando i dati nella matrice.

---

<sup>10</sup> È importante non confondere nel nostro contesto la *fitness* di un individuo durante l'esecuzione di un algoritmo genetico, con la *fitness* di un particolare sistema di BCI: benchè in questo lavoro abbiamo utilizzato lo stesso approccio per definirle, le due funzioni hanno ruoli e finalità differenti.

La matrice di confusione [Figura 12] fornisce un'informazione indirizzata alla valutazione della classificazione del sistema di BCI; tuttavia, non è facile confrontare diverse matrici di confusione: è necessario quindi un metro di valutazione, per riassumere il contenuto informativo della matrice di confusione, possibilmente in un solo valore.

		Classificazione predetta	
		Target	Non target
Classificazione reale	Target	TP	FN
	Non target	FP	TN

**Figura 12** –Rappresentazione di una matrice di confusione per 2 classi.

Per le classificazioni con *dataset* bilanciati<sup>11</sup>, la più comune metrica di valutazione è l'accuratezza totale di classificazione (71), mentre per classi non bilanciate, l'uso di questo criterio è totalmente insoddisfacente (72); recentemente, l'uso dell'accuratezza totale è stato messo in discussione anche in applicazioni con *dataset* bilanciati (73).

Alcune metriche usate nei sistemi di BCI sono le seguenti:

### 1. Accuratezza totale

L'accuratezza totale (Acc) (o ancor più comunemente il tasso d'errore,  $Err=1-Acc$ ) rappresenta il numero totale di esempi di test correttamente classificati. È stata spesso usata nella valutazione dei sistemi di BCI sincroni, ma il suo uso è molto limitato in sistemi asincroni o con *dataset* non bilanciati, a causa del peso dominante assegnato alla classe più frequente (solitamente, la classe non *target*); questo può portare ad interpretazioni errate circa le *performance* del sistema.

<sup>11</sup> Un *dataset* si dice bilanciato quando lo sono le sue classi, ovvero quando per ogni classe è presente lo stesso numero di esempi.

## 2. Information Transfer Rate (ITR)

Il tasso di trasferimento dell'informazione, derivato dalla Mutua Informazione, è stato proposto per valutare le *performance* di sistemi BCI sincroni (74); (75). La logica su cui si basa è la misura della quantità di informazione trasferita tra due punti. È considerato un buon metro di confronto tra diversi sistemi BCI, grazie al fatto di basarsi su un concetto generale di informazione e di *performance*, tuttavia, a causa della natura stessa della metrica, non può essere usato come unico metro di valutazione delle *performance* (76).

## 3. Kappa di Cohen

Il coefficiente Kappa di Cohen è una misura dell'accordo tra due stimatori (77). Dal momento che viene considerato come discriminante l'accordo dovuto al caso, è ritenuto una misura più robusta dell'accuratezza (71). I valori di Kappa variano nell'intervallo  $[-1 +1]$ : i valori negativi corrispondono a una *performance* peggiore di quella che si avrebbe con una classificazione casuale, mentre alti valori positivi indicano un accordo quasi perfetto tra i due classificatori. È anche usata misurare le *performance* di un classificatore rispetto alla classificazione attesa.

Questo coefficiente supera molte criticità della misura basata sull'accuratezza: tiene conto della distribuzione delle classificazioni errate, ovvero gli elementi fuori dalla diagonale nella matrice di confusione; le frequenze di occorrenza sono normalizzate, quindi classi con meno esempi hanno lo stesso peso di classi con molti esempi; può essere usato come confronto nel caso in cui i risultati di sistemi di BCI distinti abbiano differenze statisticamente significative (71).

## 4. ROC

Le curve ROC (dall'ingl., *Receiver Operating Characteristic*) sono un popolare metro di valutazione in presenza di classi non bilanciate; si basa sulla relazione tra la percentuale di veri positivi e la percentuale di falsi positivi; i criteri che si possono creare a partire dalla curva ROC, e usati come fitness, sono diversi (71); (78).

I maggiori problemi che possono emergere nell'uso della curva ROC sono una bassa specificità, dal momento che molti sistemi di BCI producono curve ROC simili alla perfetta curva ROC (79); e un elevato costo computazionale rispetto ad altri sistemi, dal momento che deve essere costruita una curva da cui stimare, a seconda dei casi, l'area sottesa o un valore di soglia (73).

Quando i metri di valutazione sono usati per selezionare un modello tra migliaia disponibili, nel calcolo della *fitness* a partire dalla matrice di confusione, diventa problematico il peso computazionale: per questo motivo, è auspicabile affidarsi da una misura facile da calcolare, che sintetizzi nel miglior modo possibile il contenuto informativo e le caratteristiche desiderabili del sistema di BCI.

## 5. TPR/FPR

Considerando due classi, *target* come positivi e non *target* come negativi, il rapporto tra la percentuale di veri positivi e la percentuale di falsi positivi è un altro metodo di valutazione recentemente proposto nell'ambito di sistemi di BCI a due classi (80). Questa metrica pesa molto i casi in cui si hanno basse percentuali di falsi positivi: al limite, se la percentuale di falsi positivi è nulla, il sistema non riesce a distinguere matrici di confusione con performance diverse.

## 6. Differenza HF

La differenza HF (dall'ingl., *Hit/False-difference*) è una nuova misura che tenta di sintetizzare l'informazione data dalla matrice di confusione (81); è definita come la differenza tra la percentuale di veri positivi (TP) e la percentuale di falsi positivi<sup>12</sup>.

Il vantaggio nell'uso della differenza HF è che risulta sensibile al rapporto tra falsi positivi e il totale di detezioni positive (71).

---

<sup>12</sup> La percentuale di false detezioni è pari ad  $1 - Pr$ , dove con  $Pr$  si indica la precisione.

## 7. Misura F

Strettamente legata alla differenza HF, ma maggiormente usata nel campo dell'*information retrieval* (dall'ingl., "recupero dell'informazione"), la misura F è la media armonica tra sensibilità (in questo contesto più comunemente detta *recall*) e precisione (Pr): il *recall* (Re) è il numero di classificazioni *target* corrette sul totale delle classificazioni *target* effettive, mentre la precisione è il numero delle classificazioni *target* corrette sul totale delle classificazioni *target* in uscita dal sistema. In formule, avremo:

$$\text{Pr} = \frac{\text{TP}_t}{\text{TP}_t + \text{FP}_t} \quad \text{Re} = \frac{\text{TP}_t}{\text{TP}_t + \text{FN}_t}$$

La versione tradizionale  $F_1$ , detta anche bilanciata, è la seguente:

$$F_1 = 2 \frac{\text{Pr} \cdot \text{Re}}{\text{Pr} + \text{Re}}$$

Mentre, volendo dare un peso diverso alle due componenti, si usa la formula corretta secondo un fattore non negativo  $\beta$ :

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{Pr} \cdot \text{Re}}{\beta^2 \cdot \text{Pr} + \text{Re}}$$

secondo la definizione, "F $_\beta$  misura l'efficacia (della classificazione) rispetto a un utente che dà un'importanza  $\beta$  volte maggiore al *recall* rispetto alla precisione" (82).

Si può esprimere la misura  $F_\beta$  anche in relazione ai due diversi tipi di errori possibili, ovvero i falsi positivi e i falsi negativi.

Nel tempo, la diversa connotazione che si è data alle due sorgenti di errore è stata universalmente accettata, divenendo noti come errori di tipo I (falsi positivi) ed errori di tipo II (falsi negativi)<sup>13</sup>.

---

<sup>13</sup> È famosa la similitudine dovuta a Neyman e Pearson, dell'errore di tipo I come della condanna di un innocente, e dell'errore di tipo II come dell'assoluzione di un colpevole.

È stato dimostrato quanto sia difficile il confronto tra diverse metriche di valutazione, e probabilmente non esiste una misura univoca che sia ideale per tutte le applicazioni (83): ognuno di questi metri di valutazione ha i propri punti deboli e i propri punti di forza, ed è probabilmente per questo motivo che non si è soliti motivare la scelta di un particolare metro di valutazione, se non legandola alle particolari esigenze dell'applicazione (71).

Ciò porta all'esigenza, espressa in diversi studi sul confronto tra i metodi di valutazione di *performance* in sistemi di BCI, di trovare un metro di valutazione universalmente riconosciuto (79).

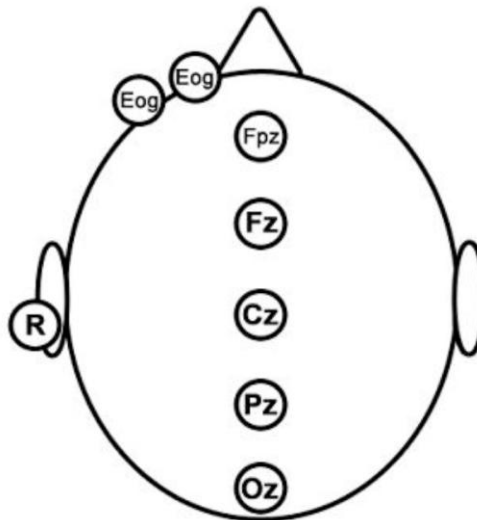
## 3. Protocollo sperimentale

---

Dodici soggetti sani (età media: 37 anni; *range*: 21-46 anni; 5 maschi e 7 femmine), hanno preso parte allo studio, approvato dal Comitato Etico dell’Ospedale IRCCS San Camillo di Venezia, dove sono avvenute le registrazioni.

Il segnale EEG è stato registrato con elettrodi (Ag/AgCl) del diametro di 10 mm, posizionati sulla cute mediante materiale adesivo e conduttivo nelle posizioni Fz, Cz, Pz e Oz del sistema 10-20 [Figura 133].

Il lobo auricolare sinistro è stato preso come riferimento mentre l’elettrodo di messa a terra è stato posto sulla fronte, Fpz [Figura 133].



**Figura 13** - Applicazione degli elettrodi per l’acquisizione dei segnali elettroencefalografico (EEG) ed elettrooculografico (EOG), secondo il Sistema Internazionale 10-20. Fz, Cz, Pz, Oz sono i canali EEG usati nel protocollo sperimentale dell’*équipe* del S. Camillo.

Gli elettrodi per la registrazione dell’elettrooculogramma (EOG) sono stati posizionati sulla palpebra inferiore e sul canto esterno di un solo occhio, normalmente

quello di sinistra. Il motivo dell'acquisizione del segnale EOG è quello di controllare ed eventualmente correggere interferenze nell'EEG dovute ad eccessivi movimenti oculari o delle palpebre (84).

L'amplificazione e il *pre-processing* del segnale sono stati realizzati da un filtro passa banda con frequenze di taglio rispettivamente a 0.15 Hz e a 30 Hz; il segnale è stato quindi digitalizzato con risoluzione a 16-bit, a una frequenza di acquisizione di 200 Hz.

Il segnale biologico è stato diviso in *sweep*, sincronizzate con l'occorrenza degli stimoli. Ogni singola *sweep* di registrazione inizia 500 ms prima dello stimolo e termina 1000 ms dopo lo stimolo, per una durata totale di 1500 ms.

## 3.1 Interfacce di stimolazione visiva

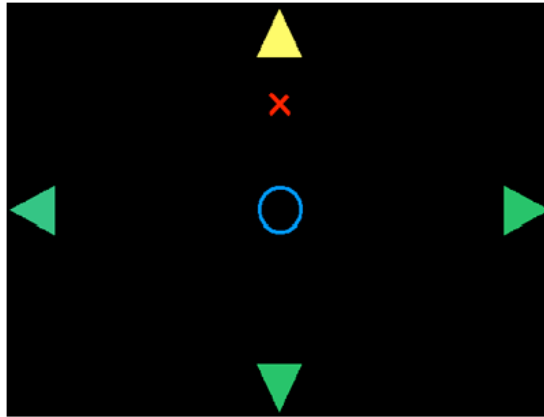
Ai partecipanti sono state presentate tre interfacce. Per tutte le interfacce si è preso spunto dallo stesso paradigma (85), dove i partecipanti controllavano il movimento di un cursore per raggiungere un *target*, a seguito di stimolazione nel campo visivo, in posizione periferica o centrale.

Le tre interfacce implementate possono essere intese come variazioni complesse del paradigma *oddball* classico: si diversificano per il tipo di icona o per modalità e tempo di stimolazione per elicitarne la P300.

La prima interfaccia, denominata "Arrows" [Figura 14], è quella più simile al paradigma originario: un simbolo a forma di croce, posto centralmente, costituisce il punto di fissazione per il soggetto; un simbolo circolare di colore blu funge da cursore e quattro frecce direzionali poste su ogni lato del monitor, a sfondo nero, che indicano le quattro direzioni cardinali (alto, basso, destra, sinistra); in ogni sessione una croce rossa indica la posizione della freccia direzionale *target*; i partecipanti devono mantenere fisso lo sguardo al punto di fissazione, evitando movimenti del capo; le quattro frecce cambiano colore alternativamente, passando dal verde al giallo, per un tempo di 150 ms, ognuna con la stessa probabilità di lampeggiare delle altre (25%); i partecipanti devono prestare attenzione al lampeggiare della freccia indicata dalla croce rossa (evento *target* con probabilità di accadimento pari al 25%) ed ignorare il lampeggiare delle altre frecce (evento non *target* con probabilità di accadimento pari al 75%); i paradigmi di

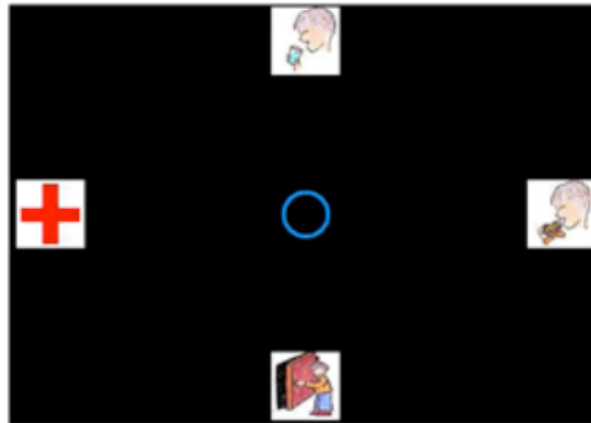


stimolazione delle altre due interfacce derivano da studi sull'orientamento dell'attenzione (36).



**Figura 14** - Interfaccia "Arrows"; possiamo notare il punto di fissazione centrale (il pallino blue) e la croce rossa che indica quale freccia rappresenta il segnale target.

La seconda interfaccia, denominata "Auto" [Figura 15], è costituita da un paradigma di stimolazione che elicitava l'orientamento esogeno dell'attenzione [1.5 Meccanismi attentivi]. L'interfaccia è composta dalla solita croce come punto di fissazione centrale, un cerchio blu che finge da cursore, sovrapposto inizialmente al punto di fissazione, e quattro icone, in luogo delle frecce direzionali, su sfondo nero; le icone sono quattro disegni in bianco e nero, selezionate tra otto disegni descrittivi di attività quotidiane (mangiare, bere, eccetera); i partecipanti devono mantenere fisso lo sguardo al punto di fissazione, evitando il più possibile movimenti del capo; le quattro icone scompaiono per un breve lasso di tempo (75 ms) con la stessa probabilità (25%), e ricompaiono nella stessa posizione; i partecipanti devono prestare attenzione all'icona *target* (evento *target* con probabilità di accadimento pari al 25%), precedentemente indicata da uno sperimentatore, ed ignorare le altre (evento *non-target* con probabilità di accadimento pari al 75%).



**Figura 15** - *Interfaccia "Auto"; possiamo notare il punto di fissazione centrale (il pallino blue) e la croce rossa che sostituisce l'immagine che rappresenta il segnale target; al posto delle frecce sono presenti immagini di attività quotidiane.*

La terza interfaccia, denominata “Vol” [Figura 16], viene usata seguendo un paradigma di stimolazione che rientra nell’orientamento endogeno dell’attenzione [1.5 Meccanismi attentivi]. Il punto di fissazione, il cursore e le quattro icone sono gli stessi usati nell’interfaccia “Auto”; tuttavia, in questo caso lo stimolo è una delle parole che nella lingua Italiana indicano le direzioni cardinali in cui sono poste le icone (“alto”, “basso”, “sinistra”, “destra”); le quattro parole sono presentate casualmente, con la stessa probabilità (25%), in corrispondenza del punto di fissazione, per un tempo di 900 ms, mentre le quattro icone restano sempre a schermo; ai partecipanti viene richiesto di prestare attenzione alla parola che indica la posizione relativa dell’icona *target* (evento *target* con probabilità di accadimento pari al 25%), precedentemente indicata da uno sperimentatore, ed ignorare le altre (evento non *target* con probabilità di accadimento pari al 75%).



**Figura 16** - Interfaccia "Vol"; possiamo notare il punto di fissazione centrale (il pallino blue), il cue anticipatorio, in questo caso la parola "sinistra", e la croce rossa che sostituisce l'immagine che rappresenta il segnale target; al posto delle frecce sono presenti immagini di attività quotidiane.

### 3.1.1 Ipotesi sperimentale

L'obiettivo dell'uso di queste tre diverse interfacce è quello di sperimentare un nuovo tipo di interfaccia per una nuova BCI basata sulla P300.

L' interfaccia "Arrows" è solo un adattamento di quella utilizzata nel paradigma sperimentale introdotto: le quattro icone poste ai lati dello schermo sono costituite da frecce indicanti le direzioni cardinali (alto, basso, destra, sinistra).

Le due interfacce completamente nuove sono state pensate a partire dagli studi sull'attenzione volontaria e automatica (36); la prima notevole differenza rispetto alla prima interfaccia è che nelle due nuove interfacce, chiamate rispettivamente "Auto" e "Vol", si è passati dall'uso di quattro frecce all'utilizzo di icone quadrate con delle raffigurazioni concettuali, le quali sono rappresentazioni di ipotetici bisogni essenziali o di richieste da parte di un paziente, sempre poste ai quattro lati dello schermo; la differenza, invece, tra le due nuove interfacce riguarda la modalità di orientamento dell'attenzione.

Nell'interfaccia "Auto" l'orientamento dell'attenzione è evocato dal *blinking* delle icone; la posizione periferica del *cue* e la sua natura (il *blinking*) possono evocare la cattura automatica dell'attenzione, secondo la condizione sperimentale (36) in cui è stato studiato l'orientamento esogeno dell'attenzione; l'interfaccia "Vol" è del tutto simile all'interfaccia "Auto", con la differenza che cambia la modalità di stimolazione:

le icone in questo caso non sono soggette al *blinking* ma restano sempre fisse sullo schermo.

Diversamente dall'interfaccia precedente, questa è ispirata alla condizione sperimentale con cui è stato studiato l'orientamento endogeno volontario dell'attenzione (36). Uno degli scopi è il confronto delle prestazioni delle tre diverse interfacce, tentando di stabilire quale di queste risulti più efficace; nello specifico si cerca di verificare se le classificazioni dei segnali EEG provenienti da esperimenti con un'interfaccia manifestino *performance* migliori rispetto alle altre interfacce.

In particolare, il nostro studio ha assunto nel seguente contesto i connotati di uno studio in cieco semplice: *a priori* non ci è stata fornita nessuna informazione riguardo la natura e le *performance* attese delle interfacce.

## 3.2 Costruzione del *dataset*

Per coprire la distanza tra la posizione centrale del cursore e l'icona prescelta sono necessari dai quattro ai sedici riconoscimenti della P300 da parte del sistema, a seconda della sessione cui partecipa il soggetto; durante la presentazione degli stimoli è stato registrato il segnale EEG utilizzato successivamente per l'addestramento *off-line* del classificatore.

Per ogni interfaccia i partecipanti hanno eseguito otto sessioni di *learning* (LS), sedici sessioni di *testing* (TS), divise in quattro sedute (le prime tre vanno a accrescere il *training set* del classificatore, l'ultima seduta, detta *last testing* (LTS), costituisce uno dei due *testing set* del classificatore), quattro sessioni di *follow-up* (FS).

Sono state eseguite due analisi distinte, una in cui il *testing set* del classificatore è la sessione *last testing*, una in cui il *testing set* del classificatore è costituito dalle le sessioni di *follow-up*: ciò in modo da testare la robustezza e la capacità di generalizzazione dei classificatori.

Il nostro sistema di BCI utilizza gli stessi dati per l'addestramento e il *testing* del classificatore, in modo da avere delle *performance* confrontabili con il classificatore nativo dell'esperimento.

Le sessioni di *learning* erano caratterizzate da un *feedback* perfetto<sup>14</sup>, ovvero ogni volta che lo stimolo *target* si verificava, il movimento atteso era eseguito indipendentemente dal fatto che fosse stata rilevata la presenza di P300; di contro, nelle sessioni di *testing* e *follow-up* il cursore si muove verso il *target* solo se viene classificata una P300 come risposta cerebrale, ovvero si usa un *feedback* reale.

Nei *dataset* non sono state selezionate *sweep* e non c'è stata nessuna operazione di *processing* aggiuntivo, sia per non avvantaggiare il nostro sistema rispetto al sistema ICA, sia per aumentare la variabilità della ricerca di soluzioni grazie alla rumorosità del segnale; inoltre, per completezza, sono stati costruiti due tipi di *dataset*: uno faceva riferimento alle *sweep* utilizzate dal sistema utilizzato dall'équipe del S.Camillo nell'analisi e nella creazione del proprio classificatore<sup>15</sup>; una faceva riferimento a tutte le *sweep* delle fasi di *training* e di *testing*.

Di seguito sono riportati i *dataset* utilizzati nell'analisi, con i rispettivi *tag*:

1. Analisi G1: classificazioni il cui *training set* è stato definito secondo i *pattern* di dati del sistema ICA tra le sessioni LS e TS, e come *testing set* LTS;
2. Analisi G2: classificazioni il cui *training set* è stato definito secondo i *pattern* di dati del sistema ICA tra le sessioni LS e TS, e come *testing set* FS;
3. Analisi T2: classificazioni il cui *training set* è costituito da tutte le *sweep* di LS e TS, e come *testing set* LTS;
4. Analisi T3: classificazioni il cui *training set* è costituito da tutte le *sweep* di LS e TS, e come *testing set* FS.

---

<sup>14</sup> Con “*feedback* perfetto”, s'intende il verificarsi della conseguenza di una corretta classificazione della P300 a seguito dell'orientamento dell'attenzione da parte del soggetto, a prescindere dal riconoscimento della P300 da parte del sistema.

<sup>15</sup> Nel sistema del S. Camillo sono stati previsti diversi *pattern* di dati, un numero arbitrario di maschere booleane per la selezione di specifiche *sweep* nei *set* di *training* e di *testing*, in modo da selezionare in base alle *performance*, i migliori *pattern* di dati; benchè sia una scelta non condivisa, per rendere confrontabili i nostri sistemi, abbiamo deciso di eseguire un'analisi anche alle condizioni sopra descritte.

## 3.3 Protocollo di analisi

Un sistema di BCI deve decidere se un segnale registrato appartenga a una (in generale anche a più di una) classe; il sistema di BCI usato in questo lavoro di tesi può essere definito come:

1. indipendente, poiché né l'*input*, né l'analisi né l'*output* dipendono da un segnale correlato con la via muscolare periferica;
2. sincrono, poiché l'*onset* all'evento è dato dal sistema e non dall'utente; non invasivo, poiché si tratta di registrazioni EEG;

Attualmente, il sistema lavora *off-line*, ovvero l'addestramento e la classificazione dello stesso vengono effettuate in una fase successiva all'esperimento; l'obiettivo futuro è però quello di riuscire ad ottenere un sistema di BCI *on-line*, il più possibile affidabile e in tempo reale.

La possibilità di classificare epoche in modo efficace senza ricorrere alla canonica successione di *pre-processing*, *processing*, estrazione di caratteristiche e classificazione, è stata introdotta recentemente nel campo delle BCI (58); (87).

Il nostro approccio si basa sul concetto di estrazione automatica di *feature*, implementata da un algoritmo genetico: le *feature* sono codificate in geni di cromosomi di lunghezza variabile; la *fitness* di un individuo è data dalla *performance* del classificatore addestrato con i geni decodificati dell'individuo.

Non è la prima volta che gli algoritmi genetici sono usati in sistemi di BCI (88); nel nostro approccio, vengono usati nella creazione e selezione delle *feature*, passate poi al classificatore, che viene addestrato nel modo classico.

### 3.3.1 Ampiezza della finestra

La finestra temporale di riferimento comprende 200 ms precedenti allo stimolo, fino agli 800 ms successivi, per un intervallo totale di 1 s.

Per includere il *rebound*, abbiamo aumentato la finestra temporale, includendo anche gli ultimi 200 ms di registrazione, in modo da confrontare *performance* tra analisi diverse del nostro sistema.

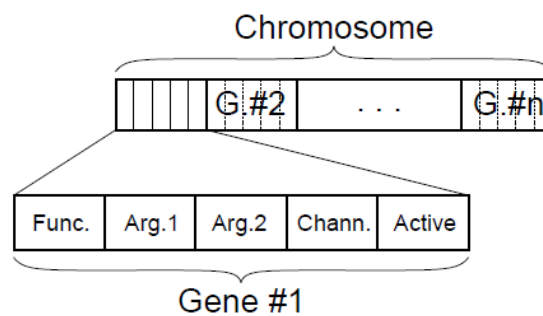
Pertanto, tutte le analisi precedentemente descritte, sono state effettuate “con *rebound*” e “senza *rebound*”, per un totale di 20 analisi ( $4 \text{ dataset} \times 2 \text{ finestre temporali}$

per le interfacce “Arrows” e “Auto”; 2 *dataset* × 2 finestre temporali per l’interfaccia “Vol”).

### 3.3.2 Struttura dell’algoritmo genetico

Ogni individuo nella popolazione è costituito da un solo cromosoma; ogni cromosoma contiene un numero variabile di geni, tutti di struttura identica, formati da cinque elementi, con diverse codifiche; i primi tre elementi definiscono la *feature* [Figura 17]:

1. Un numero intero, che designa una funzione d’estrazione di *feature* in un *set* predefinito;
2. Due parametri a valore reale, compresi nell’intervallo  $[0, +1)$ , per l’estrattore selezionato.



**Figura 17** - Struttura di un cromosoma e di un gene per il nostro sistema.

Gli estrattori di *feature* sono il primo passaggio dell’espressione del fenotipo, ovvero la particolare soluzione rappresentata nella codifica cromosomica; essi sono funzioni reali a tre argomenti, che designano univocamente il tipo di estrattore da utilizzare e la sua forma d’onda.

Il quarto elemento è un numero intero, identificativo di uno dei canali EEG da cui estrarre la *feature* codificata nel gene; l’ultimo elemento di un gene è un *flag* booleano che definisce se un gene è attivo o inattivo: i geni inattivi non sono usati nel calcolo della *fitness*, ma hanno il ruolo di riserva genetica, dato che possono essere attivati nella mutazione. La posizione di un gene nel cromosoma non è significativa, così come non è stato posto nessun limite alla lunghezza dei cromosomi.

### 3.3.2.1 Estrattore di *feature*

Gli estrattori di *feature* già implementati dal *tool*, sono stati sostituiti da un solo estrattore di *feature*, basato su una funzione Gaussiana, dimostrandone l'efficacia in termini di *performance*.

L'idea alla base dell'uso di una funzione Gaussiana è la possibilità di descrivere adeguatamente la forma d'onda del segnale per mezzo della somma di funzioni Gaussiane, i cui rispettivi fattori di forma  $b$ , e i rispettivi valori centrali  $a$ , vengono selezionati durante l'esecuzione dell'algoritmo genetico: a questi parametri fanno riferimento gli elementi a valore reale descritti nel Paragrafo precedente.

$$f(x) = e^{-\frac{(x-a)^2}{2b^2}}$$

Per limitare la scelta di Gaussiane, sono state selezionate delle funzioni Gaussiane massime e minime, funzioni corrispondenti ai punti di massimo e di minimo dei parametri codificati nei geni:

1. la funzione Gaussiana massima è stata individuata in quella che potesse abbracciare tutto il segnale; pertanto l'intervallo temporale è stato fatto coincidere con l'intervallo  $[-2b, +2b]$ , dove  $b$  è il parametro della funzione Gaussiana legato all'ampiezza della funzione<sup>16</sup>;
2. la funzione Gaussiana minima è stata individuata in quella che potesse abbracciare almeno sei campioni, i quali ci sono sembrati il giusto compromesso per rappresentare con adeguata accuratezza le caratteristiche del segnale senza che l'algoritmo genetico evolvesse verso fenotipi non "realistici";
3. un fattore moltiplicativo dipendente dal numero di campioni è stato inserito per tener conto della variabilità dell'intervallo di tempo considerato;
4. l'ampiezza della funzione Gaussiana è stata mantenuta a valore unitario.

Lo schema di base è stato tuttavia mantenuto inalterato: il segnale di *input*, ovvero il segnale EEG, è moltiplicato da una funzione peso (l'estrattore di *feature*), e il

---

<sup>16</sup> Più correttamente, il parametro legato alla larghezza di una funzione è il FWHM (dall'ingl., *Full Width at Half Maximum*); per la funzione Gaussiana, questo parametro è legato a  $b$  dall'espressione:  $\text{FWHM} = 2\sqrt{2\ln 2} b \approx 2.3548 b$



risultato è integrato sul tempo: in altre parole, si calcola la cross-correlazione a ritardo nullo tra la funzione peso e il segnale. Se chiamiamo  $s(\cdot)$  il segnale EEG e  $u(\cdot)$  l'estrattore di *feature*, la *feature*  $x$  risulta:

$$x = \sum_{t=1}^T u(t) \cdot s(t)$$

### 3.3.2.2 Fitness, Selezione, Mutazione e *Crossover*

La *fitness* di un cromosoma è determinata misurando la *performance* di un classificatore logistico addestrato sulle *feature* che il cromosoma codifica. Per avere una stima corretta della *performance*, viene applicato uno schema di cross-validazione a quattro parti, e la media sulle quattro ripetizioni viene usata come *fitness*.

Il criterio per valutare gli individui durante l'esecuzione dell'algoritmo genetico è una combinazione tra precisione e *recall* per i segnali *target*

$$f = \frac{2}{3} p_T + \frac{1}{3} r_T$$

dove abbiamo:

$$p_T = \frac{TP_T}{TP_T + FP_T} \quad r_T = \frac{TP_T}{TP_T + FN_T}$$

Il motivo di questa scelta è che, mentre è vantaggioso riconoscere un gran numero di segnali *target*, caratteristica rappresentata dal *recall*, tuttavia è molto più dannoso un falso positivo di un falso negativo, pertanto la precisione è stata pesata il doppio rispetto al *recall*.

In fase di confronto, per evitare di favorire il nostro sistema rispetto al sistema usato al S. Camillo, abbiamo usato come *fitness* del sistema la misura F bilanciata.

In un algoritmo genetico, la *fitness* è usata per selezionare gli individui più promettenti per la generazione successiva; il meccanismo di selezione da noi usato è il *tournament selection* con elitismo, un *set-up* standard negli algoritmi genetici.

L'elitismo è la pratica di mantenere nel *pool* genetico gli individui con il maggior *fitting*, anche quando la selezione li esclude, o la mutazione e il *crossover* li modifica.

Nel nostro lavoro abbiamo testato diverse dimensioni del *tournament*, selezionando infine una dimensione del *tournament* di 10 cromosomi e un elitismo pari a 2 individui per ogni generazione.

Dopo la selezione, gli individui sono sottoposti a *crossover* e mutazione. Questi due operatori sono stati un po' modificati per adattarli alla particolare struttura cromosomica usata.

Il *crossover* è applicato a paia di cromosomi tra gli individui selezionati, scelti a caso, con una probabilità di 0.7; entrambi i cromosomi sono divisi in due segmenti sul limite di un gene, in modo casuale, e i quattro segmenti risultanti sono ricombinati; poiché l'ordine dei geni in un cromosoma non è importante, un segmento da un cromosoma può essere accoppiato con entrambi i segmenti dell'altro, così da avere due possibili accoppiamenti: quale dei due venga utilizzato è scelto in modo casuale, ma l'importante è che possano essere scelti entrambi in modo da che aumenti il mescolamento di materiale genetico; il *crossover* può essere applicato a individui con un comune progenitore, e in tal modo possono condividere parte del *pool* genetico; per rafforzare le *feature*, è opportuno che i cromosomi contengano geni duplicati accumulati nel tempo: in tal modo le *feature* si conservano nel processo di evoluzione; i geni duplicati tuttavia sono ignorati in fase di valutazione della *fitness*, per favorire l'emergere di nuove *feature*.

La mutazione opera sugli elementi dei geni; ogni elemento in ogni gene muta con probabilità 0.005; gli elementi sono modificati in modo diverso a seconda del loro tipo: gli elementi discreti sono modificati scegliendo casualmente una delle possibili alternative, mentre agli elementi continui viene aggiunta una perturbazione secondo una distribuzione Gaussiana; se il risultato esce dall'intervallo ammissibile, la differenza viene riportata all'inizio dell'intervallo.

### 3.3.2.3 Dimensione della popolazione e Criterio d'arresto

La dimensione della popolazione è costante durante l'esecuzione dell'algoritmo genetico; per gli esperimenti abbiamo usato una popolazione di 120 individui;

l'inizializzazione è totalmente casuale; la lunghezza iniziale dei cromosomi è estratta da una distribuzione geometrica con media 20 geni.

L'ultima componente per completare la descrizione del nostro algoritmo genetico è il criterio d'arresto; facendo tesoro delle informazioni raccolte da precedenti lavori (89), abbiamo deciso di prolungare di poco il numero di generazioni, inizialmente di 15, portandolo a 20, mentre per il criterio d'arresto, abbiamo considerato la crescita della *fitness* come discriminante: se il valore massimo e la *fitness* media sono costanti per almeno 4 generazioni, l'evoluzione è considerata terminata (89).

### 3.3.2.4 Validazione del *Set* di *Feature*

Alla fine di ogni esecuzione dell'algoritmo genetico, le *performance* di individui con alta *fitness* sono validate su un *set* di test, nuovo rispetto all'algoritmo genetico; gli individui scelti hanno una *fitness* pari ad almeno il 99% della *fitness* massima dell'ultima generazione; valutare più di un individuo e non solo il migliore costituisce una valutazione più robusta dell'efficacia del metodo.

Per ogni individuo selezionato, vengono estratte le *feature* codificate nei geni da tutte le *sweep* del *set* di addestramento, con cui viene addestrato un classificatore logistico. Le stesse *feature* sono estratte da un *set* di test, e vengono così valutate le *performance* del classificatore.

L'algoritmo genetico ha un grosso peso computazionale, poiché la lunghezza dei cromosomi cresce di generazione in generazione; di contro l'estrattore di *feature* e il classificatore logistico addestrato sono molto rapidi, e possono pertanto essere usati in applicazioni *on-line*.

### 3.3.2.5 *Template*

L'algoritmo genetico ottimizza le *feature* in relazione al classificatore usato: in tal modo, le *feature* sono strettamente connesse al classificatore.

Chiamiamo  $s(\cdot)$  il segnale EEG per un singolo canale di un'epoca, e consideriamo solo le *feature* generate dall'algoritmo genetico per questo canale. Per l'estrattore di *feature* descritto precedentemente, la *feature*  $x_j$  è ottenuta dalla cross-correlazione a ritardo nullo del segnale con la funzione peso  $u_j(\cdot)$ :

$$x_j = \sum_{t=1}^T u_j(t)s(t) \quad (9)$$

dove  $j$  indica che  $x_j$  è la *feature* codificata nel  $j$ -esimo gene, e  $T$  è il numero di campioni per ogni epoca. Si noti che  $u_j(\cdot)$  non è una generica funzione, ma la precisa funzione codificata da tutti gli elementi contenuti in un gene; un classificatore logistico addestrato stima la probabilità che il segnale  $s(\cdot)$  appartenga a una classe, secondo l'equazione:

$$P(y = +1|\mathbf{x}) = \frac{1}{1 + \exp(\omega_0 + \sum_{j=1}^n \omega_j x_j)} \quad (10)$$

dove in questo caso  $x_j$  è dato dall'estrattore di *feature*; rappresenta la probabilità per la classe *target*, mentre per la classe non *target* la probabilità è  $1 - P(y = +1|\mathbf{x})$ . Riscriviamo l'argomento dell'esponenziale esplicitando tutti i fattori:

$$\begin{aligned} \omega_0 + \sum_{j=1}^n \omega_j x_j &= \omega_0 + \sum_{j=1}^n \omega_j \left( \sum_{t=1}^T u_j(t)s(t) \right) \\ &= \omega_0 + \sum_{t=1}^T \left( \sum_{j=1}^n \omega_j u_j(t) \right) s(t) \end{aligned} \quad (11)$$

Il termine

$$v(t) = \sum_{j=1}^n \omega_j u_j(t) \quad (12)$$

dipende solo dal *set* di *feature* e dal classificatore, quindi è lo stesso per ogni epoca del *dataset*, relativa a un addestramento del classificatore, quindi a una generazione. L'equazione diviene:

$$P(y = +1 | s(\cdot)) = \frac{1}{1 + \exp(\omega_0 + \sum_{t=1}^T v(t)s(t))} \quad (13)$$

che in notazione vettoriale diviene

$$P(y = +1 | s(\cdot)) = \frac{1}{1 + \exp(\omega_0 + \langle \mathbf{v}, \mathbf{s} \rangle)} \quad (14)$$

Il vettore  $\mathbf{v}$  è un *template*, uscita finale dell'algoritmo genetico, mentre il prodotto scalare tra  $\mathbf{v}$  e  $\mathbf{s}$  è un'altra correlazione: la classificazione è basata sulla somiglianza dei segnali dell'epoca considerata con i *template*  $\mathbf{v}$  espressi da ogni canale EEG.

Da qui l'utilizzo della funzione Gaussiana in luogo delle funzioni native del *tool*: poiché la reale uscita dell'algoritmo genetico sono i *template*, abbiamo ritenuto opportuno che essi avessero una forma d'onda simile a quella fisiologica dei segnali da classificare.

### 3.3.2.6 Metrica di valutazione del sistema

Sono stati selezionati i 5 migliori cromosomi in uscita ad ogni esecuzione dell'algoritmo genetico in base al Kappa di Cohen, calcolato tra la classificazione reale e la classificazione ottenuta dal nostro sistema; la scelta dell'uso di questa misura dipende dal fatto che il sistema di BCI è basato, come abbiamo visto, su una funzione di merito molto particolare: per non avere alcun vantaggio rispetto al sistema ICA, abbiamo scelto il Kappa di Cohen tra le altre possibili alternative in uscita al nostro classificatore (accuratezza, mutua informazione, precisione, *recall* ed appunto Kappa di Cohen) al posto della funzione *fitness* usata dal sistema; la scelta di questa misura deriva dal fatto di essere una misura di *performance* generica e robusta (71).

Per il confronto con il sistema ICA abbiamo invece utilizzato la misura F bilanciata, una misura generica di prestazioni di un classificatore, basata sulla matrice di confusione; la scelta è stata dettata da diversi fattori:

1. L'impossibilità di usare la *fitness* usata come funzione di merito dall'algoritmo genetico del nostro sistema, per non avvantaggiarlo rispetto al sistema ICA;
2. L'utilità di combinare in un'unica misura, veloce da calcolare e definita in modo chiaro e semplice, la precisione e il *recall* su segnali *target*, i fattori che quantificano la bontà di una classificazione.

3. Sono stati esclusi l'accuratezza, poiché le classi non sono bilanciate; la Kappa di Cohen, poiché misura l'accordo tra due classificatori, ed è quindi una misura relativa; la mutua informazione, poiché è una misura generica legata all'efficienza del trasferimento d'informazione e non alla sua qualità.

## 4. Risultati

---

Per una migliore lettura, i risultati sono stati divisi per interfaccia e per tipo di finestra temporale (senza *rebound* e con *rebound*); la misura di prestazioni utilizzata è la F-bilanciata, per non avvantaggiare il nostro sistema rispetto al sistema del S. Camillo.

Si farà riferimento al sistema usato dall'équipe del S. Camillo come a sistema ICA, in quanto la procedura del sistema di BCI dell'équipe del S. Camillo è basata sulla decomposizione ICA (49); (85).

Il confronto è basato sulla media delle *performance* dei 5 cromosomi migliori e della *performance* del sistema ICA; in aggiunta vengono presentate anche le *performance* ottenute scegliendo solo il cromosoma migliore.

Per ogni interfaccia sono stati eseguiti i seguenti confronti:

1. Confronto tra il nostro sistema e il sistema ICA;
2. Confronto all'interno dei risultati del nostro sistema, tra le *performance* in presenza o in assenza di *rebound*;
3. Confronto rispetto a diversi *set* di *testing* (*last testing* vs *follow-up*), per testare la robustezza del sistema.

Per semplicità di lettura il nostro sistema e il sistema ICA sono stati denominati nei grafici e nelle tabelle rispettivamente come GA ed ICA.

Abbiamo eseguito infine un'analisi statistica tra le diverse interfacce, per confrontare le diverse *performance* e stabilire per quale interfaccia i sistemi di classificazione abbiano le prestazioni migliori.

Poiché le *performance* del sistema ICA sono relative solo ai *pattern* di dati con cui sono stati costruiti i *dataset* G1 e G2, anche per i *dataset* T2 e T3 si usano gli stessi risultati; la scelta di presentare comunque i risultati deriva dal fatto che scegliere quali dati includere nell'analisi potrebbe aver avvantaggiato il sistema ICA: abbiamo pertanto confrontato le prestazioni relative ai diversi *dataset*, arrivando alla conclusione che, almeno per quanto riguarda il nostro sistema, la scelta di includere una *sweep* piuttosto

che un'altra non cambia le prestazioni del sistema, anzi potrebbe causare una perdita di diversità, quindi un peggioramento generale nella ricerca dell'ottimo da parte dell'algoritmo genetico, eventualità che comunque non sembra essersi verificata.

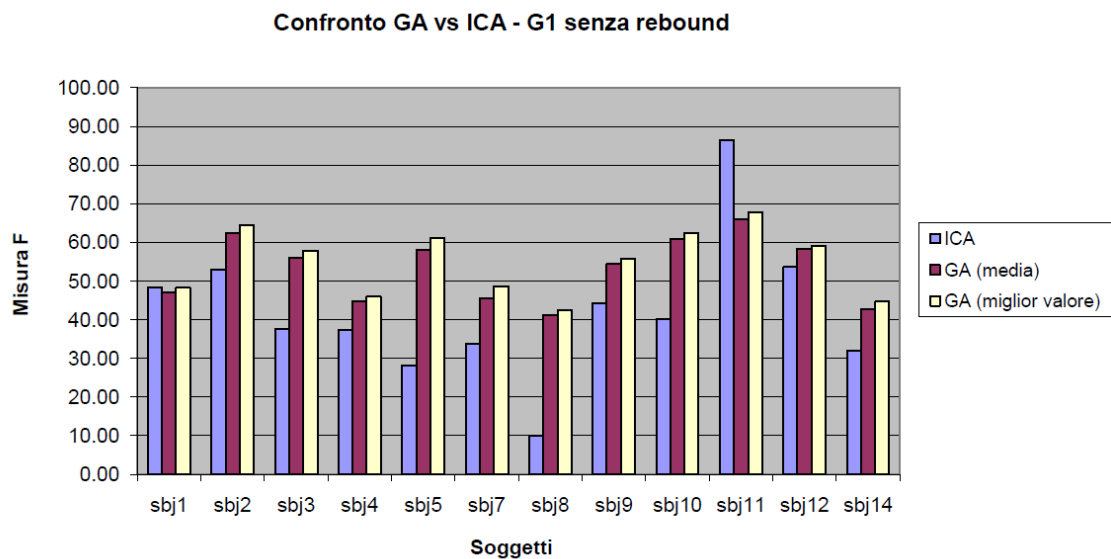


## 4.1 Interfaccia “Arrows”

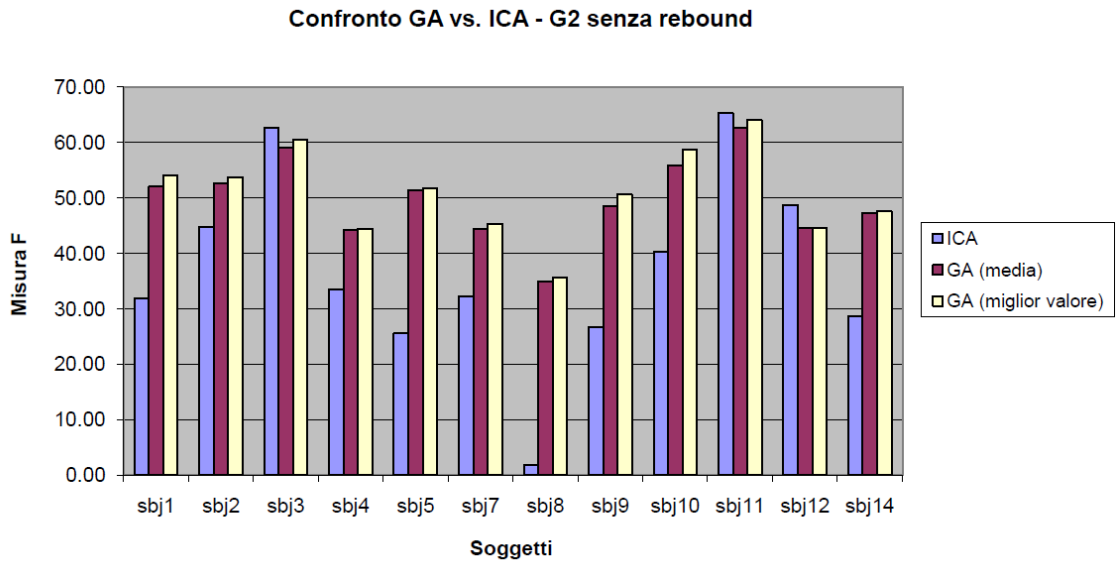
### 4.1.1 GA vs ICA

Confrontiamo le *performance* del nostro sistema e le *performance* del sistema ICA per l’interfaccia “Arrows”.

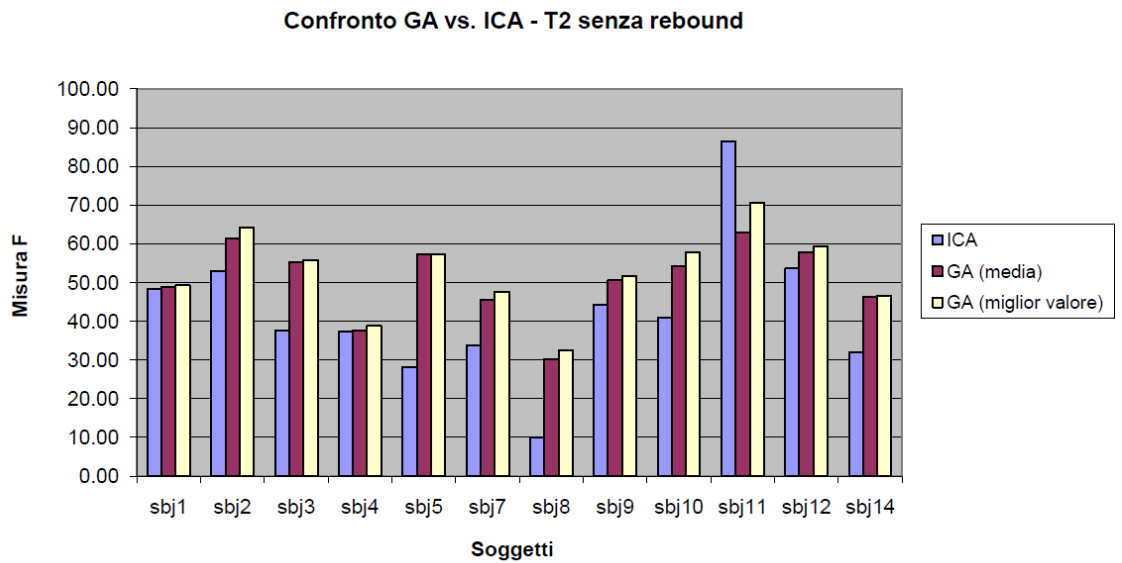
Nei grafici presentati di seguito [Figura 18Figura 19Figura 20Figura 21], si mettono a confronto per ogni *dataset* il sistema ICA (in colore viola) con il nostro sistema, rappresentato sia come valor medio dei 5 cromosomi migliori (in colore fucsia), sia come prestazione del cromosoma migliore (in colore giallo). Come ascissa abbiamo la misura F-bilanciata come definita nel Capitolo 2; come ordinata abbiamo i soggetti, denominati con la stessa sigla usata da parte dell’*équipe* del S. Camillo.



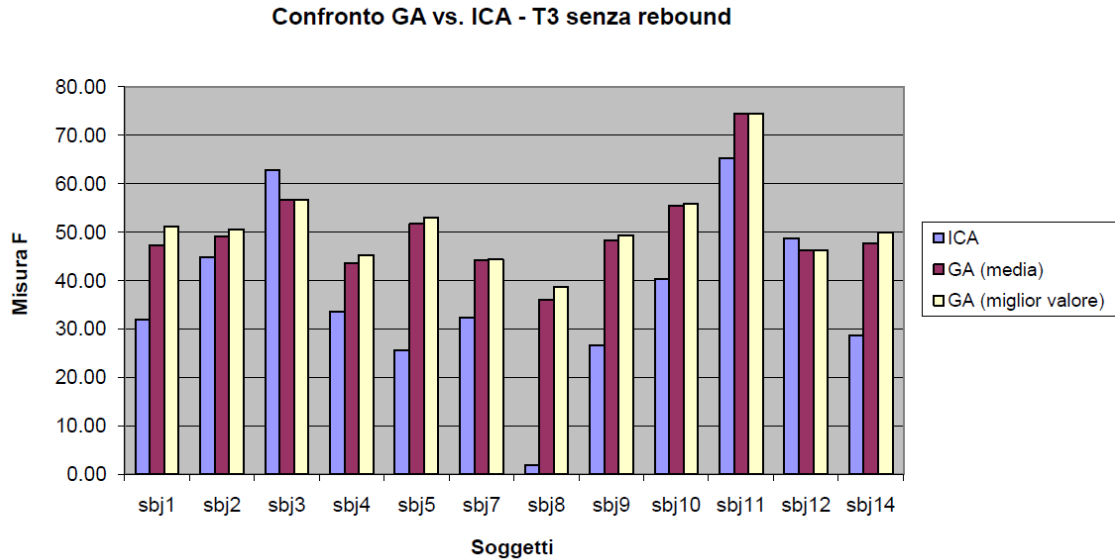
**Figura 18** - Confronto performance GA vs ICA secondo il dataset G1.



**Figura 19** - Confronto performance GA vs ICA secondo il dataset G2.



**Figura 20** - Confronto performance GA vs ICA secondo il dataset T2.



**Figura 21** - Confronto performance GA vs ICA secondo il dataset T3.

Anche a prima vista si può notare come le *performance* del nostro sistema si attestino su valori più alti, mantenendosi costanti tra i soggetti.

In Figura 18, relativa al confronto GA vs ICA per il *dataset* G1, possiamo osservare come il sistema ICA oscilli tra valori molto lontani: per il soggetto sbj8 abbiamo un valore estremamente basso, mentre per il soggetto sbj11 abbiamo un valore che indica una classificazione quasi perfetta, vicina a un valore di misura F-bilanciata pari a quasi 90; il nostro sistema si mantiene su valori più vicini tra loro, raggiungendo un picco di *performance* sicuramente più basso, ma risultando superiore per 10 soggetti su 12, e non andando mai al di sotto di un valore di misura F-bilanciata pari a 40.

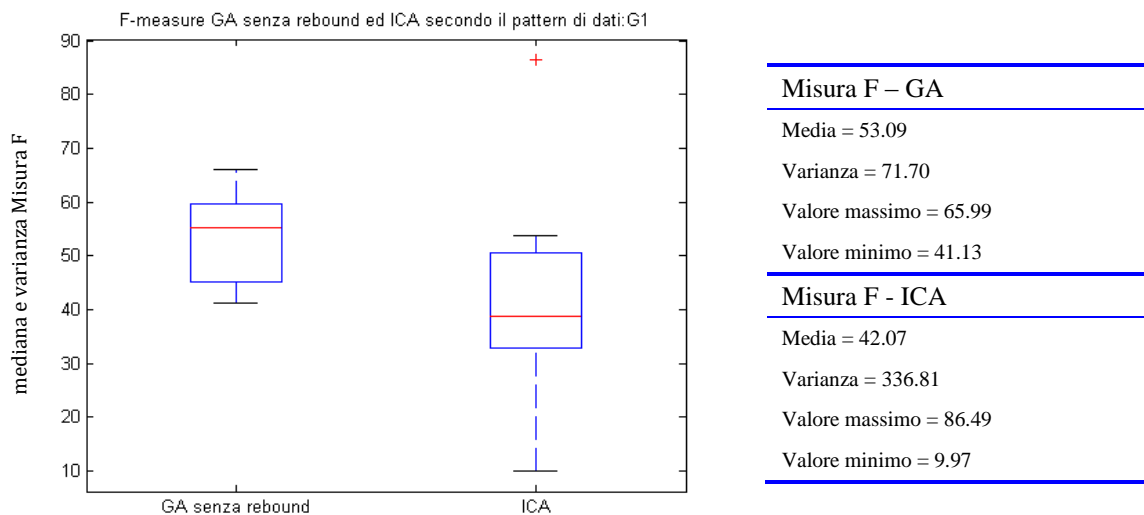
Considerazioni analoghe possono essere fatte per la Figura 19, relativa al confronto GA vs ICA per il *dataset* G2. In questo caso, il sistema ICA non raggiunge picchi di *performance* molto elevati, e peggiora ulteriormente la sua prestazione minima, molto vicina allo 0; anche il nostro sistema peggiora generalmente le proprie *performance*, ma continua a risultare superiore per 9 soggetti su 12. Anche se rispetto al *dataset* G1 viene modificato il *set* di *testing*, restano in linea le *performance* inter-soggettive relative al nostro sistema: le prestazioni migliori si hanno per i soggetti sbj11, sbj10, sbj5, sbj3 e sbj02; per il sistema ICA, a parte il caso del soggetto sbj11, ci sono molte oscillazioni nelle prestazioni tra i due *dataset*.

In Figura 20, relativa al confronto GA vs ICA per il *dataset* T2, il nostro sistema risulta superiore al sistema ICA per 11 soggetti su 12. Resta molto distante il picco

raggiunto per il soggetto sbj11 da parte del sistema ICA, ma il nostro sistema si mantiene costante nelle prestazioni, peggiorando notevolmente solo nel caso del soggetto sbj8, altro caso particolare, diametralmente opposto al soggetto sbj11.

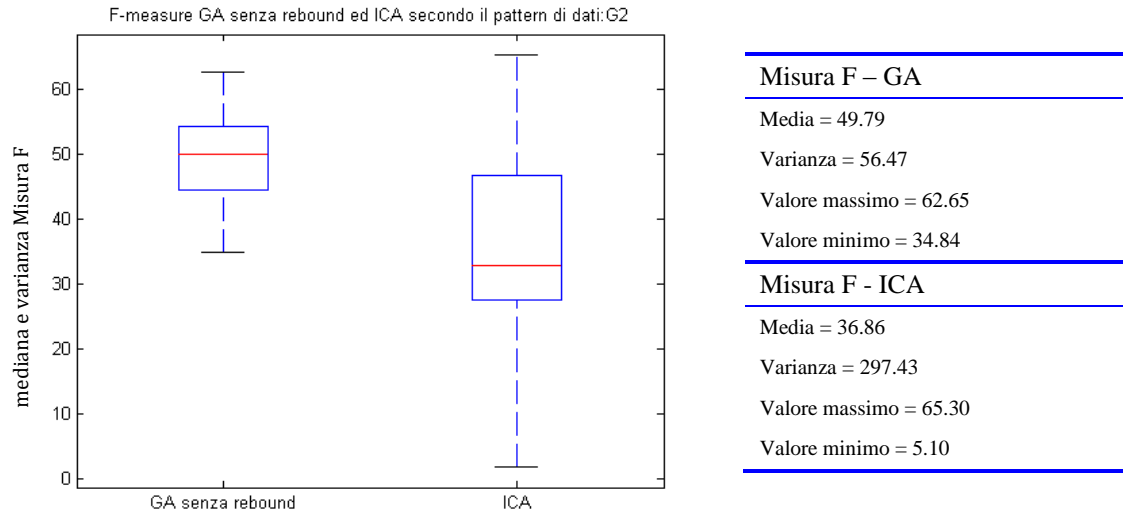
In Figura 21, relativa al confronto GA vs ICA per il *dataset* T3, il nostro sistema raggiunge finalmente le prestazioni migliori, ottenendo per il soggetto sbj11 il valore massimo di misura F-bilanciata per l'intero *dataset*. Nel confronto inter-soggettivo, il nostro sistema risulta superiore in 10 casi su 12, mantenendosi quasi sempre sopra la soglia di misura F-bilanciata pari a 40, a parte il soggetto sbj8, sul quale sono stati ottenuti comunque degli ottimi risultati rispetto alle prestazioni del sistema ICA.

Di seguito riportiamo i *boxplot*<sup>17</sup> relativi alle *performance* sui 12 soggetti, confrontando le *performance* medie tra il nostro sistema e il sistema ICA.

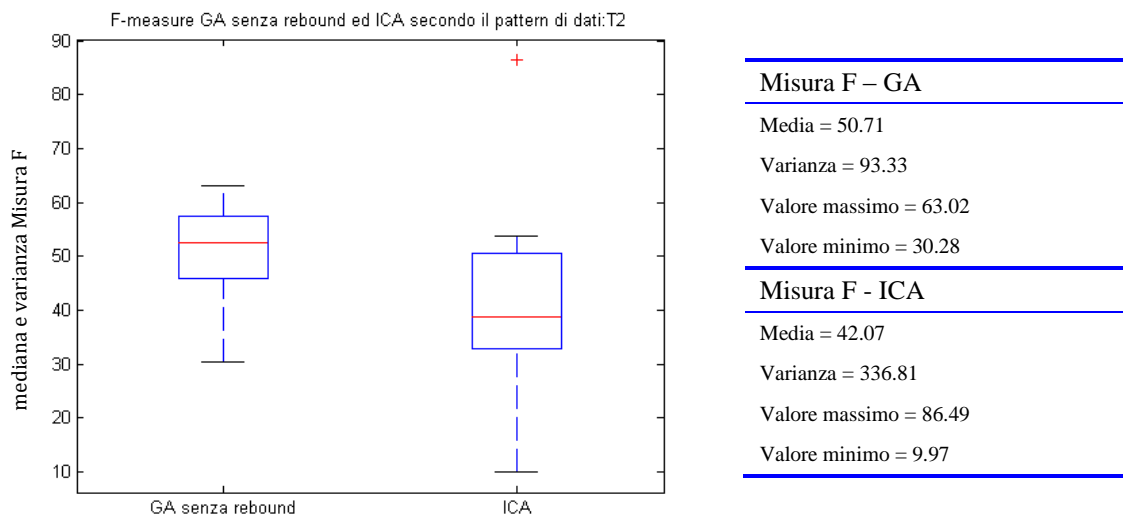


**Figura 22** - *Boxplot del confronto performance GA vs ICA secondo il dataset G1, con relativi valori numerici.*

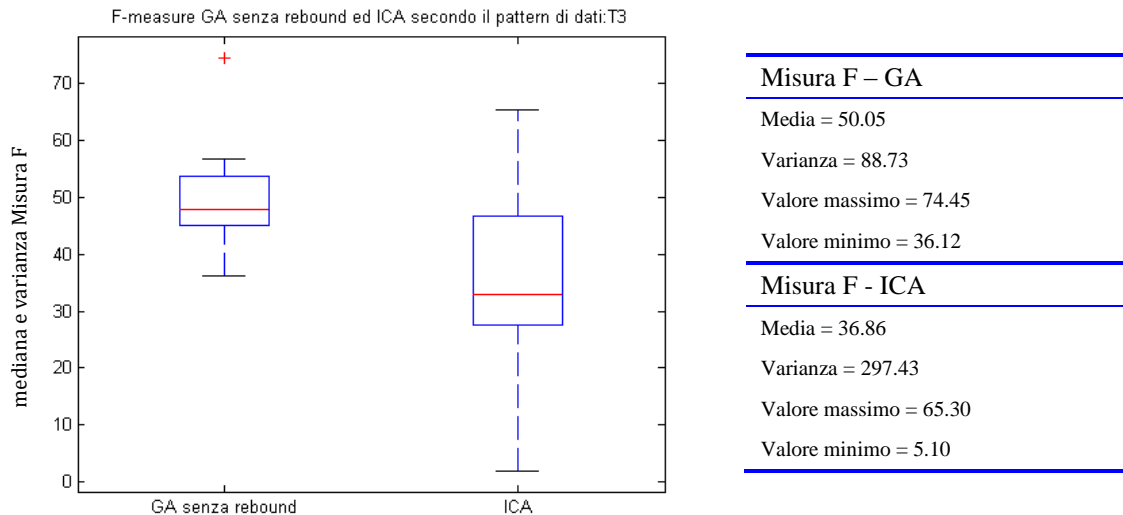
<sup>17</sup> Nei *boxplot* realizzati tramite *tool* Matlab viene usata la mediana per rappresentare i dati: questo perché la mediana è una statistica più robusta (solitamente non varia di molto in presenza di *outlier*) ed è la misura statistica riassuntiva preferibile alla media per definire la posizione “centrale” di una distribuzione asimmetrica.



**Figura 23** - Boxplot del confronto performance GA vs ICA secondo il dataset G2 con relativi valori numerici.



**Figura 24** - Boxplot del confronto performance GA vs ICA secondo il dataset T2 con relativi valori numerici.



**Figura 25** - Boxplot del confronto performance GA vs ICA secondo il dataset T3 con relativi valori numerici.

Si può notare come le *performance* siano migliori per il nostro sistema e che i risultati si mantengono in un *range* di valori più limitato, nonostante la variabilità inter-soggettiva.

Analizzando nello specifico le prestazioni, notiamo in Figura 22, relativa alle *performance* sul dataset G1, come la media delle *performance* del nostro sistema sia nettamente superiore a quella del sistema ICA, che si assesta su un valore di 42.07 rispetto al 53.09 del nostro sistema: c'è pertanto un miglioramento di più del 25% rispetto alle prestazioni del sistema ICA. Un dato interessante, anticipato nel commento alla Figura 18, è l'alta variabilità delle prestazioni osservate per il sistema ICA: la varianza è quasi 5 volte superiore alla varianza dei risultati del nostro sistema (336.81 contro 71.70); come conferma, osserviamo la grande differenza tra valore massimo e valore minimo del sistema ICA; nonostante la grande varianza espressa, sembra che il valore relativo al soggetto sbj11 sia un *outlier*, anche se non sembra molto significativo questo dato a causa della bassa numerosità del campione.

In Figura 23 osserviamo le *performance* relative al dataset G2: in questo caso il miglioramento osservato risulta ancora più netto, a fronte di una media che in valore assoluto risulta inferiore rispetto al dataset precedente (49.79 contro 36.86 espresso dal sistema ICA); anche la varianza mostra un valore più contenuto per entrambi i sistemi, con il sistema ICA che mostra comunque una variabilità sulle prestazioni molto alta, passando da un valore minimo di soli 5.10 a un valore massimo di 65.30, a fronte dei

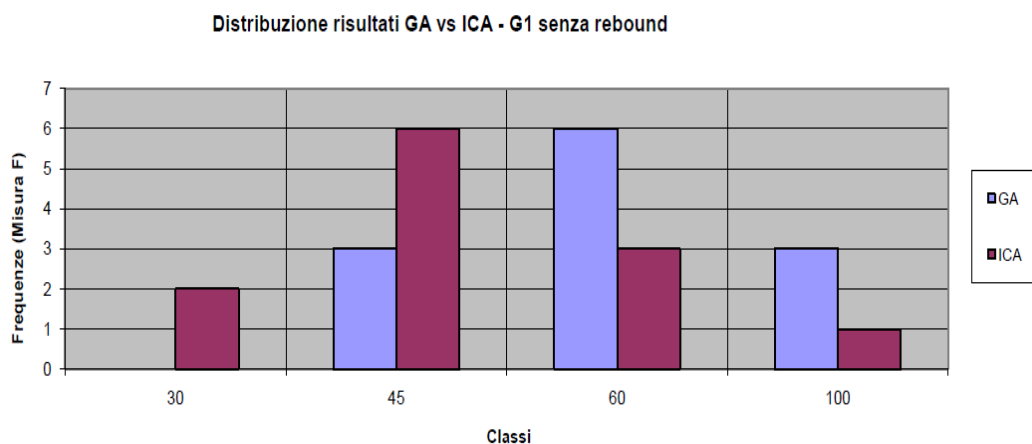
valori relativi al nostro sistema, che mostrano un valore minimo di 34.84 e un valore massimo di 62.65.

In Figura 24, relativa alle prestazioni sul *dataset* T2, osserviamo un generale peggioramento delle *performance* da parte del nostro sistema: il confronto con il *dataset* G1 [Figura 22], con cui il *dataset* T2 condivide gran parte dei dati, vede superiore l'addestramento sul *dataset* con *sweep* selezionate (G1), anche se le prestazioni non sono molto distanti tra loro; rispetto al sistema ICA, le prestazioni sono comunque migliori, anche se in modo non così netto come le prestazioni sul *dataset* G1.

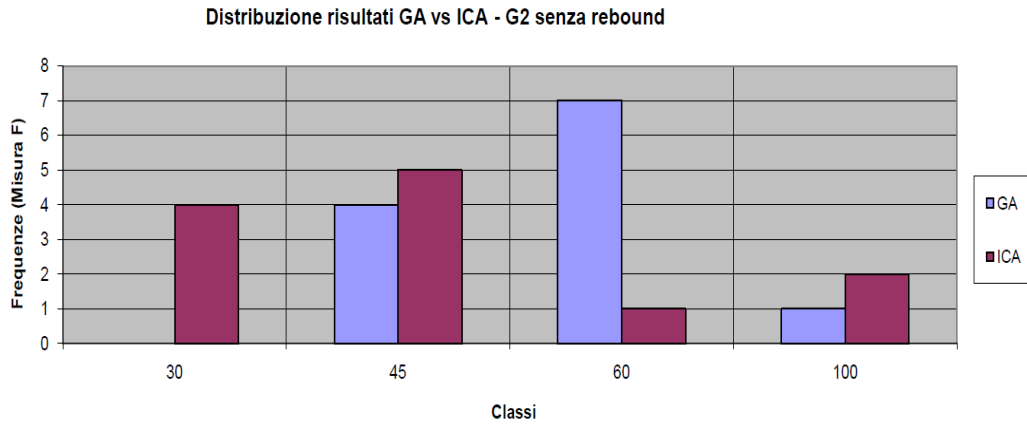
In Figura 25, relativa al *dataset* T3, vediamo un comportamento opposto del nostro sistema: in questo caso, includere *sweep* non considerate nel *dataset* G2 [Figura 19] sembra aver portato un leggero vantaggio (valor medio di 50.05 contro 49.79, valor massimo 74.45 contro 62.65) al nostro sistema.

Possiamo desumere che molto probabilmente la natura del nostro algoritmo di ricerca, fa sì che possano verificarsi delle piccole variazioni di prestazioni da una ripetizione all'altra dell'algoritmo, variazioni che però non precludono il risultato finale: il nostro sistema appare migliore del sistema ICA, rispetto alla metrica di valutazione considerata, per tutti i *dataset* considerati.

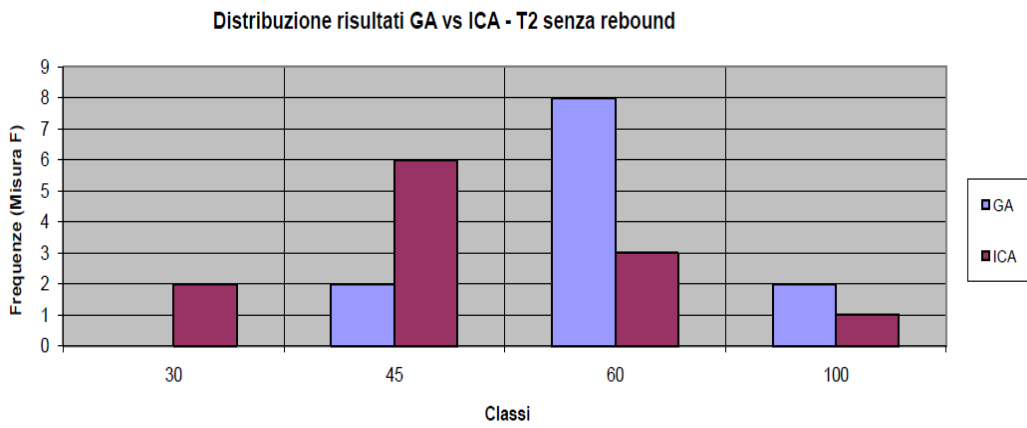
Per evidenziare la differenza tra le prestazioni nei due metodi e per stabilire se sussiste una significatività statistica, abbiamo quindi rappresentato i risultati in frequenze relative [Figura 26, Figura 27, Figura 28, Figura 29] ed eseguito un test t di Student delle *performance* [Tabella 1]; l'ipotesi nulla è che le differenze osservate tra le distribuzioni siano dovute al caso, con un livello di significatività pari a 0.05.



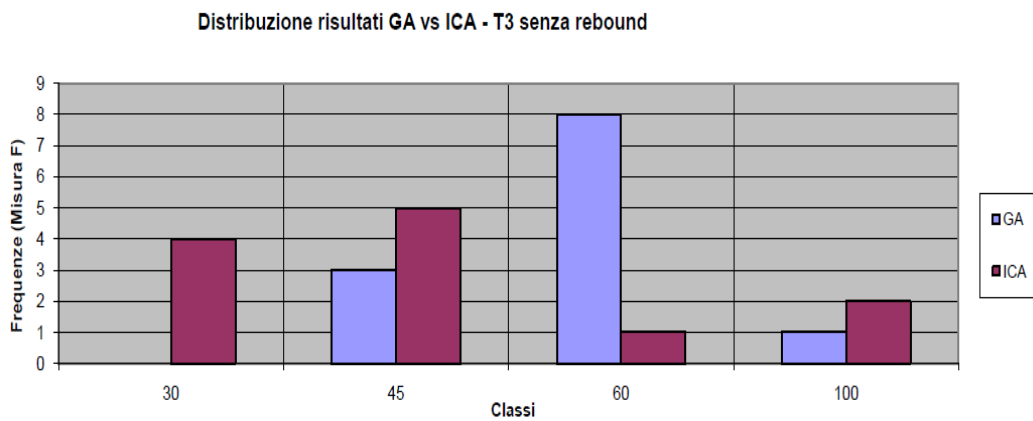
**Figura 26** - Frequenze relative performance GA vs ICA secondo il *dataset* G1.



**Figura 27** - Frequenze relative performance GA vs ICA secondo il dataset G2.



**Figura 28** - Frequenze relative performance GA vs ICA secondo il dataset T2.



**Figura 29** - Frequenze relative performance GA vs ICA secondo il dataset T3.



GA vs ICA: tipo di <i>dataset</i>	p-value
G1 senza <i>rebound</i>	0.01
G2 senza <i>rebound</i>	0.002
T2 senza <i>rebound</i>	0.02
T3 senza <i>rebound</i>	0.001

**Tabella 1-** Risultati test *t* di Student relativi al confronto performance GA vs ICA per tutti i *dataset*.

Nelle Figura 26, 27, 28Figura 29, possiamo osservare come per nessun *dataset* il nostro sistema ha prestazioni inferiori a 30, mentre per il sistema ICA si hanno 2 casi relativi al *dataset* G1 [Figura 26] e 4 relativi al *dataset* G2 [Figura 27]. La distribuzione dei risultati del nostro sistema sembra essere una simil-Gaussiana per tutti i *dataset*, centrata nella classe  $45 < x < 60$ ; per il sistema ICA appare un deciso sbilanciamento verso i valori più bassi, con una distribuzione dei risultati senza un particolare andamento (si veda in particolare la Figura 27).

Si può notare [Tabella 1] come le differenze siano statisticamente significative per tutte i *dataset*: il nostro sistema ha quindi *performance* mediamente migliori, e il miglioramento osservato è statisticamente significativo.

## 4.1.2 GA senza *rebound* vs GA con *rebound*

Confrontiamo all'interno dei risultati del nostro sistema le *performance* in presenza o in assenza di *rebound*, per l'interfaccia "Arrows"; la finestra temporale relativa al caso con *rebound* è [-0.2 1] secondi, mentre quella relativa al caso senza *rebound* è [-0.2 0.8].

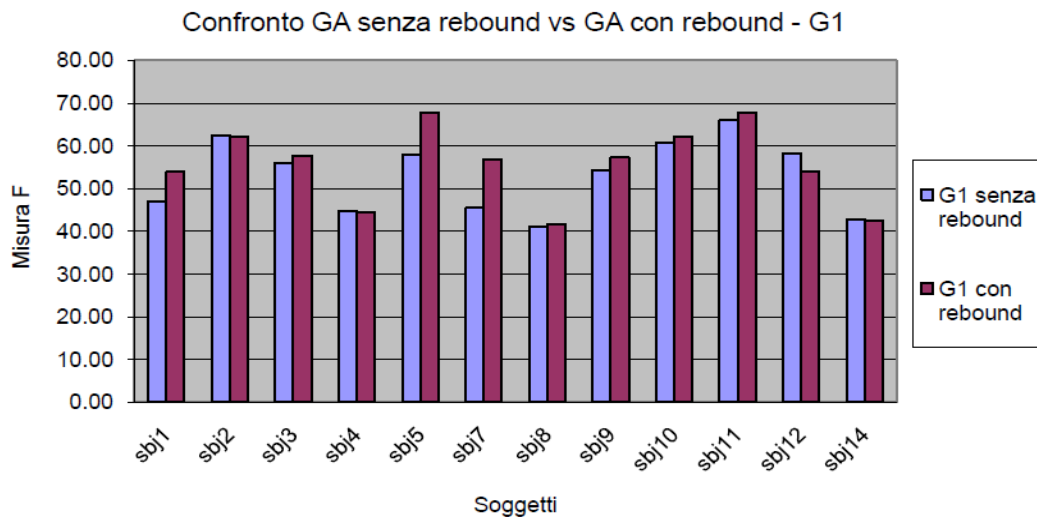


Figura 30 - Confronto performance GA senza rebound vs GA con rebound, secondo il dataset G1.

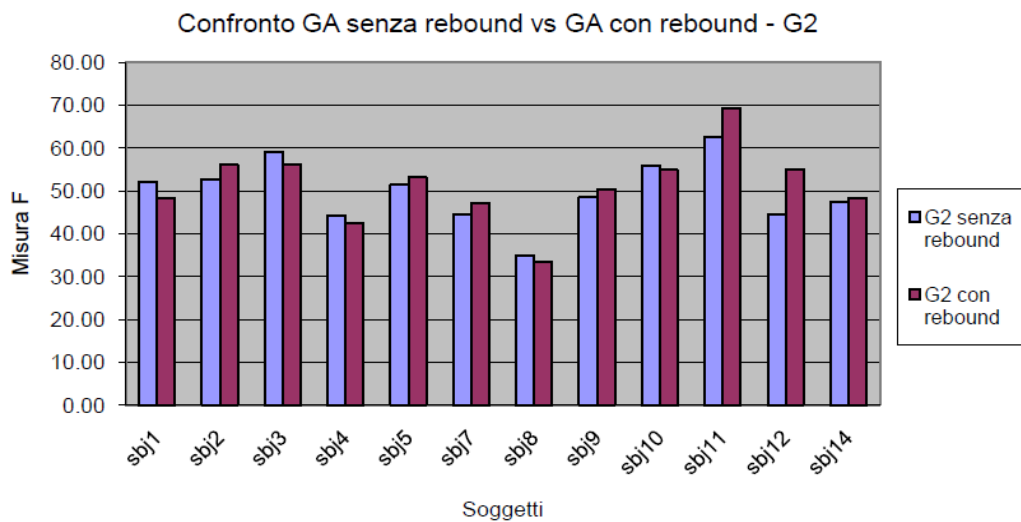


Figura 31 - Confronto performance GA senza rebound vs GA con rebound, secondo il dataset G2.

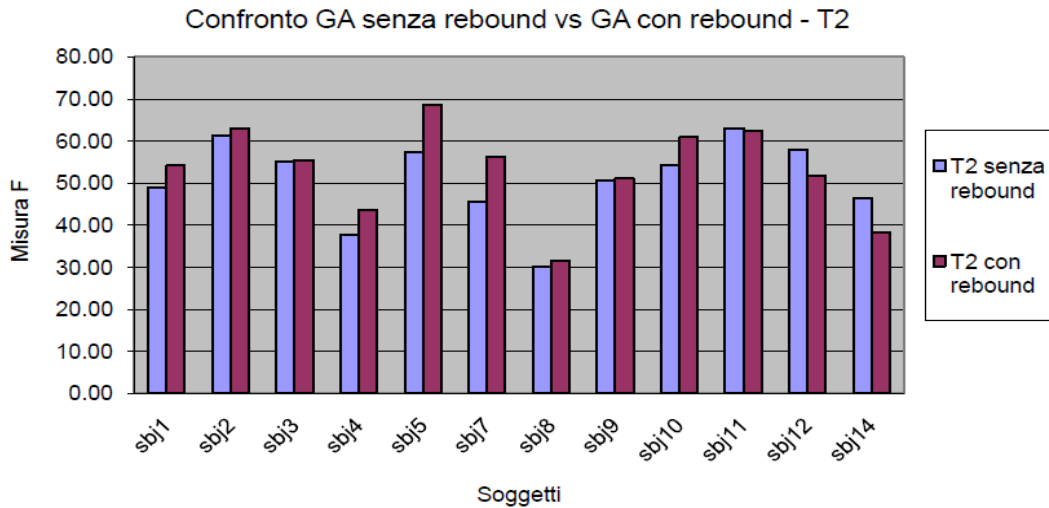


Figura 32 - Confronto performance GA senza rebound vs GA con rebound, secondo il dataset T2.

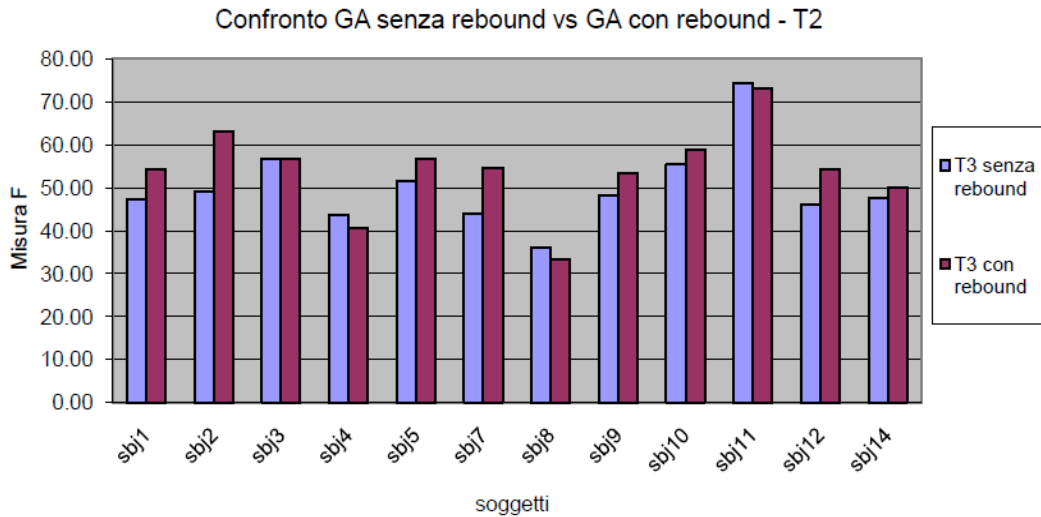


Figura 33 – Confronto performance GA senza rebound vs GA con rebound, secondo il dataset T3.

In questo confronto le *performance* mostrano minori differenze, ma bisogna considerare che è stata aggiunta solo una piccola parte del segnale, circa 200 ms; nonostante ciò, inter-soggettivamente le performance sono nella maggior parte dei casi migliori nel caso con *rebound*.

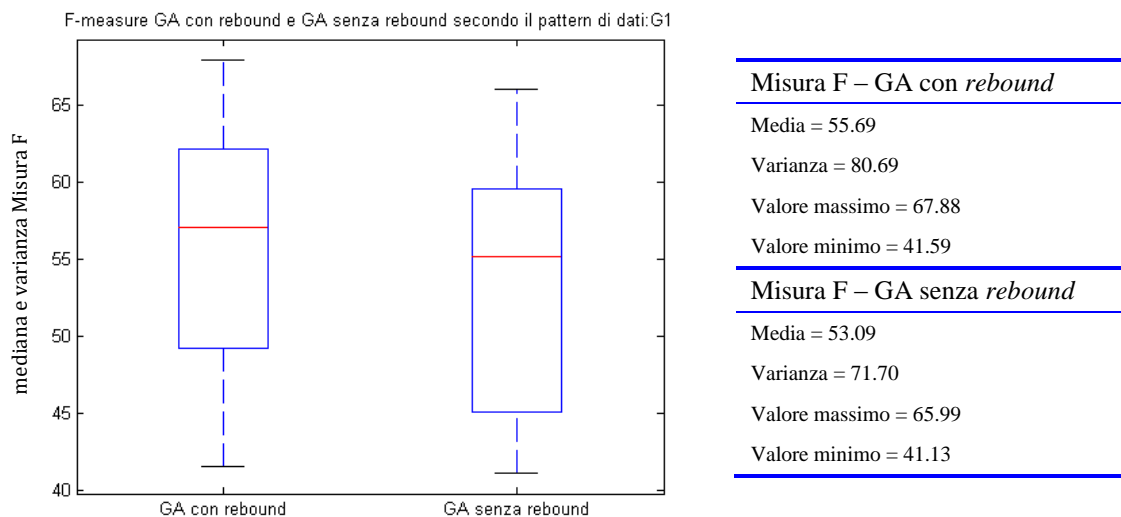
In Figura 30, relativa al confronto sul *dataset* G1, a parte i soggetti sbj5 e sbj7, in cui si registra un aumento sostanziale delle *performance*, per il resto le prestazioni sono confrontabili; tuttavia, in 8 soggetti su 12 l'uso del *rebound* risulta superiore nel confronto; per i soggetti in cui risulta superiore il sistema senza *rebound*, le differenze risultano comunque molto basse.

In Figura 31 possiamo trarre le stesse conclusioni: a parte i soggetti sbj11 e sbj12, le *performance* sono confrontabili; l'uso del *rebound* risulta vantaggioso per 7 soggetti su 12, quindi ancora per la maggior parte dei casi.

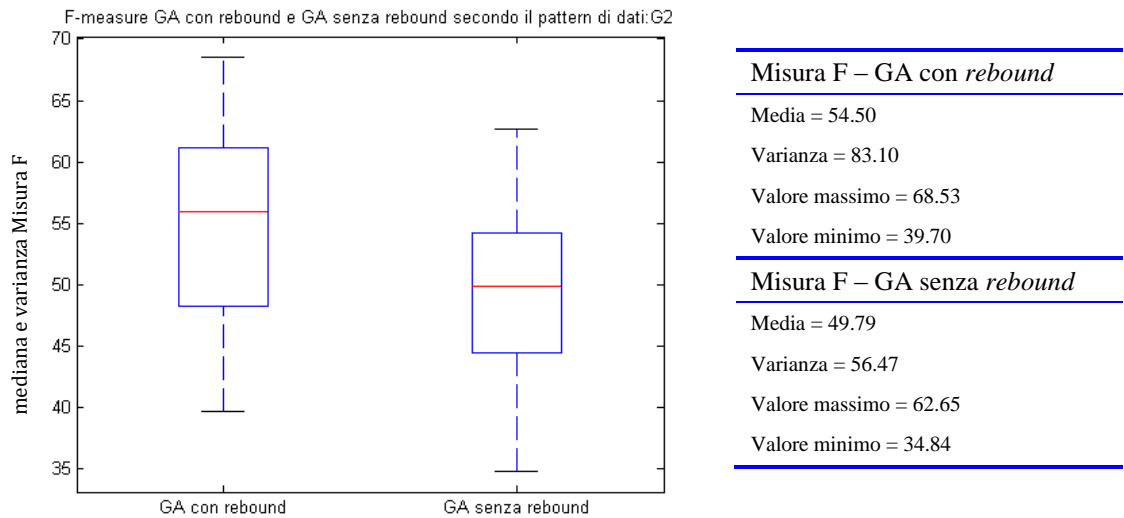
In Figura 32 osserviamo che per 8 soggetti su 12 il sistema addestrato con finestra temporale [0.2 1] ha dei vantaggi ben visibili, con differenze maggiori per i soggetti sbj5 e sbj7, in analogia con il *dataset* G1 [Figura 30].

Conclusioni simili traiamo dall'analisi del grafico in Figura 33, relativa al *dataset* T3, solo che in questo caso i soggetti su cui il sistema trae più vantaggio dall'inclusione del *rebound* sono sbj2 e sbj7.

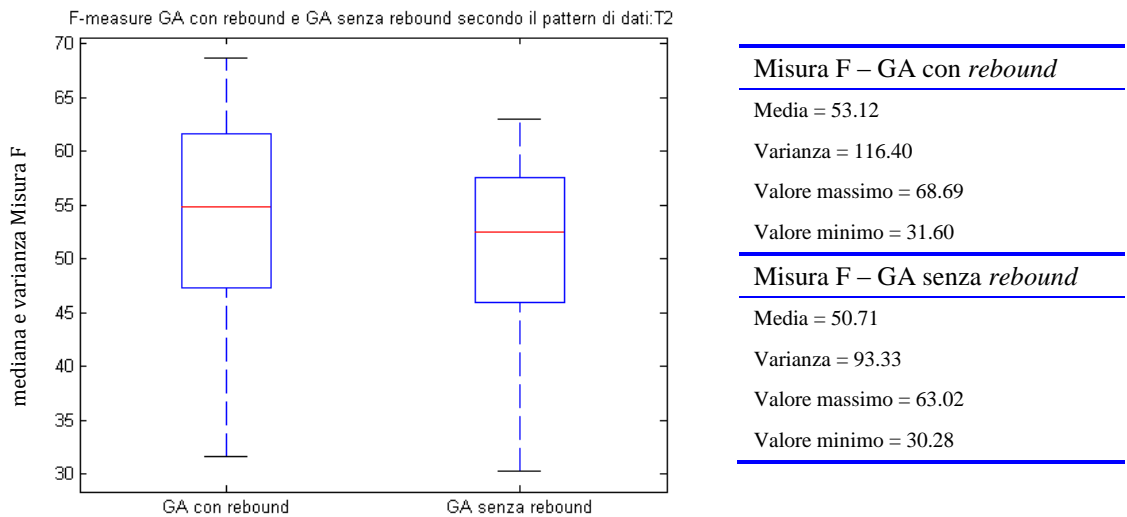
Osservando i dati statistici e i *boxplot* [Figura 34Figura 35Figura 36Figura 37], questi dati appaiono più evidenti.



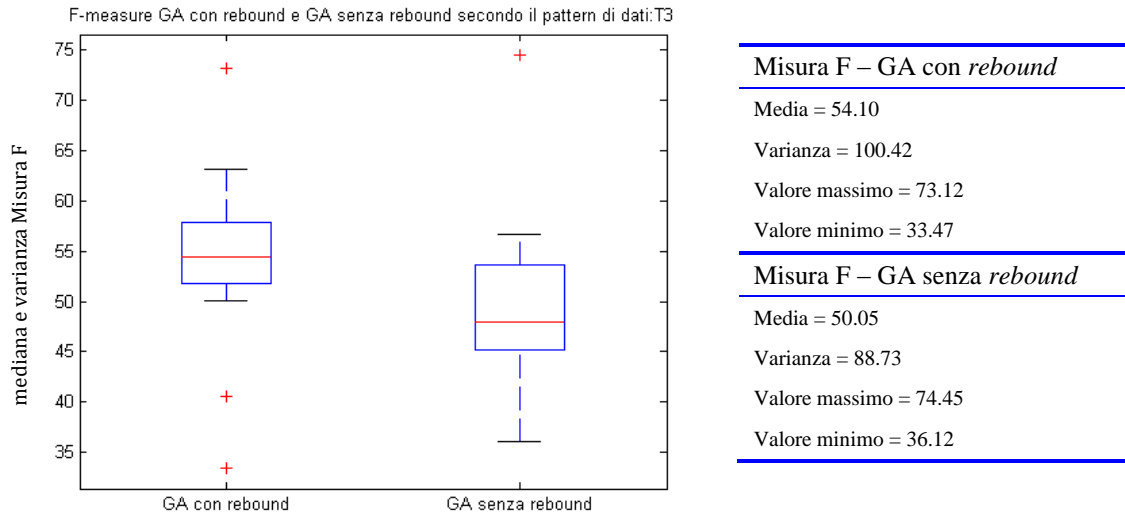
**Figura 34** - *Boxplot del confronto performance GA con rebound vs GA senza rebound, secondo il dataset G1, con relativi valori numerici.*



**Figura 35** - Boxplot del confronto performance GA con rebound vs GA senza rebound, secondo il dataset G2, con relativi valori numerici.



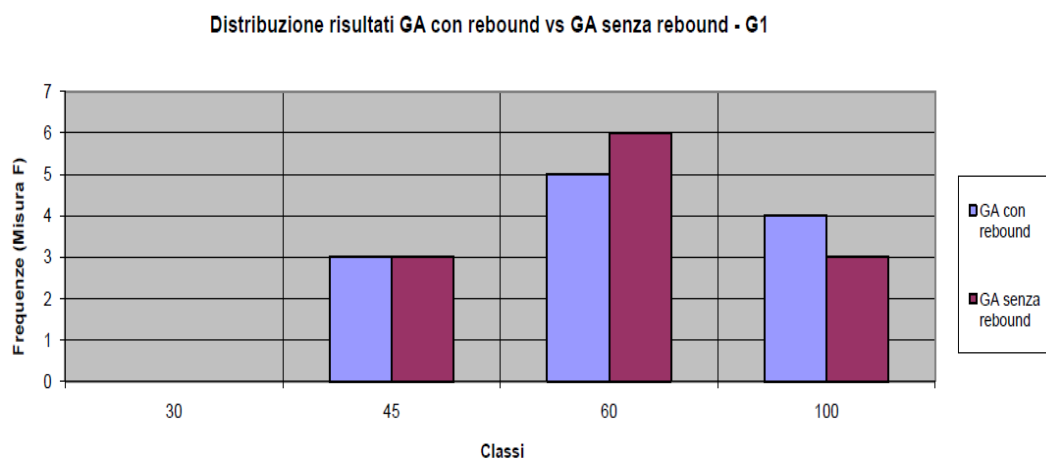
**Figura 36** - Boxplot del confronto performance GA con rebound vs GA senza rebound, secondo il dataset T2, con relativi valori numerici.



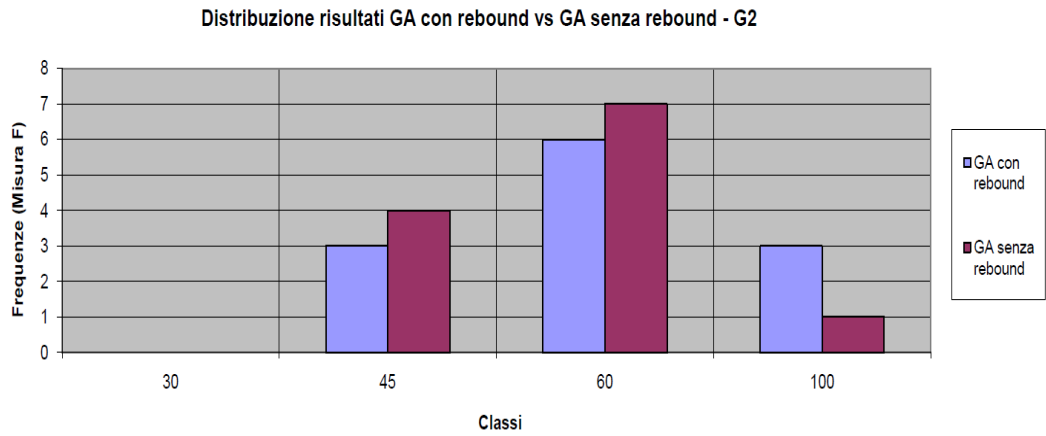
**Figura 37** - Boxplot del confronto performance GA con rebound vs GA senza rebound, secondo il dataset T3, con relativi valori numerici.

Possiamo notare come in tutti i *dataset* ci sia stato un miglioramento in media e nei valori massimi e minimi, escluso il *dataset* T3, che mostra un peggioramento in questi ultimi due valori, nonostante sia il *dataset* che anche senza *rebound* restituisce la *performance* migliore per il soggetto sbj11 [Figura 36]; i miglioramenti registrati in media vanno da una differenza di 5 unità [Figura 35], fino a un minimo di meno di 3 unità [Figura 36].

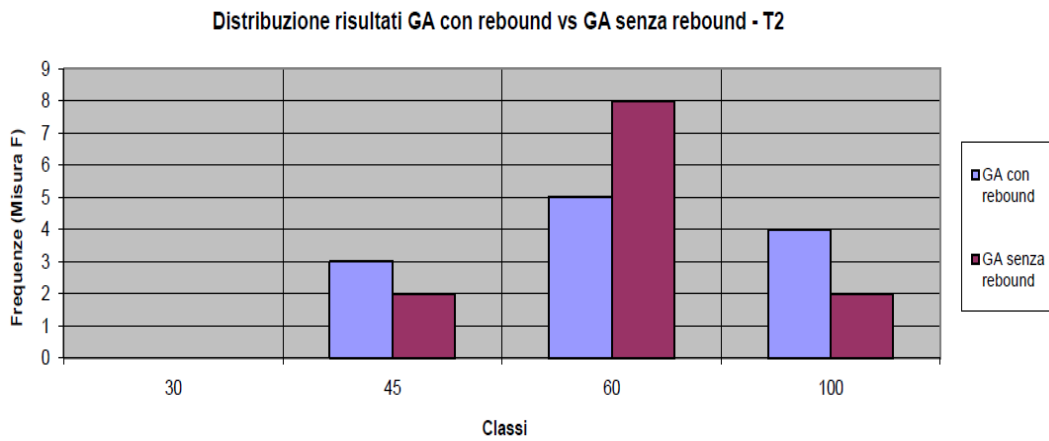
Passiamo ad analizzare la distribuzione secondo classi di *performance* e i test t di Student.



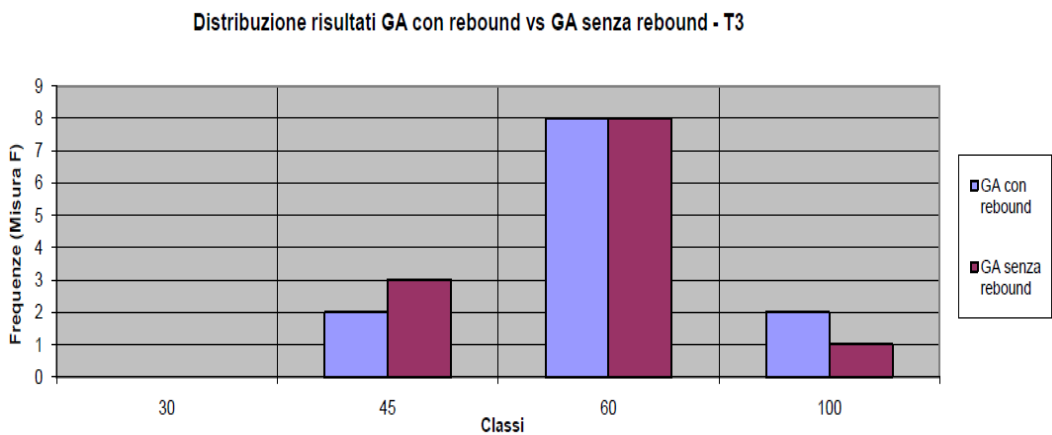
**Figura 38** - Frequenze relative performance GA con rebound vs GA, senza rebound, secondo il dataset G1.



**Figura 39** - Frequenze relative performance GA con rebound vs GA, senza rebound, secondo il dataset G2.



**Figura 40** - Frequenze relative performance GA con rebound vs GA, senza rebound, secondo il dataset T2.



**Figura 41** - Frequenze relative performance GA con rebound vs GA, senza rebound, secondo il dataset T3.

GA con <i>rebound</i> vs GA senza <i>rebound</i> : tipo di <i>dataset</i>	p-value
G1	0.04
G2	0.01
T2	0.09
T3	0.01

**Tabella 2-** Risultati test *t* di Student relativi al confronto performance GA con *rebound* vs GA, senza *rebound*, per tutti i *dataset*.

Nell'analisi delle Figura 38Figura 39Figura 40Figura 41, il miglioramento è osservabile nell'aumento del numero di risultati nella classe maggiore (>60) per l'addestramento con *rebound*, a parte un singolo caso per il *dataset* T2 [Figura 40], in cui aumenta il numero dei risultati anche nella classe inferiore (<45).

Si può notare in Tabella 2- **Risultati test *t* di Student relativi al confronto performance GA con *rebound* vs GA, senza *rebound*, per tutti i *dataset***. come le differenze siano statisticamente significative per quasi tutti i *dataset*: allargare la finestra temporale fino ad includere parte del *rebound* permette di ottenere *performance* mediamente migliori, e il miglioramento osservato è statisticamente significativo; a questo proposito, è corretto aggiungere che la parte di segnale aggiunta rispetto al settaggio iniziale del *tool* non sembra rappresentare a pieno la componente lenta che inizialmente si voleva includere; l'uso di registrazioni EEG maggiormente rappresentative, che terminino almeno 1.5 s dopo l'invio del segnale visivo, completerebbe lo studio iniziato in questo contesto, quantificando l'utilità del *rebound* nella classificazione di segnali *target* e non *target*.

L'importanza del nostro studio risiede nel tentativo di stabilire se ci siano altre componenti caratteristiche di un segnale *target* utilizzabili nel contesto delle BCI, e la risposta, benchè soggetta ad ulteriori approfondimenti e verifiche, sembra essere positiva.



### 4.1.3 Confronto *dataset*

Per testare la robustezza della classificazione, abbiamo confrontato i *dataset* G1 vs. G2 e T2 vs. T3, che condividono la stessa struttura di *training* ma differiscono nel *set* di *testing* (G1 e T2 hanno come *set* di *testing* la sessione *last testing*; G2 e T3 la sessione di *follow-up*).

Confronto <i>dataset</i> ; GA senza <i>rebound</i>	p-value
G1 vs G2	0.04
T2 vs T3	0.37

**Tabella 3** - Risultati test *t* di Student relativi al confronto *dataset* per GA, senza *rebound*.

Confronto <i>dataset</i> ; GA con <i>rebound</i>	p-value
G1 vs G2	0.26
T2 vs T3	0.30

**Tabella 4** - Risultati test *t* di Student relativi al confronto *dataset* per GA, con *rebound*.

Le *performance* non mostrano differenze statisticamente significative [Tabella 3 e 4] modificando il *set* di *testing*, escluso il caso di G1 vs G2 per l'esecuzione senza *rebound*, probabilmente a causa delle non eccellenti prestazioni relative, ottenute dall'algoritmo genetico nel *dataset* G2 con finestra temporale [-0.2 0-8] secondi, essendo il *dataset* che registra il massimo scarto tra le *performance* senza *rebound* e le *performance* con *rebound*.

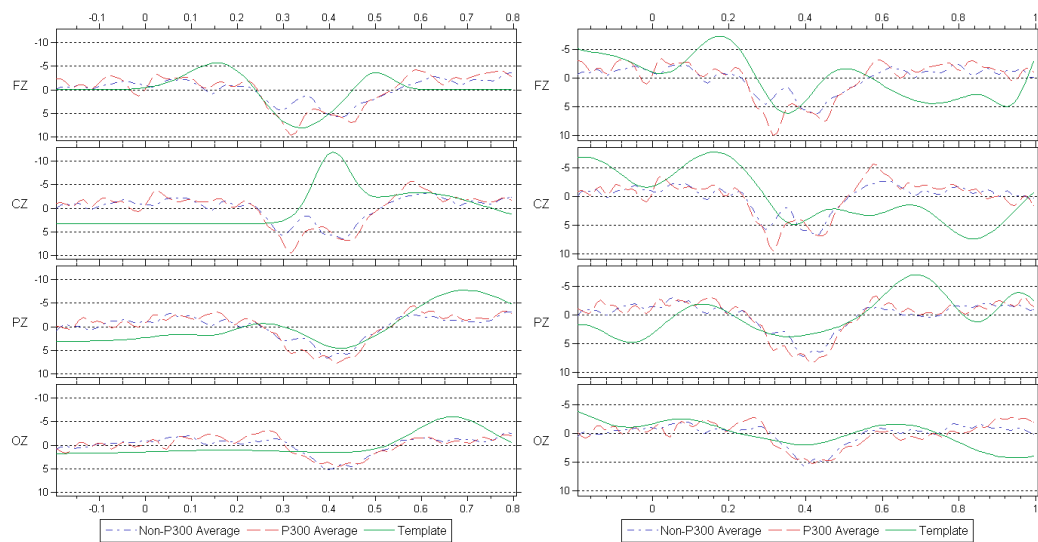
Questi risultati mostrano una buona generalizzazione del sistema; ciò è probabilmente dovuto alla cross-validazione in fase di esecuzione dell'algoritmo.

Il sistema ICA non sembra manifestare la stessa capacità di generalizzazione (nel confronto G1 vs G2 il *p-value* è pari a 0.08, poco al di sopra del livello di significatività), anche se l'alta varianza dei risultati e i numerosi *outlier*, impediscono di trarre conclusioni in merito.

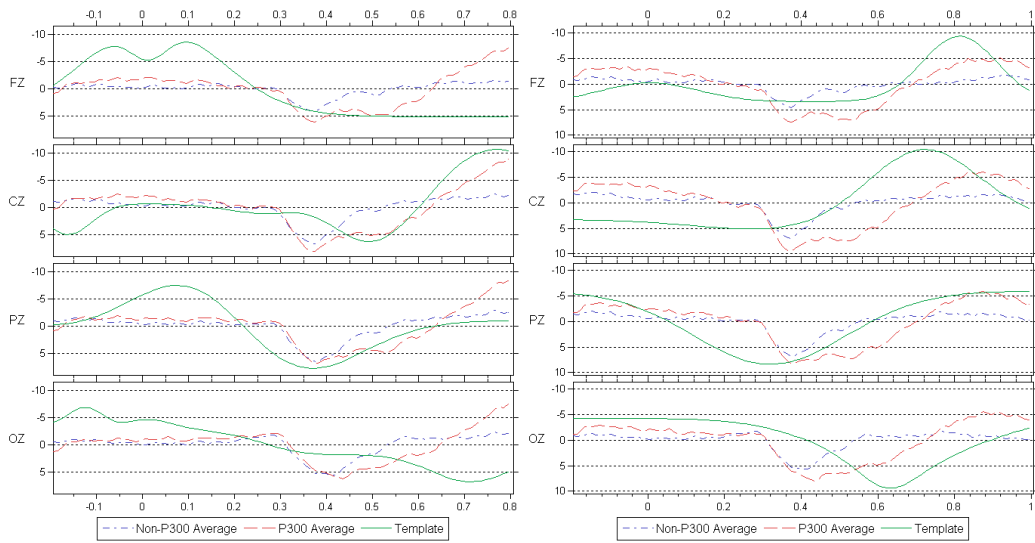
#### 4.1.4 *Template*

Per le caratteristiche del sistema, i *template* ottenuti non dipendono né dal segno né dall'ampiezza del segnale; *a posteriori* abbiamo normalizzato ogni *template* rispetto al *range* di valori del segnale originario, e tentato di assegnare un segno; nei casi in cui è assente il *template* per uno o più canali, significa che l'algoritmo non è riuscito ad estrarre da quel canale *feature* con una *fitness* sufficientemente alta da risultare superiori in fase di selezione.

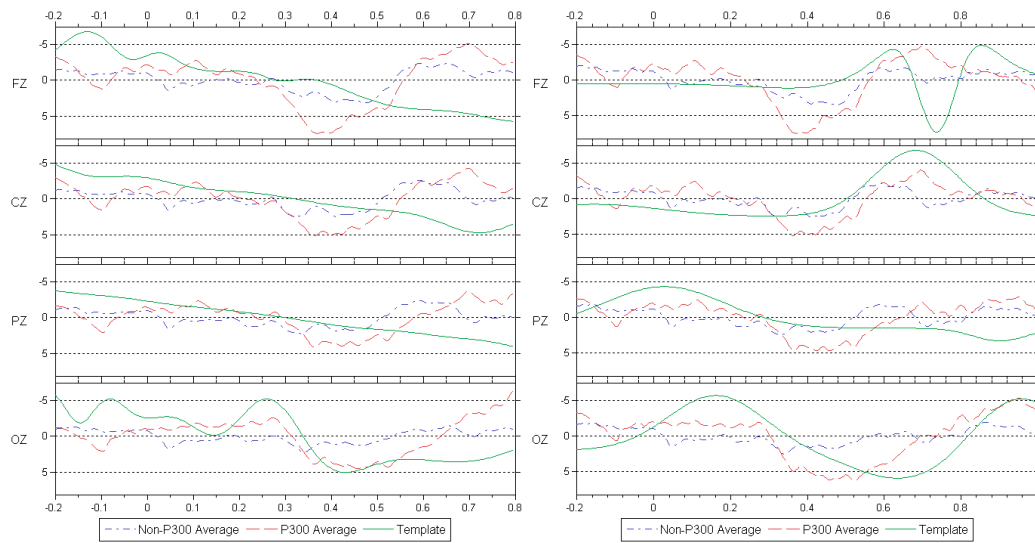
Per semplicità di trattazione, vengono presentati solo alcuni *template* rappresentativi.



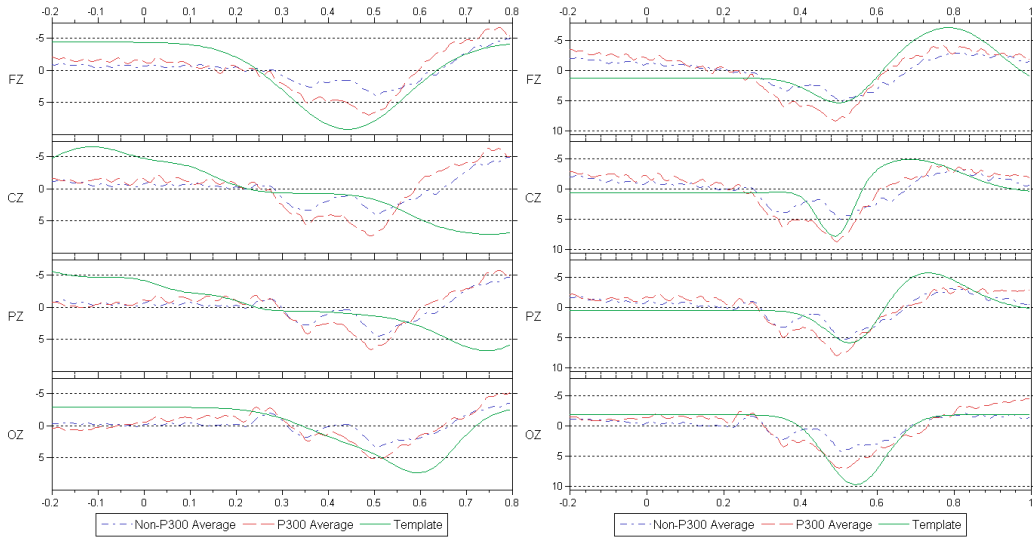
**Figura 42** - *Template* estratto per il soggetto *sbj01* sul dataset *G1* (senza rebound a sinistra e con rebound a destra).



**Figura 43** - Template estratto per il soggetto sbj03 sul dataset G2 (senza rebound a sinistra e con rebound a destra).

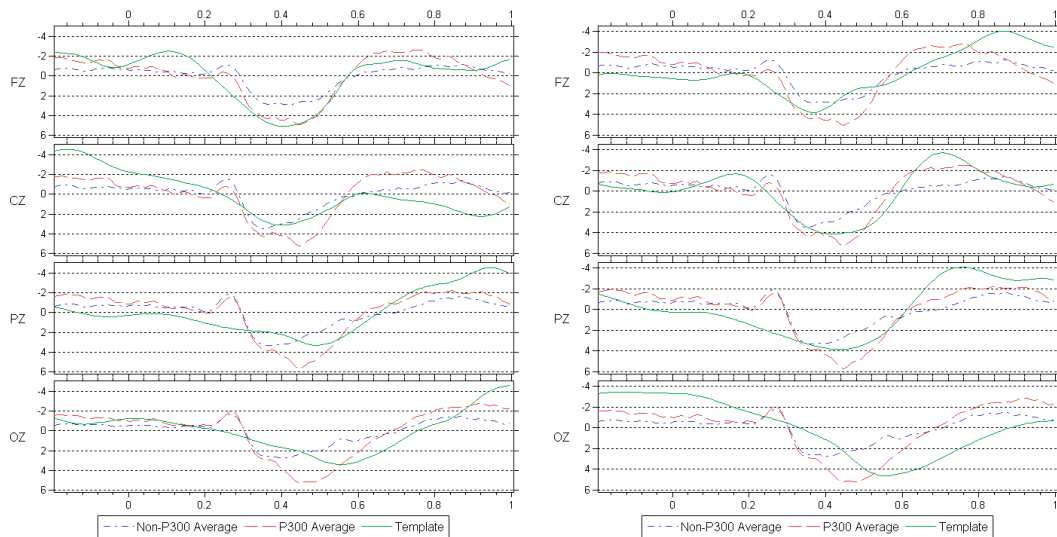


**Figura 44** - Template estratto per il soggetto sbj09 sul dataset T2 (senza rebound a sinistra e con rebound a destra).

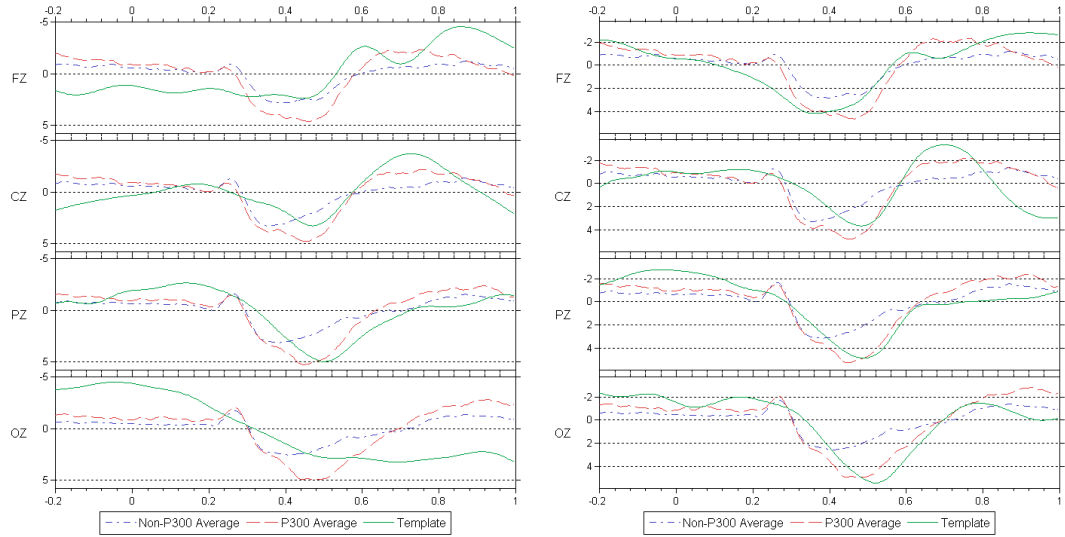


**Figura 45** - *Template* estratto per il soggetto *sbj12* sul dataset *T3* (senza rebound a sinistra e con rebound a destra).

Mediando per ogni canale le *sweep* di tutti i soggetti [Figura 46Figura 47], si può notare come i *template* seguano abbastanza fedelmente i *pattern* dei segnali *target*.



**Figura 46** - *Template* medi sui dataset *G1* (sinistra) e *G2* (destra), con rebound.

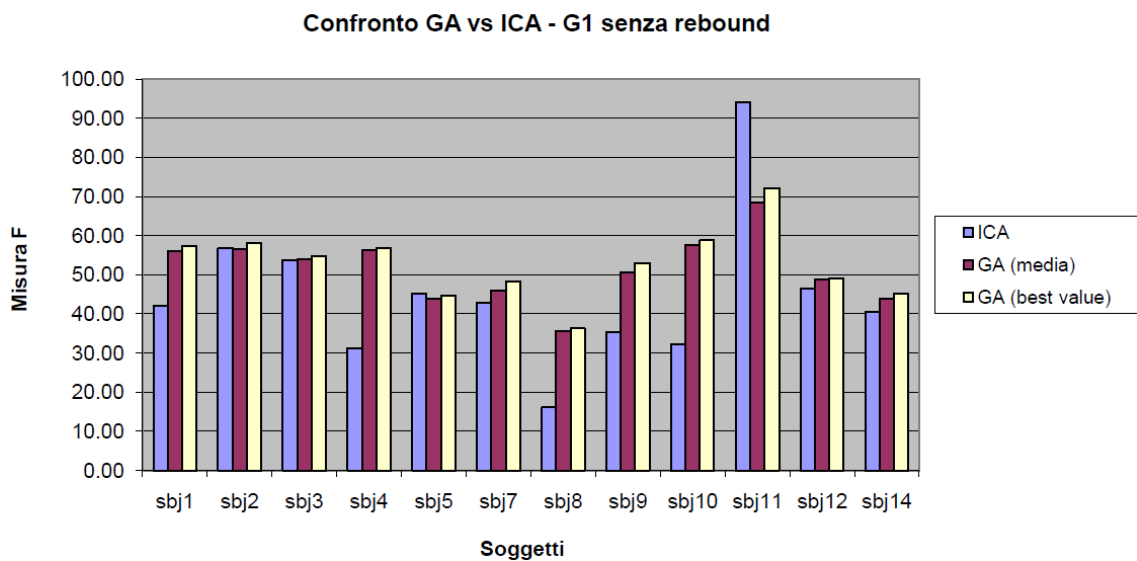


**Figura 47** - Template medi sui dataset T2 (sinistra) e T3 (destra), con rebound.

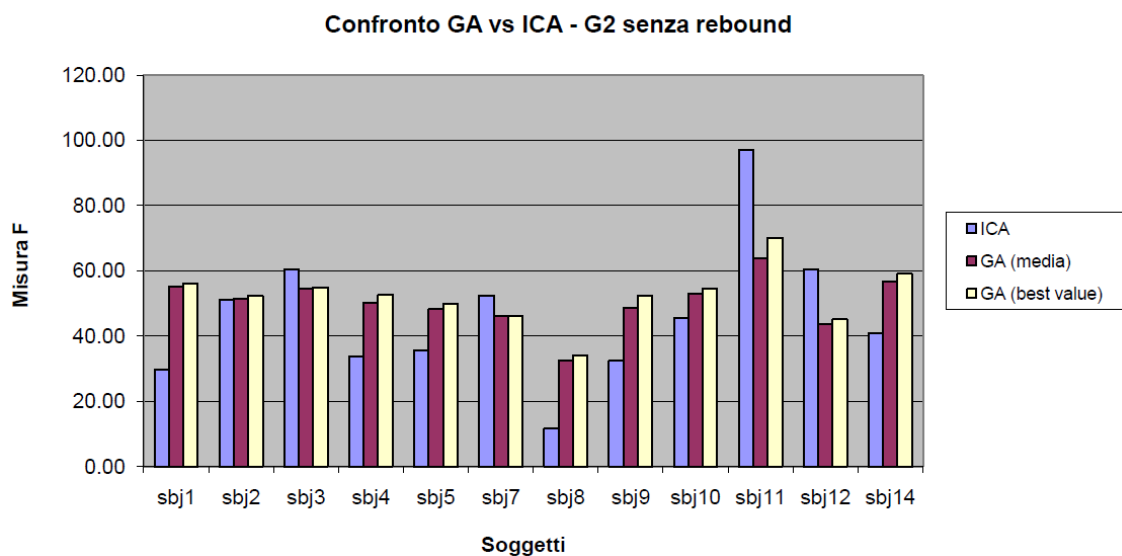
## 4.2 Interfaccia “Auto”

### 4.2.1 GA vs ICA

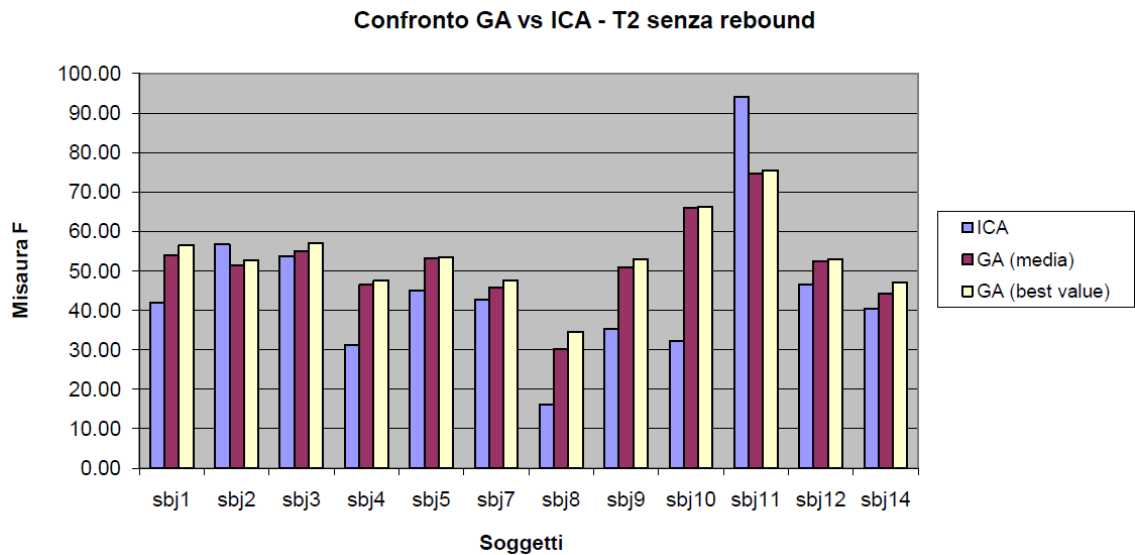
Confrontiamo le *performance* del nostro sistema e le *performance* del sistema ICA per l’interfaccia “Auto”, senza *rebound*.



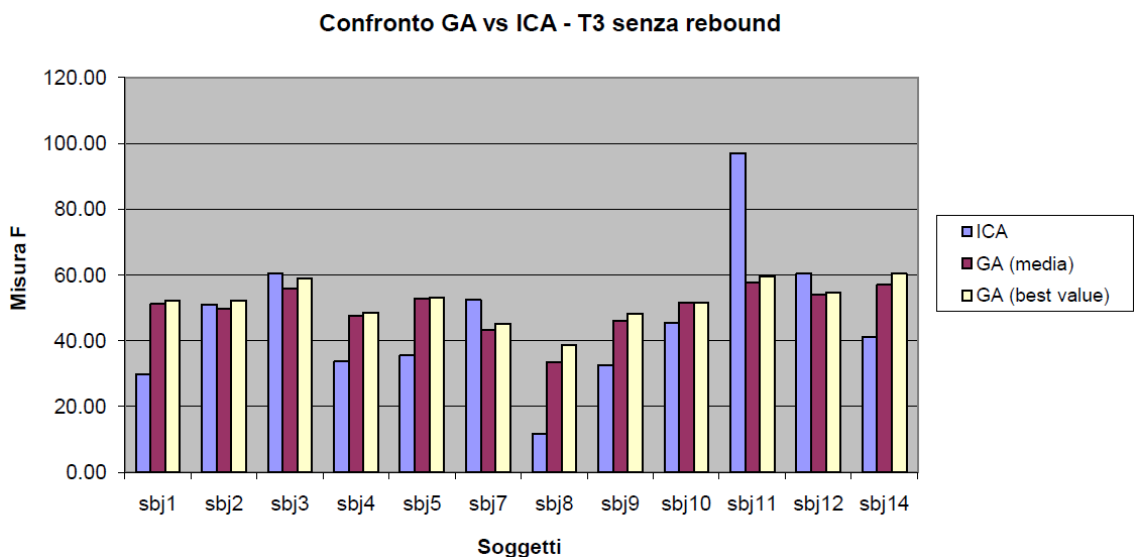
**Figura 48** - Confronto performance GA vs ICA secondo il dataset G1.



**Figura 49** - Confronto performance GA vs ICA secondo il dataset G2.



**Figura 50** - Confronto performance GA vs ICA secondo il dataset T2.

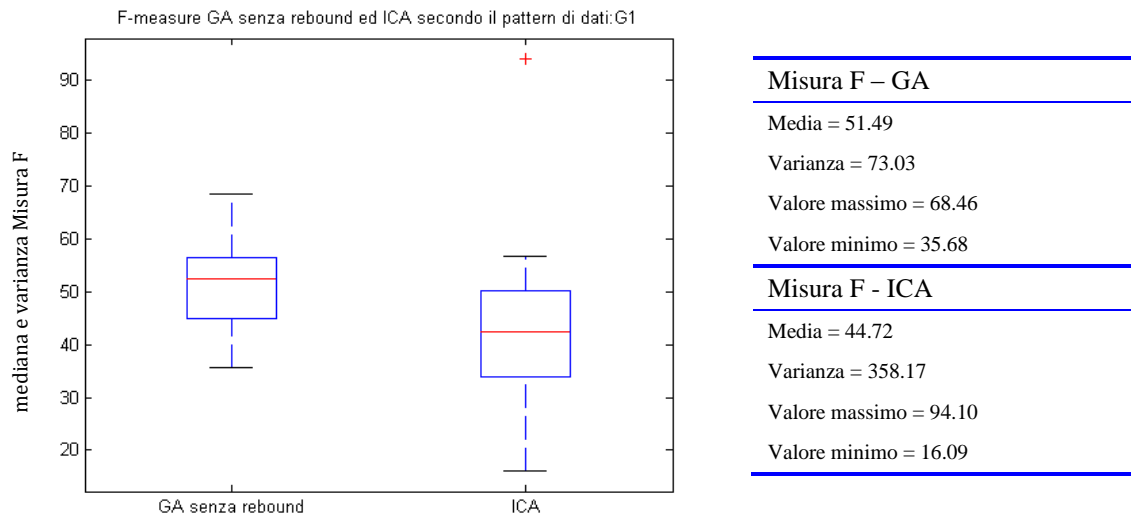


**Figura 51** - Confronto performance GA vs ICA secondo il dataset T3.

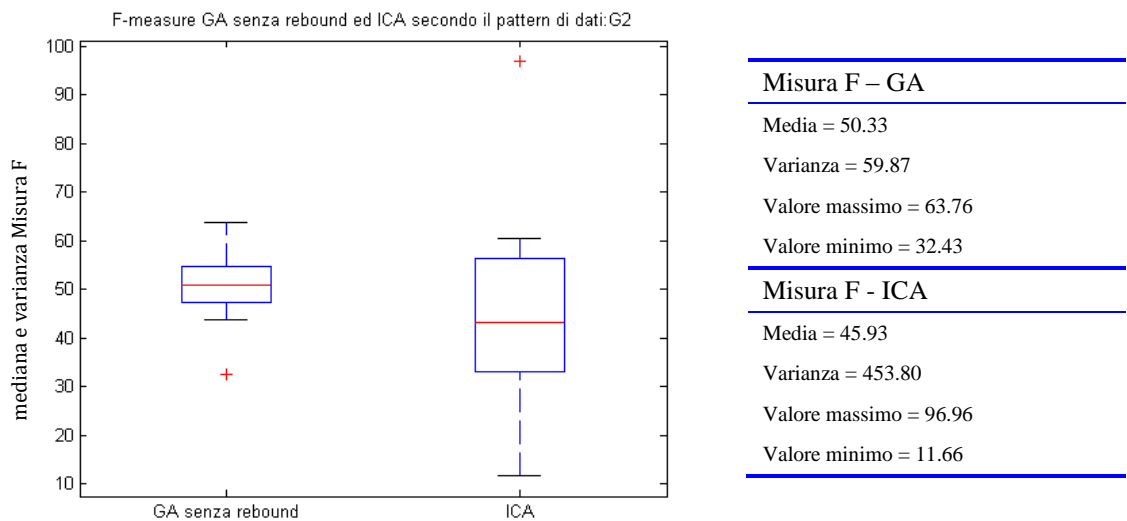
Nel caso dell'interfaccia "Auto", si può notare come le *performance* del nostro sistema si mantengano costanti inter-soggettivamente; ad esempio, si faccia riferimento al confronto tra le Figura 18 e Figura 48: benchè le *performance* siano più basse per l'interfaccia "Auto", l'andamento è perfettamente speculare. Si può osservare come il numero dei soggetti superiori nel confronto sia diminuito per i *dataset* G2 e T3, con 7 soggetti su 12 che hanno migliori prestazioni per il nostro sistema [Figura 49 e Figura 51].

C'è la conferma, per il soggetto sbj11, di *performance* eccellenti da parte del sistema ICA, che però non vengono ripetute per gli altri soggetti.

Di seguito [Figura 52Figura 53Figura 54Figura 55] riportiamo i *boxplot* relativi alle *performance* sui 12 soggetti, confrontando le distribuzioni tra il nostro sistema e il sistema ICA.

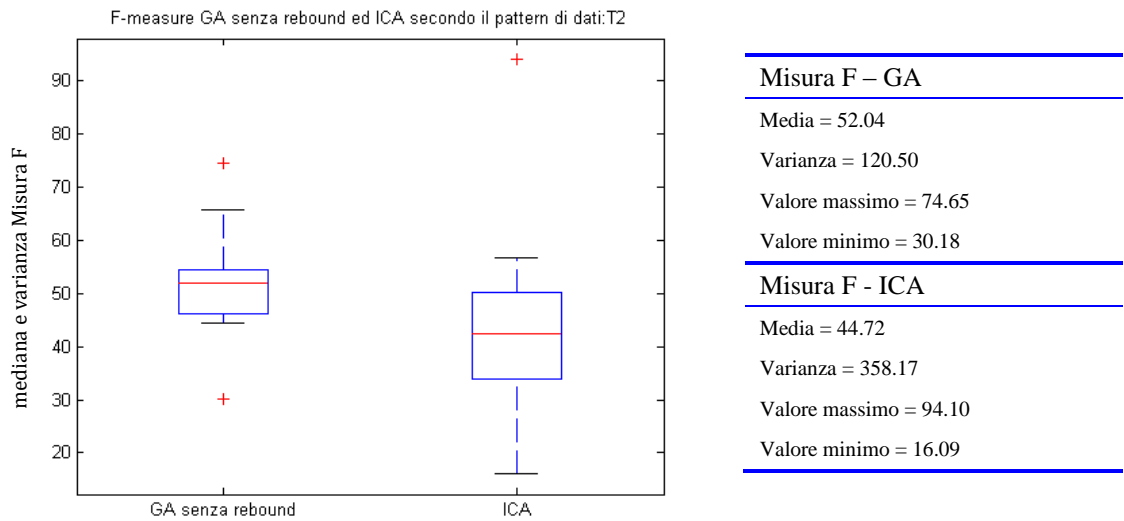


**Figura 52** - *Boxplot del confronto performance GA vs ICA secondo il dataset G1, con relativi valori numerici.*

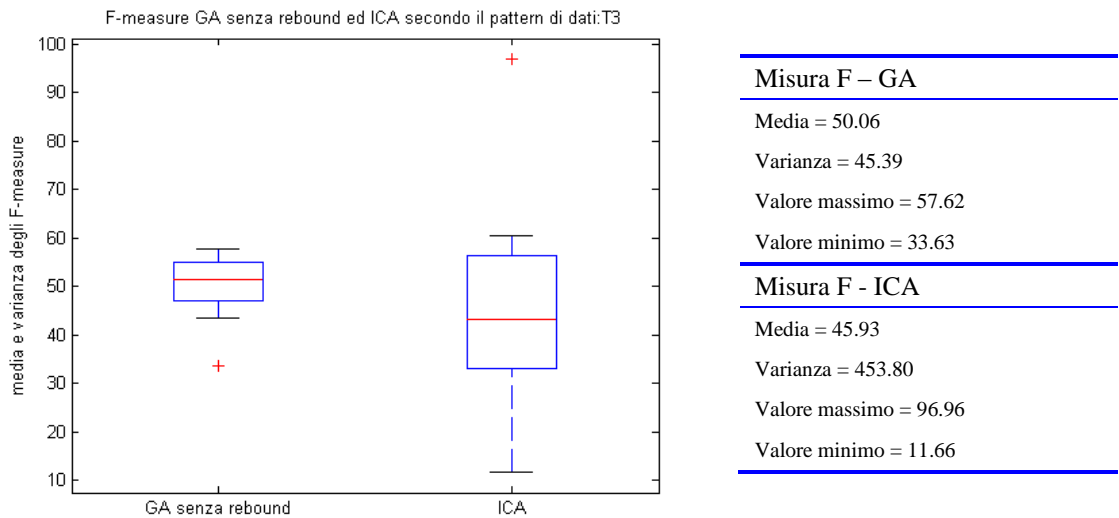


**Figura 53** - *Boxplot del confronto performance GA vs ICA secondo il dataset G2, con relativi valori numerici.*





**Figura 54** - Boxplot del confronto performance GA vs ICA secondo il dataset T2, con relativi valori numerici.

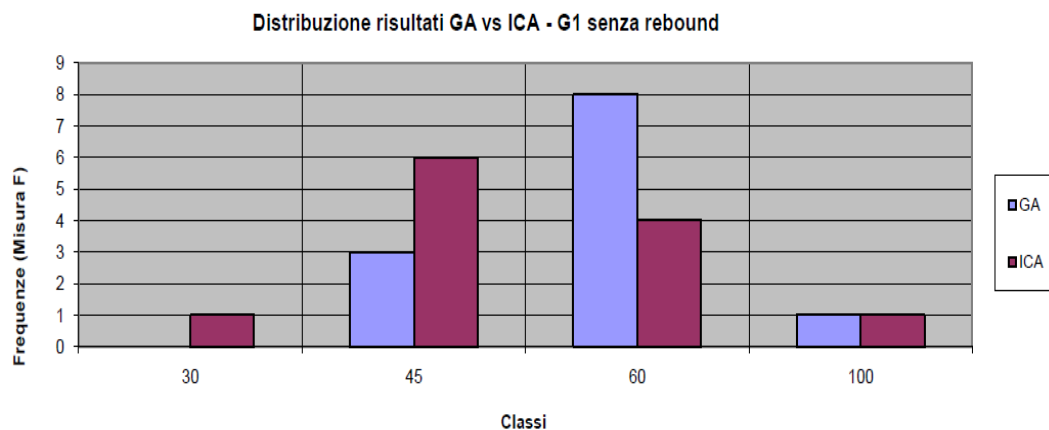


**Figura 55** - Boxplot del confronto performance GA vs ICA secondo il dataset T3, con relativi valori numerici.

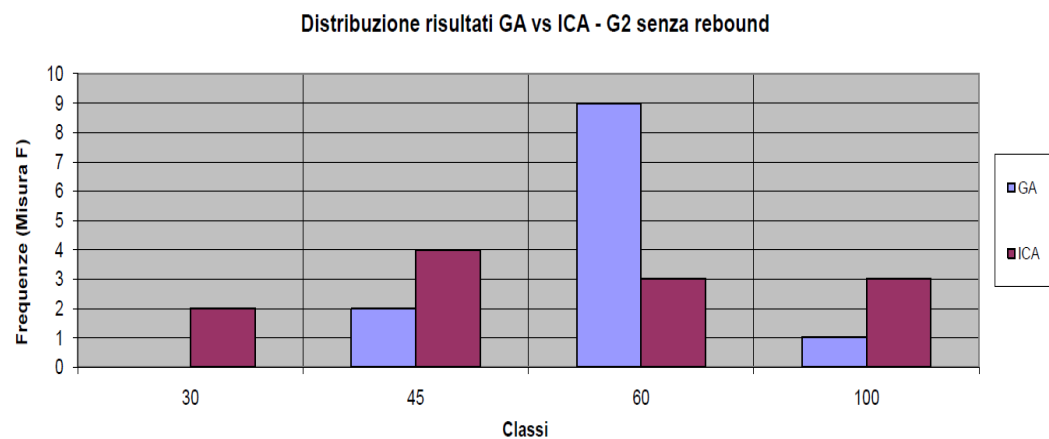
Come conferma dell'osservazione dei grafici precedenti [Figura 48Figura 49Figura 50Figura 51], le differenze di *performance* tra i due sistemi sono diminuite; tuttavia possiamo concludere che la causa non è imputabile a un peggioramento delle prestazioni da parte del nostro sistema, che si mantengono in linea con le prestazioni osservate per l'interfaccia "Arrows", ma a un miglioramento da parte del sistema ICA,

che manifesta però un aumento della varianza dei risultati (già molto elevata per l'interfaccia precedente), soprattutto nel *dataset* con la sessione di *follow-up* come *set* di *testing* [Figura 53Figura 55]; un altro dato interessante in questi *dataset*, è la presenza della particolare *performance* per il sistema ICA sul soggetto sbj11: le *performance* di questo soggetto si mantengono sempre, per tutte le interfacce analizzate, ben al di sopra della media, ma per l'interfaccia "Auto" assumono valori straordinariamente alti (94.10 e 96.96 [Figura 48Figura 49]); ciò è dovuto probabilmente alla particolare forma d'onda dei segnali EEG del soggetto, che in qualche modo facilita la classificazione tra segnali *target* e non *target*; si può osservare come anche il nostro sistema abbia risultati ottimi su questo soggetto, benchè lontani da quelli ottenuti dal sistema ICA.

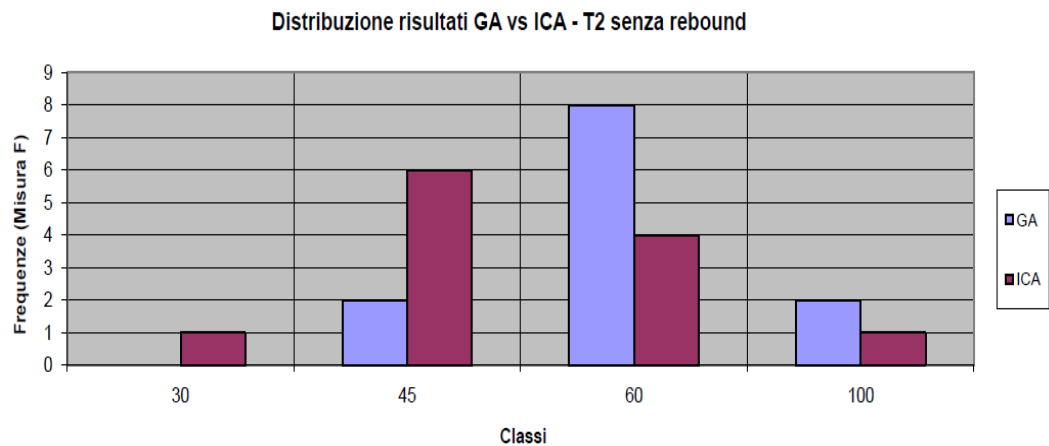
Abbiamo quindi rappresentato i risultati in frequenze relative [Figura 56Figura 57Figura 58 eFigura 59] ed eseguito un test t di Student delle *performance* [Tabella 5].



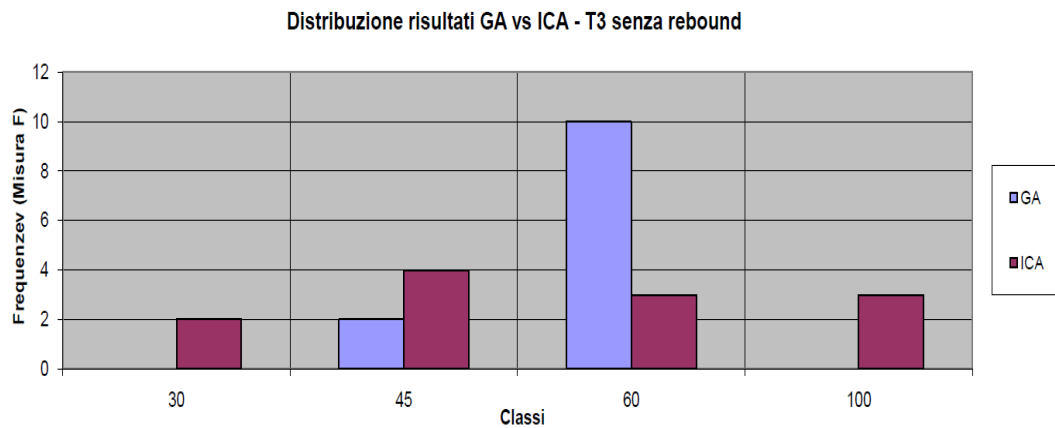
**Figura 56** - Frequenze relative performance GA vs ICA secondo il dataset G1.



**Figura 57** - Frequenze relative performance GA vs ICA secondo il dataset G2.



**Figura 58** - Frequenze relative performance GA vs ICA secondo il dataset T2.



**Figura 59** - Frequenze relative performance GA vs ICA secondo il dataset T3.

GA vs ICA: tipo di <i>dataset</i>	p-value
G1 senza <i>rebound</i>	0.06
G2 senza <i>rebound</i>	0.2
T2 senza <i>rebound</i>	0.04
T3 senza <i>rebound</i>	0.22

**Tabella 5** - Risultati test *t* di Student relativi al confronto performance GA vs ICA per tutti i *dataset*.

In Figura 56 possiamo osservare come per il nostro sistema la distribuzione dei risultati sia ancora una volta una simil-Gaussiana centrata nella classe  $45 < x < 60$ , mentre per il sistema ICA si ha una simil-Gaussiana centrata nella classe  $30 < x < 45$ ; ancora una

volta va sottolineato che il nostro sistema non presenta risultati nella classe  $<30$ , mentre è evidente il miglioramento del sistema ICA rispetto all'interfaccia "Arrows" [Figura 26].

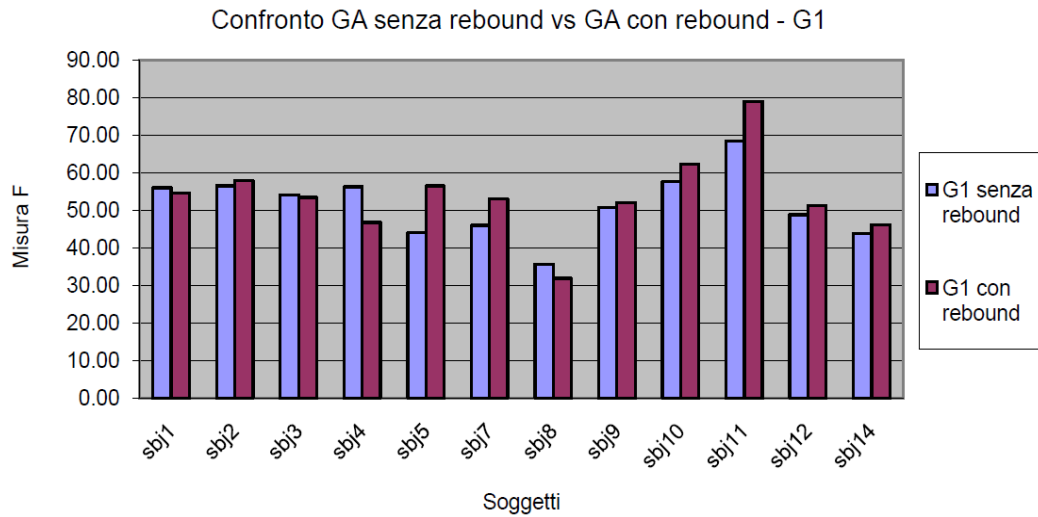
In Figura 57 osserviamo per il nostro sistema una distribuzione simil-Gaussiana, perfettamente bilanciata e centrata nella classe  $45 < x < 60$ , mentre per il sistema ICA osserviamo una distribuzione quasi uniforme, con una omogenea distribuzione dei risultati nelle 4 classi.

Analoghe considerazioni possono essere fatte, rispettivamente, per i grafici nelle Figura 58 e Figura 59; sottolineiamo che, nonostante le prestazioni medie non siano peggiorate [Figura 55], non si riscontra nessun valore nella classe  $>60$ , quindi nessun picco di *performance* [Figura 59].

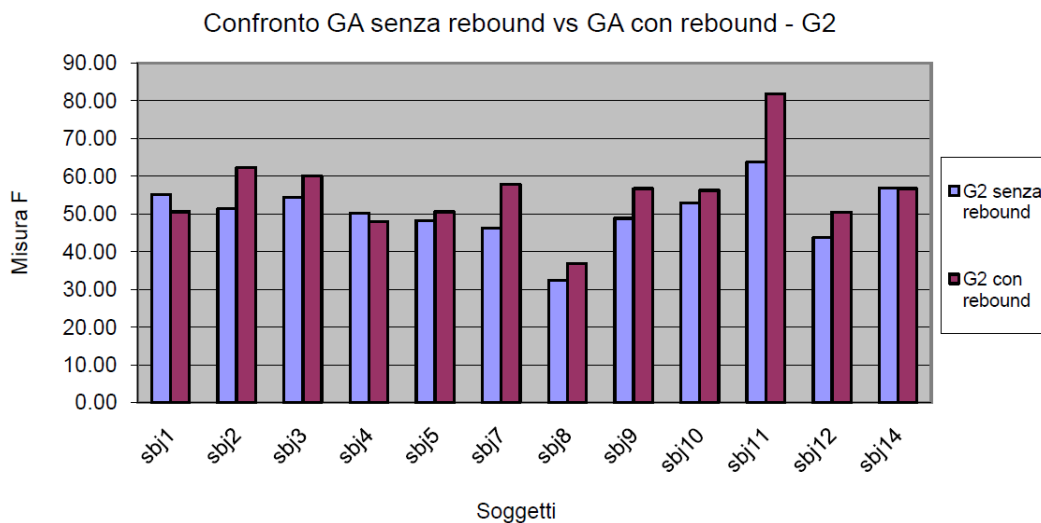
Si può notare in Tabella 5 come le differenze non siano statisticamente significative, a parte il *dataset* T2: il nostro sistema ha *performance* mediamente migliori, ma il miglioramento osservato non è statisticamente significativo, a causa del miglioramento delle *performance* da parte del sistema ICA e della presenza di un dato di *performance* (per il soggetto sbj11) ben oltre le prestazioni medie del sistema.

## 4.2.2 GA senza *rebound* vs GA con *rebound*

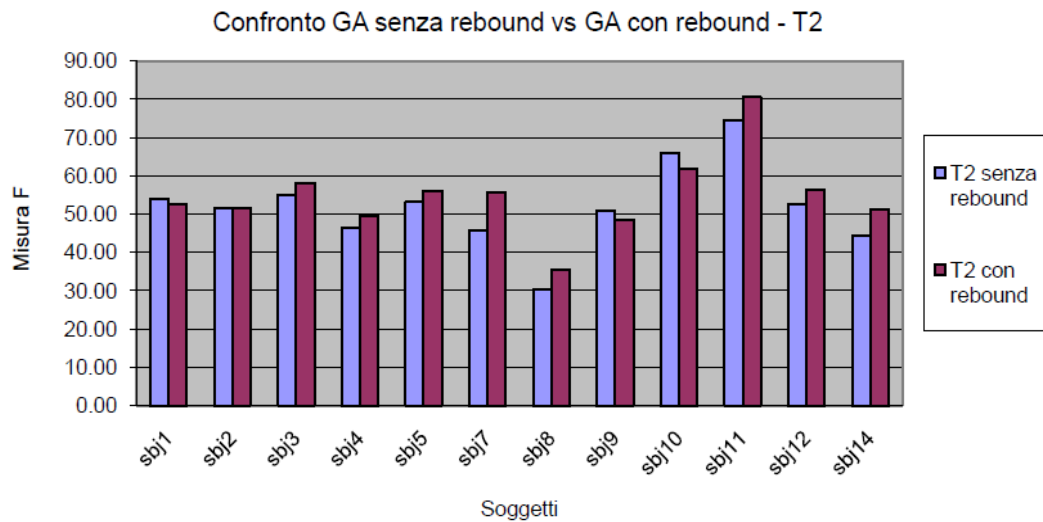
Confrontiamo all'interno dei risultati del nostro sistema le *performance* in presenza o in assenza di *rebound*, per l'interfaccia "Auto".



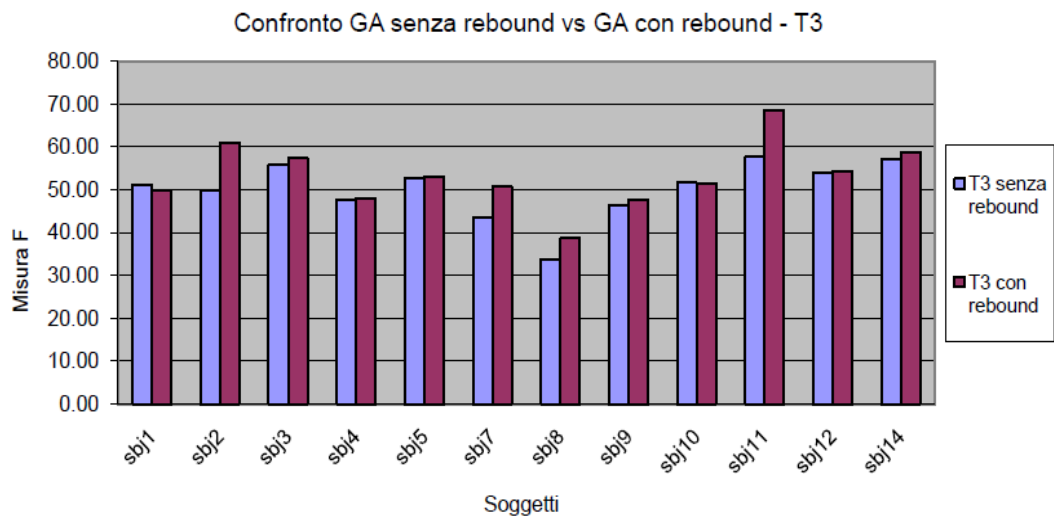
**Figura 60** - Confronto performance GA senza rebound vs GA con rebound, secondo il dataset G1.



**Figura 61** - Confronto performance GA senza rebound vs GA con rebound, secondo il dataset G2.



**Figura 62** - Confronto performance GA senza rebound vs GA con rebound, secondo il dataset T2.



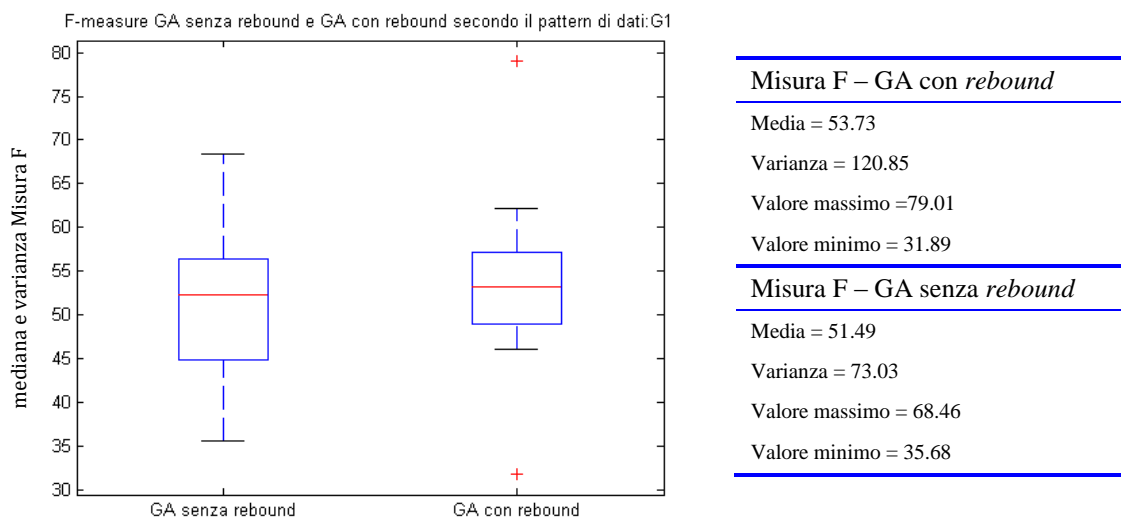
**Figura 63** – Confronto performance GA senza rebound vs GA con rebound, secondo il dataset T3.

In questo caso le *performance* sembrano essere notevolmente migliori: inter-soggettivamente le *performance* sono nella maggior parte dei casi migliori nel caso di analisi con *rebound*.

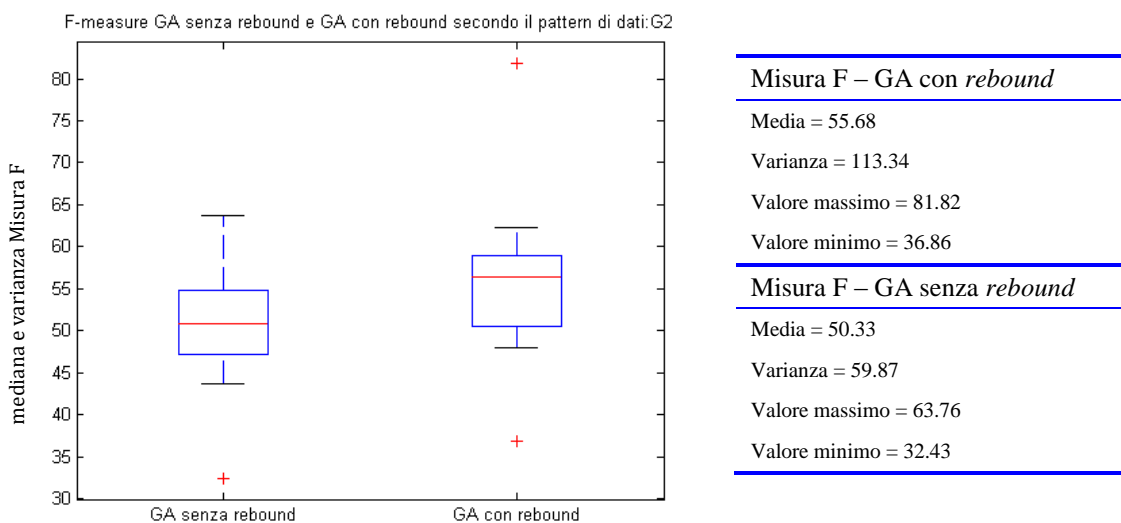
In [Figura 60], relativo al confronto sul *dataset* G1, possiamo osservare un generale miglioramento delle *performance*, con differenze accentuate per i soggetti sbj5 e sbj11; per quest'ultimo soggetto si raggiungono *performance* quasi in linea con le *performance* espresse dal sistema ICA [Figura 48], raggiungendo un valore di misura F-bilanciata di quasi 80. Il sistema con *rebound* risulta superiore per 8 soggetti su 12, e in alcuni casi come detto con un netto miglioramento nelle *performance*.

In [Figura 61], relativo al confronto sul *dataset* G2, possiamo osservare un miglioramento ancora più accentuato (si osservino i soggetti *sbj2* *sbj7* e *sbj11*); in particolare nel caso del soggetto *sbj11* le prestazioni sono ancora maggiori, superando il valore di misura F-bilanciata di 80, mentre il sistema senza *rebound* supera di poco il valore di 60. Il sistema con *rebound* risulta superiore nel confronto per 9 soggetti su 12.

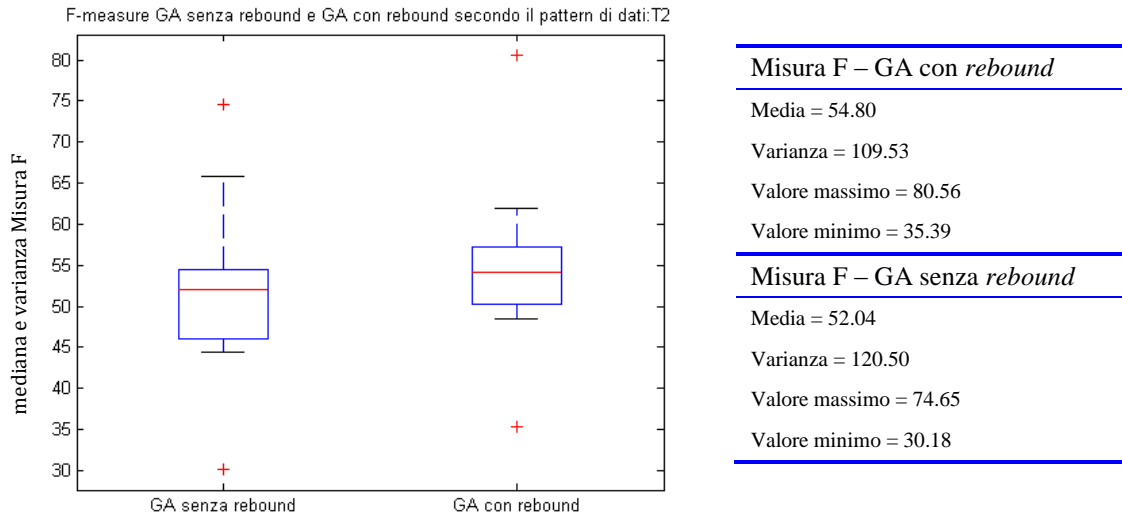
Per i *dataset* T2 e T3 [Figura 62Figura 63] valgono le stesse considerazioni fatte, con un riferimento particolare alle prestazioni del sistema con *rebound* per il *dataset* T3, che risulta superiore per 10 soggetti su 12.



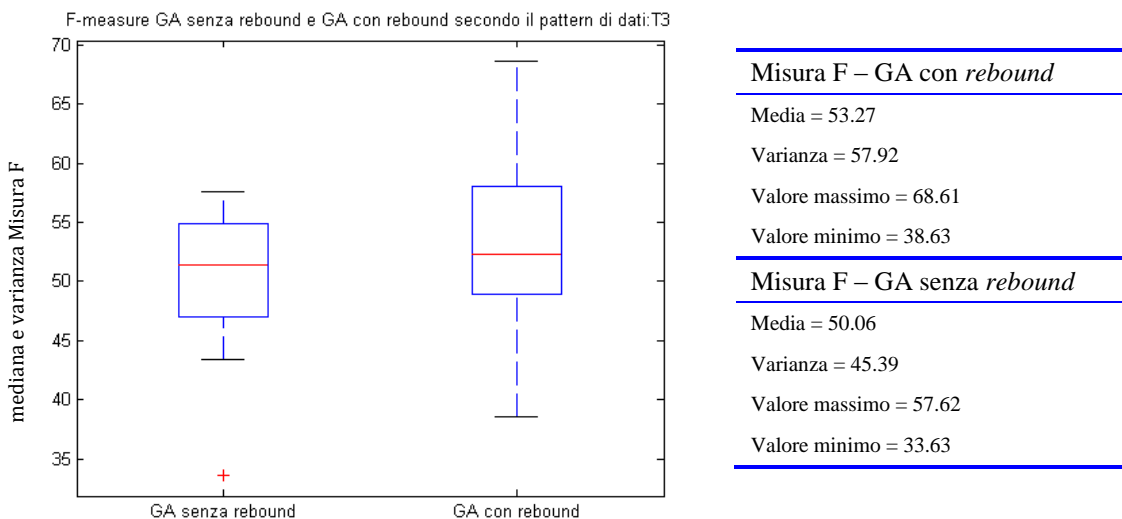
**Figura 64** - Boxplot del confronto performance GA con *rebound* vs GA senza *rebound*, secondo il *dataset* G1, con relativi valori numerici.



**Figura 65** - Boxplot del confronto performance GA con *rebound* vs GA senza *rebound*, secondo il *dataset* G2, con relativi valori numerici.



**Figura 66** - Boxplot del confronto performance GA con rebound vs GA senza rebound, secondo il dataset T2, con relativi valori numerici.

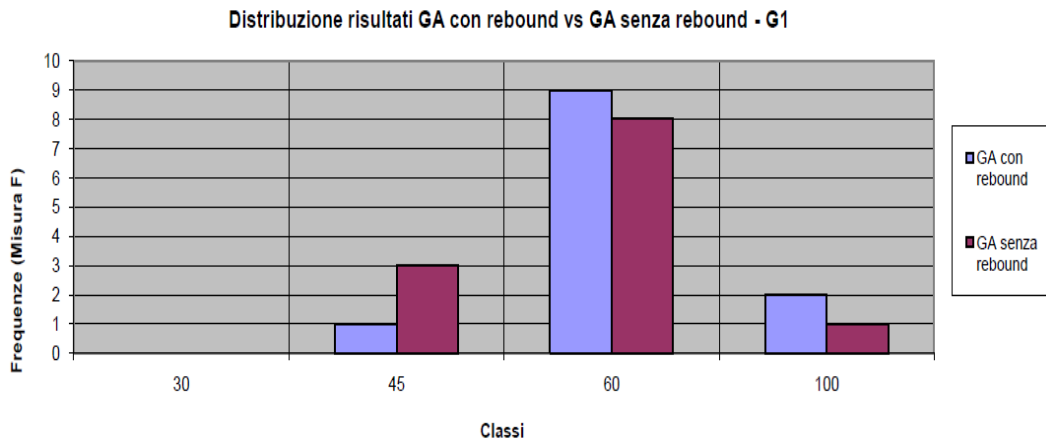


**Figura 67** - Boxplot del confronto performance GA con rebound vs GA senza rebound, secondo il dataset T3, con relativi valori numerici.

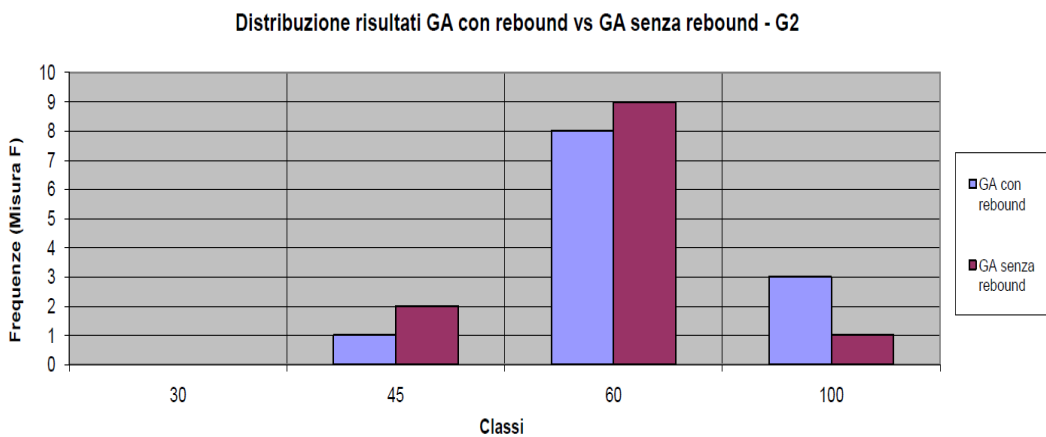
Possiamo notare come in tutti i *dataset* ci sia stato un miglioramento in media e nei valori massimi e minimi; inoltre, vengono per la prima volta raggiunte *performance* paragonabili ai picchi di *performance* del sistema ICA; i miglioramenti più evidenti si hanno per i *dataset* G2 e T3 [Figura 65Figura 67], interessante il picco di prestazione nel *dataset* G2 [Figura 65], dove il sistema con *rebound* raggiunge una *performance* di 81.82.



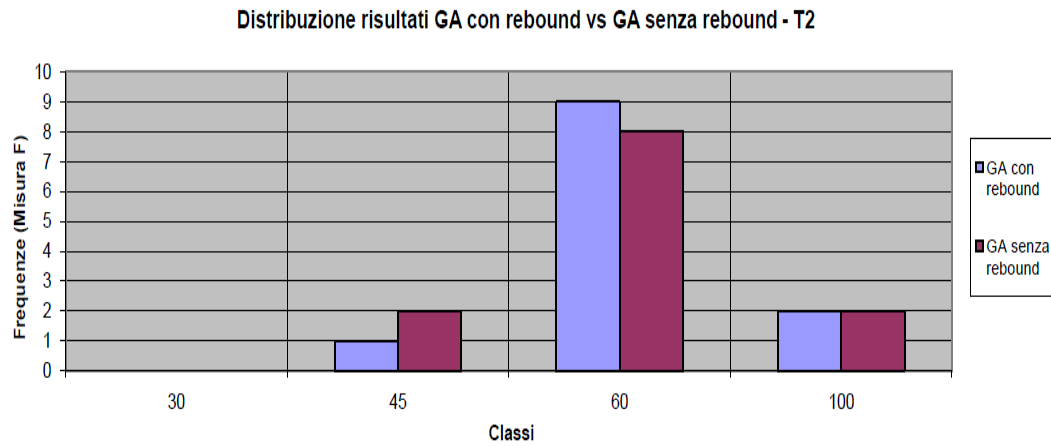
Passiamo ad analizzare la distribuzione secondo classi di *performance* e i test t di Student.



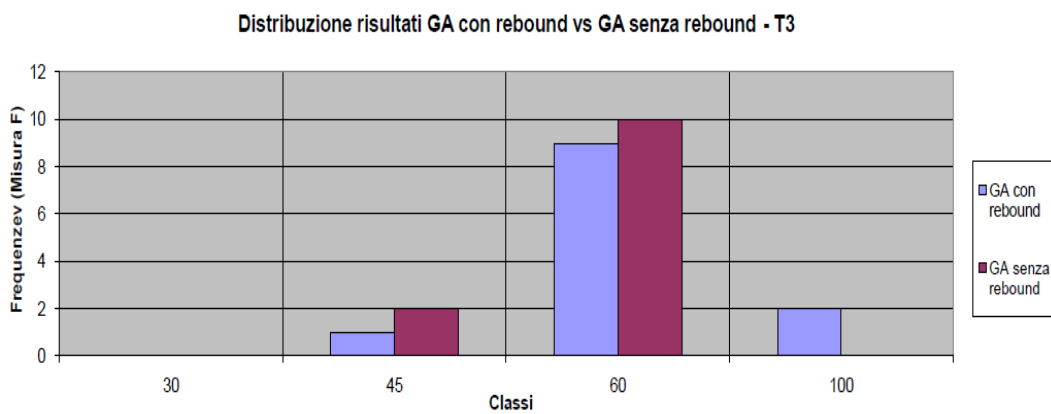
**Figura 68** - Frequenze relative performance GA con rebound vs GA, senza rebound, secondo il dataset G1.



**Figura 69** - Frequenze relative performance GA con rebound vs GA, senza rebound, secondo il dataset G2.



**Figura 70** - Frequenze relative performance GA con rebound vs GA, senza rebound, secondo il dataset T2.



**Figura 71** - Frequenze relative performance GA con rebound vs GA, senza rebound, secondo il dataset T3.

GA con rebound vs GA senza rebound: tipo di dataset	p-value
G1	0.11
G2	0.01
T2	0.02
T3	0.01

**Tabella 6**- Risultati test t di Student relativi al confronto performance GA con rebound vs GA, senza rebound, per tutti i dataset.

Anche per l'interfaccia "Auto" le distribuzioni tra le classi, restano quasi invariate: l'unico miglioramento evidente si ha nella classe >60, che aumenta di numerosità in tutti i *dataset* [Figura 68, Figura 69, Figura 70 e Figura 71]; un dato interessante è per il *dataset* T3 [Figura 71], dove a fronte della totale assenza di risultati nella classe >60 per il sistema senza *rebound*, si hanno 2 risultati nella stessa classe per il sistema con *rebound*.

Si può notare come anche in questo caso le differenze siano statisticamente significative per quasi tutti i *dataset*, escluso il *dataset* G1, dove in effetti avevamo individuato un miglioramento più contenuto nelle prestazioni.

### 4.2.3 Confronto *dataset*

Anche nel caso dell'interfaccia "Auto", le *performance* non mostrano differenze statisticamente significative modificando il *set* di *testing* [Tabella 7 Tabella 8].

Confronto <i>dataset</i> ; GA senza <i>rebound</i>	p-value
G1 vs G2	0.11
T2 vs T3	0.20

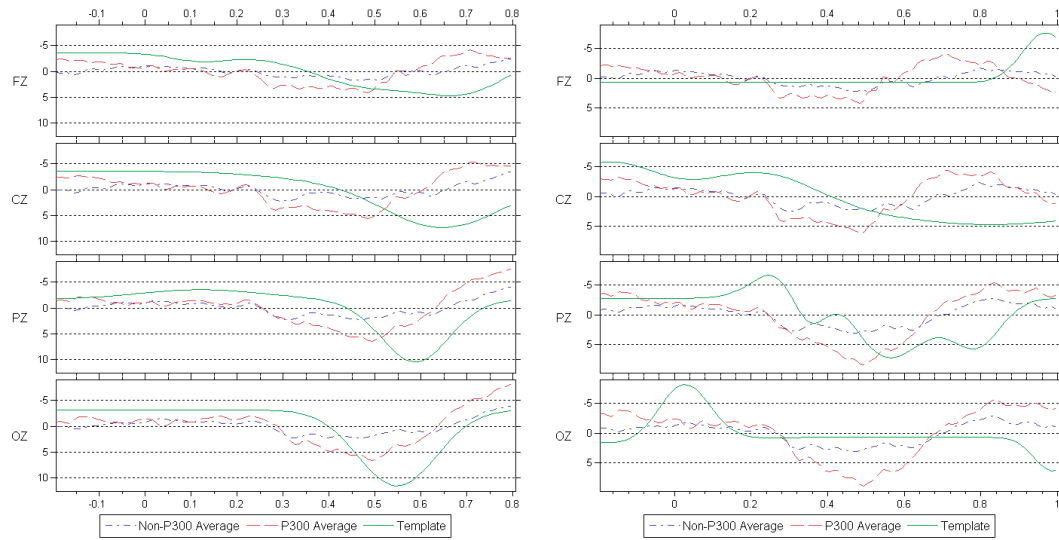
**Tabella 7** - Risultati test *t* di Student relativi al confronto *dataset* per GA, senza *rebound*.

Confronto <i>dataset</i> ; GA con <i>rebound</i>	p-value
G1 vs G2	0.23
T2 vs T3	0.21

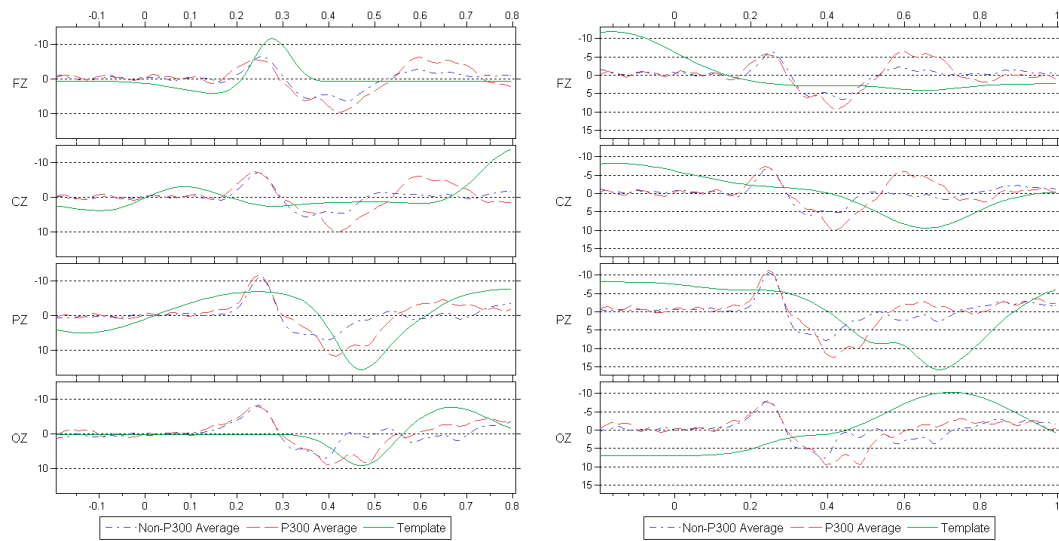
**Tabella 8** - Risultati test *t* di Student relativi al confronto *dataset* per GA, con *rebound*.

Il sistema ICA in questo caso sembra manifestare capacità di generalizzazione (nel confronto G1 vs G2 il *p-value* è pari a 0.32), anche se restano valide (anzi si rafforzano) gli stessi dubbi espressi per l'interfaccia "Arrows": l'alta varianza dei risultati e un evidente *outlier*, il risultato per il soggetto sbj11, impediscono di trarre conclusioni in merito.

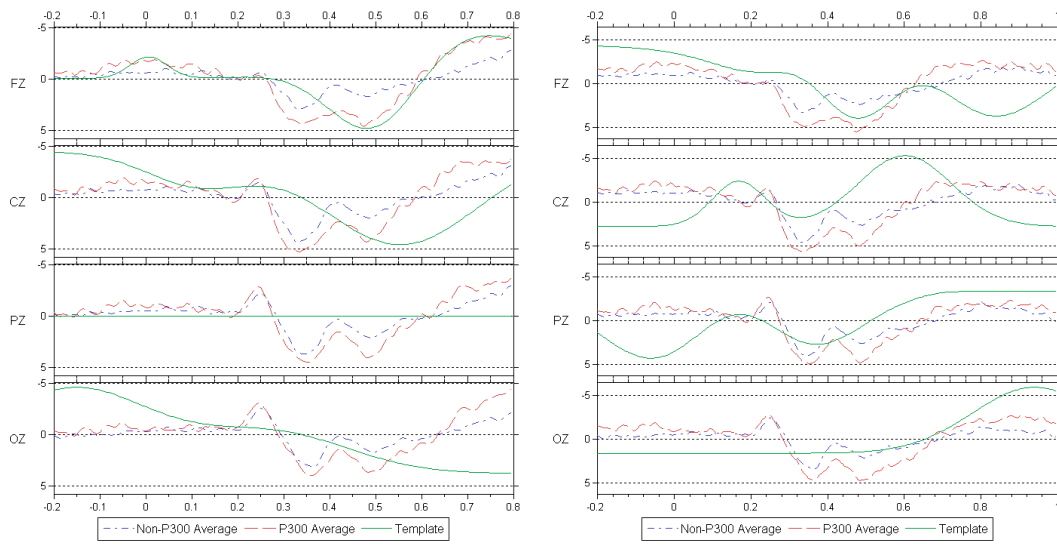
## 4.2.4 Template



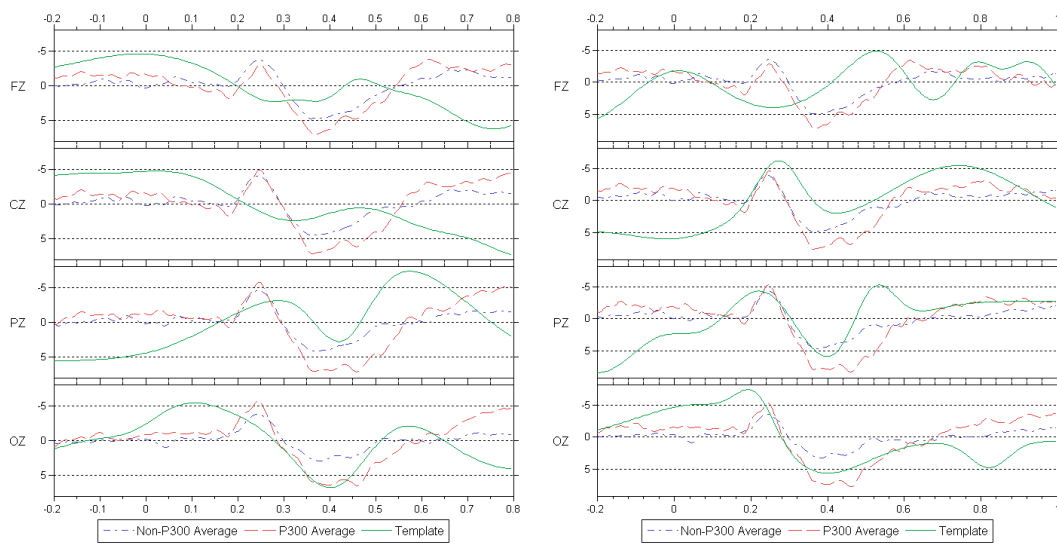
**Figura 72** - Template estratto per il soggetto sbj02 sul dataset G1 (senza rebound a sinistra e con rebound a destra).



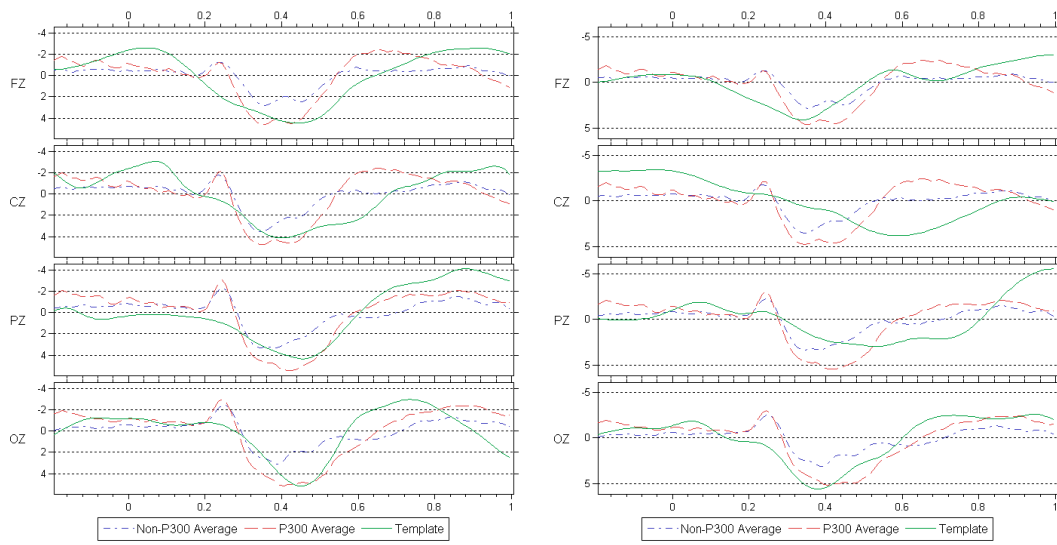
**Figura 73** - Template estratto per il soggetto sbj10 sul dataset G2 (senza rebound a sinistra e con rebound a destra).



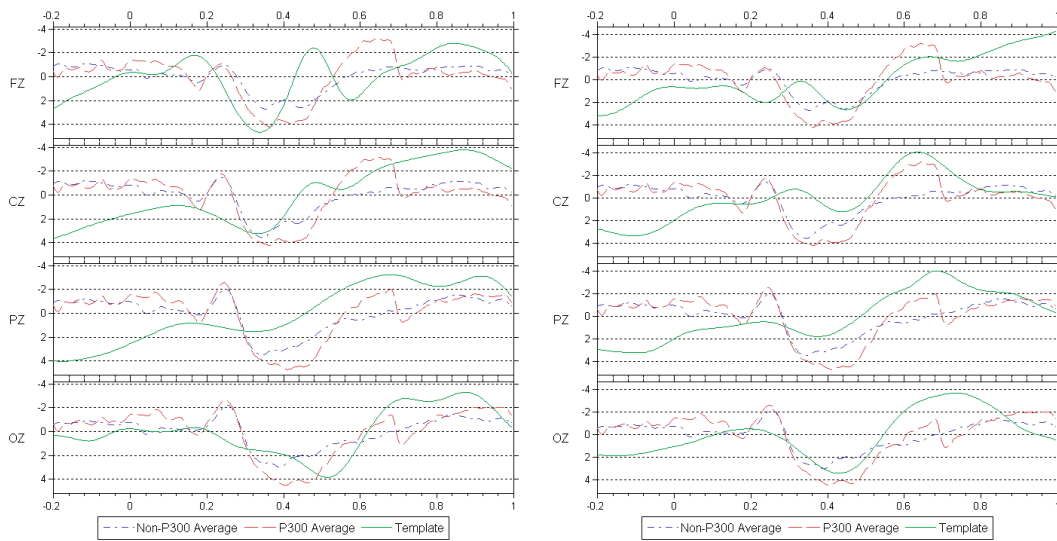
**Figura 74** - *Template* estratto per il soggetto *sbj12* sul dataset *T2* (senza rebound a sinistra e con rebound a destra).



**Figura 75** - *Template* estratto per il soggetto *sbj14* sul dataset *T3* (senza rebound a sinistra e con rebound a destra).



**Figura 76** - *Template medi sui dataset G1 (sinistra) e G2 (destra), con rebound.*

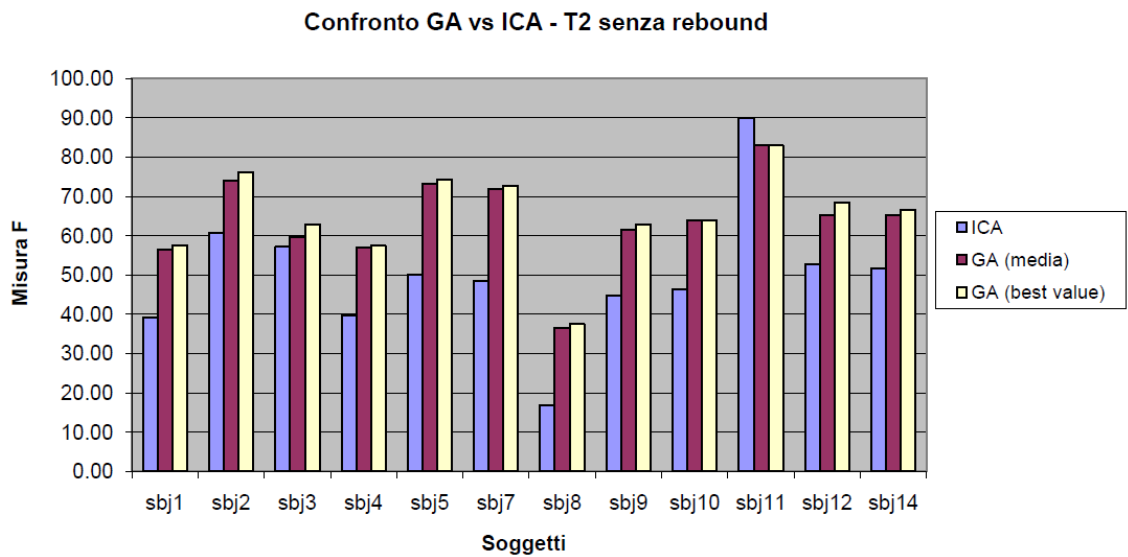


**Figura 77** - *Template medi sui dataset T2 (sinistra) e T3 (destra), con rebound.*

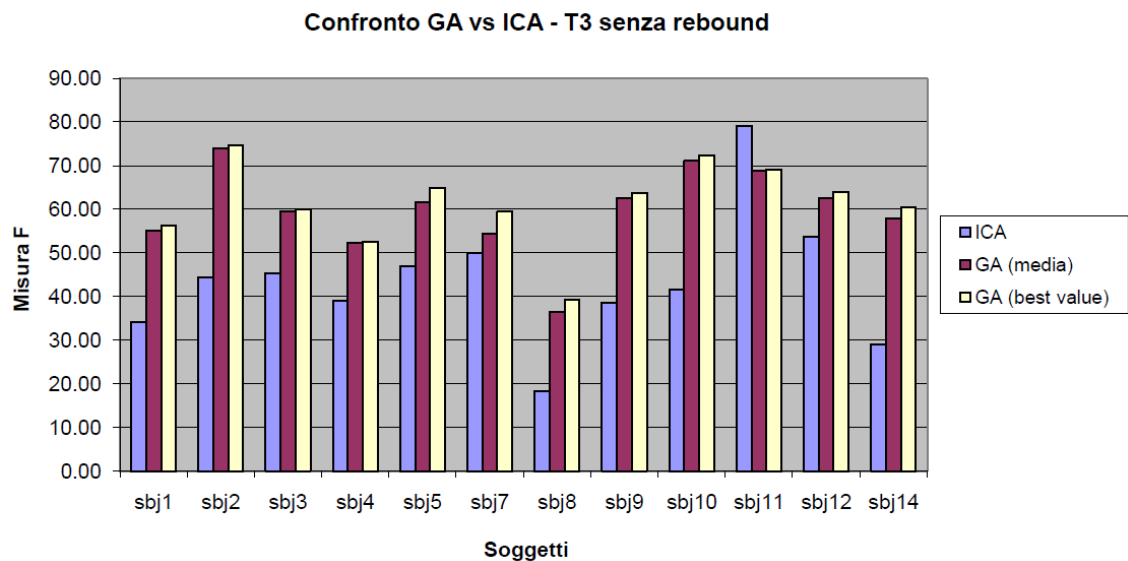
## 4.3 Interfaccia “Vol”

### 4.3.1 GA vs ICA

Per questa interfaccia abbiamo solo i risultati per quanto riguarda i *dataset* T2 e T3; iniziamo con il confrontare le *performance* del nostro sistema e le *performance* del sistema ICA.



**Figura 78** - Confronto performance GA vs ICA secondo il dataset T2.

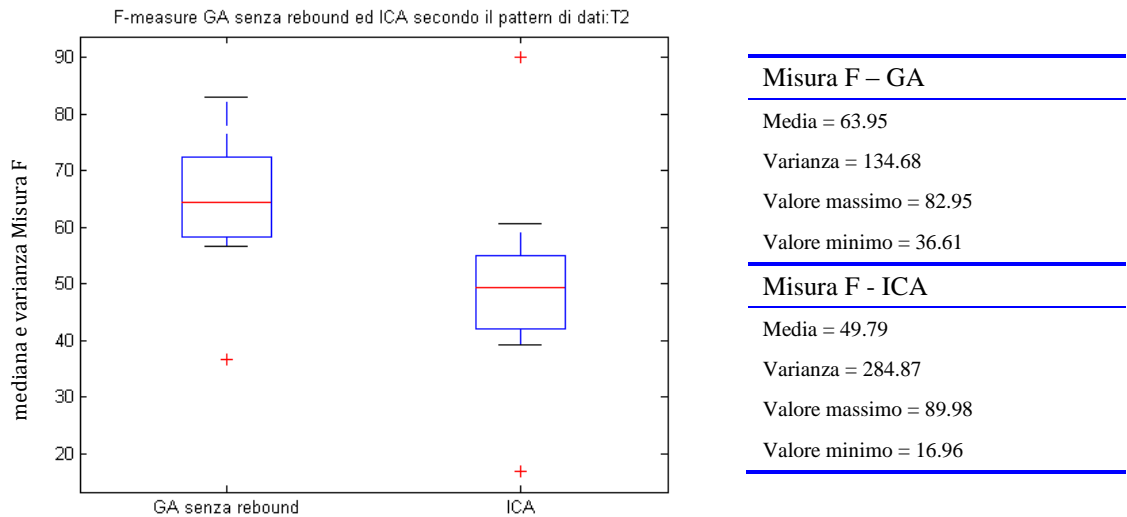


**Figura 79** - Confronto performance GA vs ICA secondo il dataset T3.

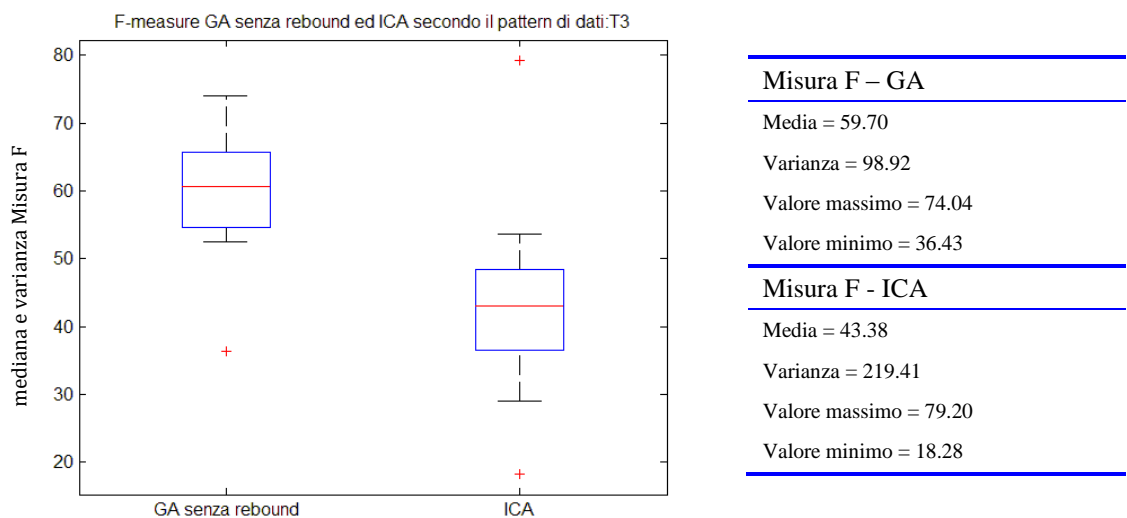


Nel caso dell'interfaccia "Vol", possiamo notare qualitativamente che le *performance* del nostro sistema sono nettamente migliori sia rispetto al sistema ICA [Figura 78Figura 79], sia rispetto alle interfacce precedenti; questo conferma le ipotesi elaborate dall'équipe di ricerca del S. Camillo, ipotesi introdotte nel Capitolo 3.

Di seguito riportiamo i *boxplot* relativi alle *performance* sui 12 soggetti, confrontando le distribuzioni tra il nostro sistema e il sistema ICA.

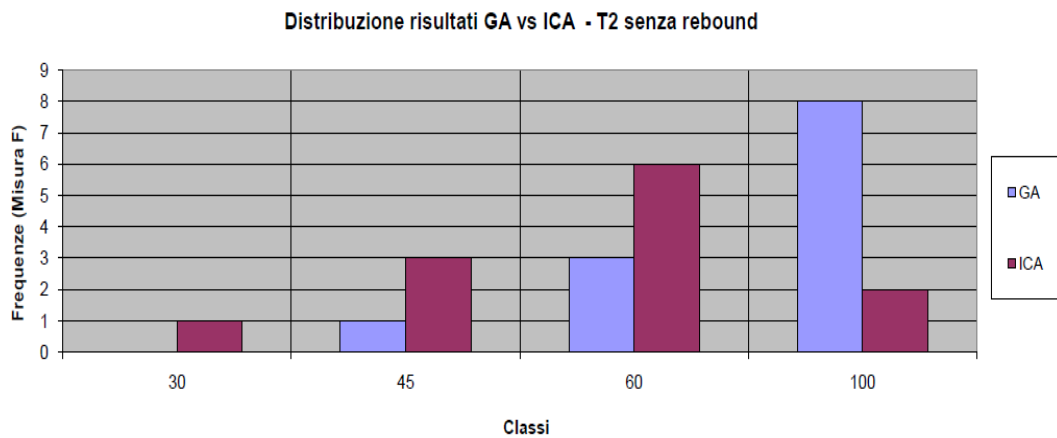


**Figura 80** - Boxplot del confronto performance GA vs ICA secondo il dataset T2, con relativi valori numerici.

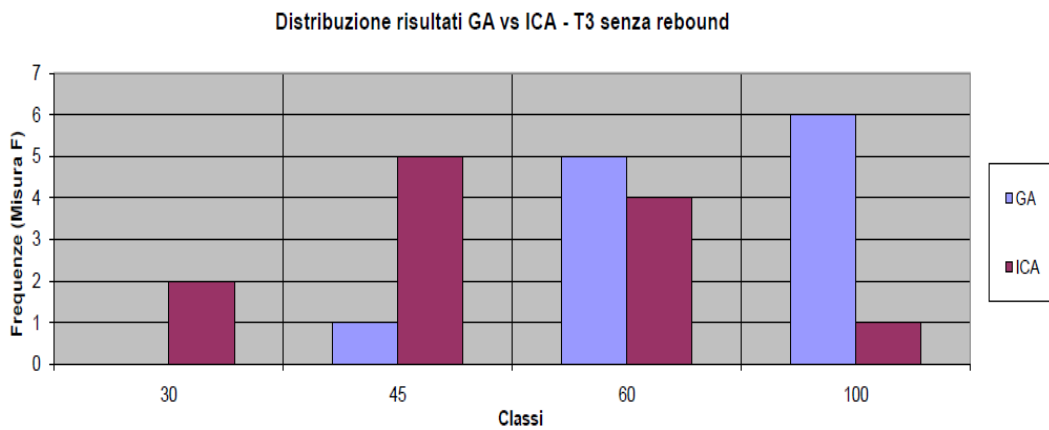


**Figura 81** - Boxplot del confronto performance GA vs ICA secondo il dataset T3, con relativi valori numerici.

Come conferma dell'osservazione dei grafici precedenti [Figure 80 e 81], notiamo come le differenze di *performance* tra i due sistemi siano notevolmente aumentate, per merito di un netto miglioramento da parte del nostro sistema, che sembra avere dei vantaggi quando viene usata l'interfaccia "Vol": le medie osservate nelle Figura 80 e Figura 81 sono le più alte registrate in questo lavoro di tesi per quanto riguarda i *dataset* senza *rebound*; il valore massimo per il *dataset* T2 [Figura 80] è il risultato migliore registrato dal nostro sistema.



**Figura 82** - Frequenze relative performance GA vs ICA secondo il dataset T2.



**Figura 83** - Frequenze relative performance GA vs ICA secondo il dataset T3.

GA vs ICA: tipo di <i>dataset</i>	p-value
T2 senza <i>rebound</i>	0.0001
T3 senza <i>rebound</i>	0.0003

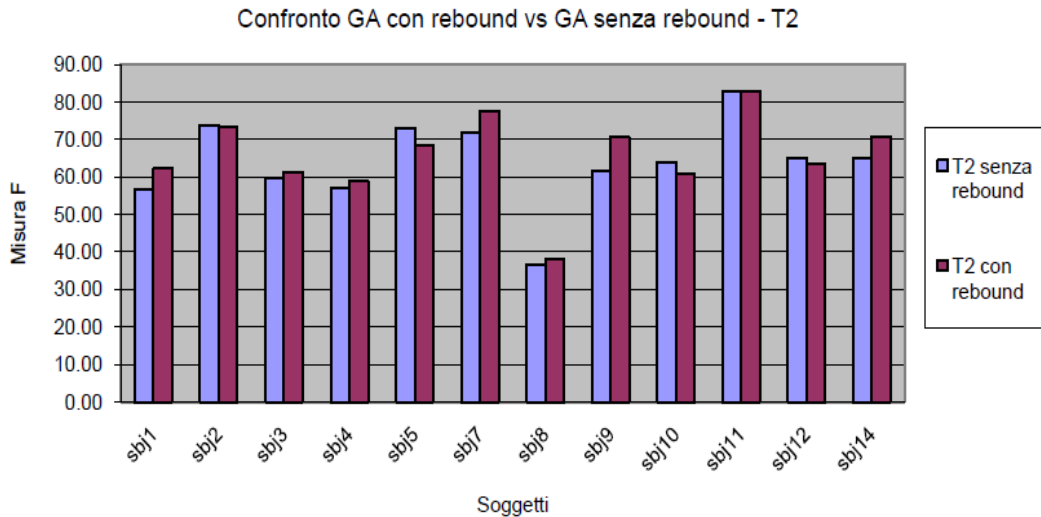
**Tabella 9-** Risultati test *t* di Student relativi al confronto performance GA vs ICA per i *dataset* T2 e T3, senza *rebound*.

Osservando le distribuzioni dei risultati nelle **Errore. L'origine riferimento non è stata trovata.**, possiamo notare che la classe con la maggiore numerosità per il nostro sistema è la classe >60: per la maggior parte dei soggetti si registrano quindi prestazioni eccellenti (8 su 12 per il *dataset* T2 e 6 su 12 per il *dataset* T3), e a parte il soggetto sbj8, tutti i soggetti superano la soglia di 50 come misura F-bilanciata [Figura 78Figura 79].

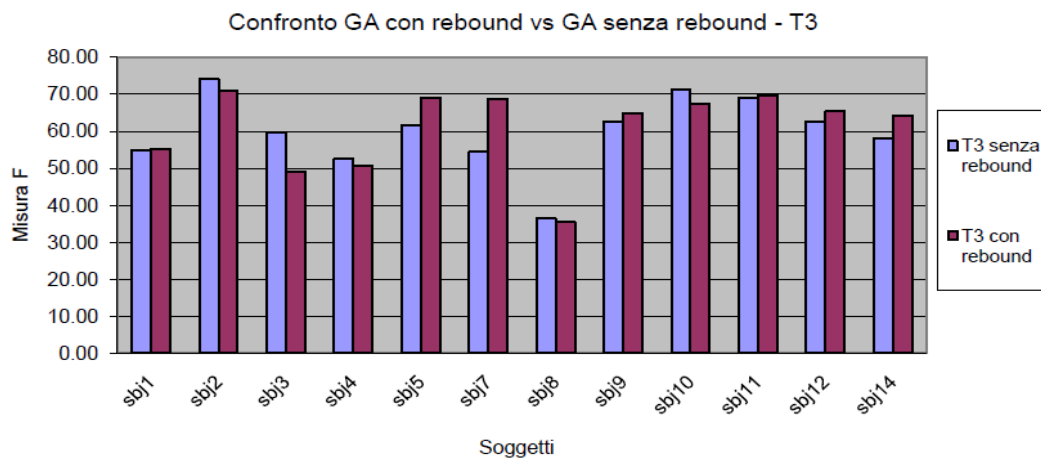
In Tabella 9 si può notare come le differenze siano statisticamente significative per tutte i *dataset*; viene confermata ancora una volta l'osservazione iniziale, secondo cui l'interfaccia "Vol" avvantaggia il nostro sistema, che ottiene prestazioni molto migliori rispetto al sistema ICA.

### 4.3.2 GA senza *rebound* vs GA con *rebound*

Confrontiamo all'interno dei risultati del nostro sistema le *performance* in presenza o in assenza di *rebound*, per l'interfaccia "Vol".



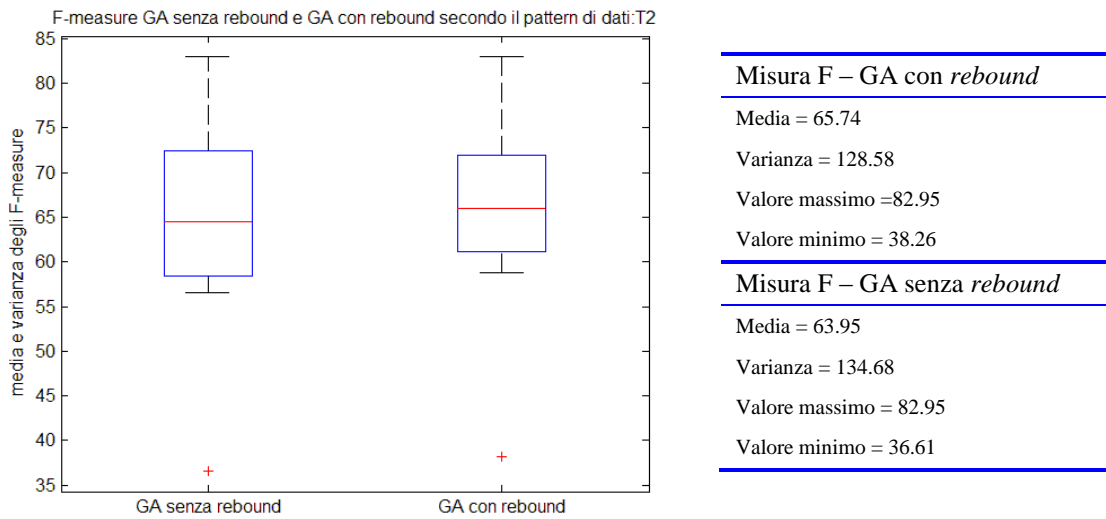
**Figura 84** - Confronto performance GA senza rebound vs GA con rebound, secondo il dataset T2.



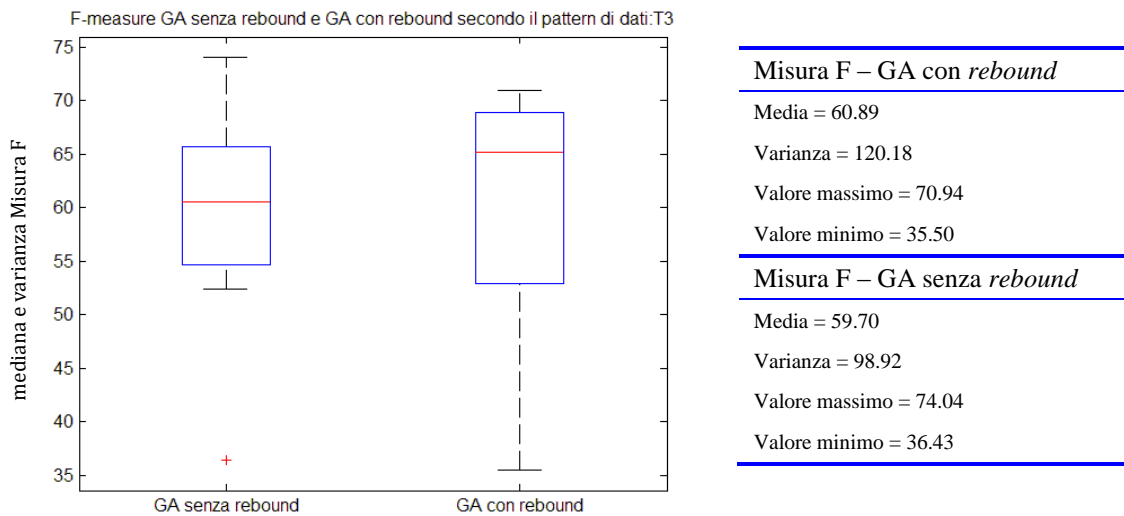
**Figura 85** – Confronto performance GA senza rebound vs GA con rebound, secondo il dataset T3.

Analizzando qualitativamente le *performance* [Figura 84Figura 85], non sembra esserci un netto miglioramento nel caso con *rebound*, a parte un paio di soggetti per il *dataset* T3 [Figura 85]: per molti soggetti le *performance* sono migliori nel caso di analisi senza *rebound*; si registra infatti che il sistema con *rebound* risulta superiore in 7 soggetti su 12 per il *dataset* T2 [Figura 84], e solo nella metà dei casi per il *dataset* T3 [Figura 855].

Osservando i dati statistici e i *boxplot* [Figura 86Figura 87], questo dato appare più evidente.



**Figura 86** - Boxplot del confronto performance GA con rebound vs GA senza rebound, secondo il dataset T2, con relativi valori numerici.

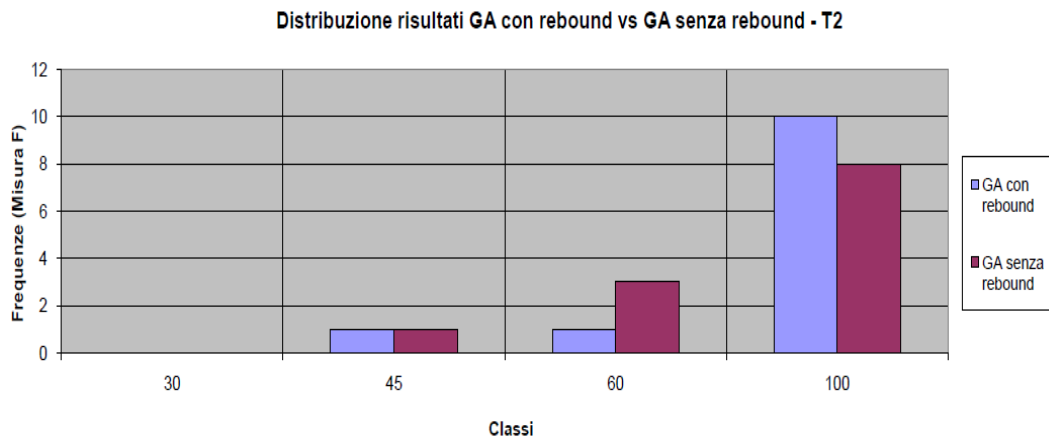


**Figura 87** - Boxplot del confronto performance GA con rebound vs GA senza rebound, secondo il dataset T3, con relativi valori numerici.

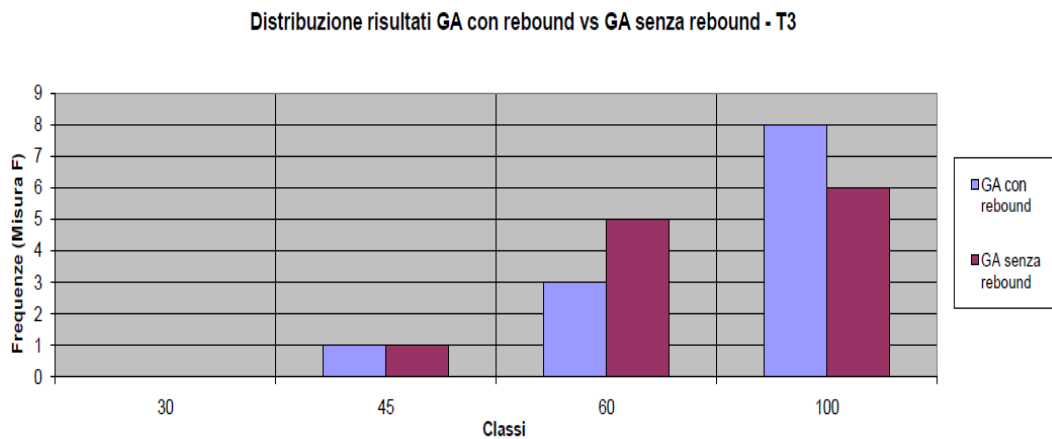
In questo caso il miglioramento, benchè sia comunque presente, risulta molto limitato [Figura 86Figura 87]; il motivo potrebbe risiedere nel fatto che l'algoritmo raggiunge soluzioni molto vicine all'ottimo, e un ulteriore contenuto informativo sembra ridondante rispetto alle informazioni estrapolate dal confronto dei segnali *target* e non *target*. Suggeriamo comunque di ripetere le analisi includendo una porzione di

segnale più ampia: il *dataset* T2 con *rebound* [Figura 86] ha le migliori prestazioni di tutto il lavoro di tesi.

Passiamo ad analizzare la distribuzione secondo classi di *performance* ed i test t di Student [Figura 88Figura 89].



**Figura 88** - Frequenze relative performance GA con rebound vs GA, senza rebound, secondo il dataset T2.



**Figura 89** - Frequenze relative performance GA con rebound vs GA, senza rebound, secondo il dataset T3.

GA con <i>rebound</i> vs GA senza <i>rebound</i> : tipo di <i>dataset</i>	p-value
T2	0.08
T3	0.26

**Tabella 10-** Risultati test t di Student relativi al confronto performance GA con rebound vs GA, senza rebound, per tutti i dataset.

Nonostante non c'è un grande miglioramento in media, possiamo osservare che la numerosità nella classe  $>60$  è ulteriormente aumentata: per ben 10 soggetti su 12 per il *dataset* T2 si sono registrate prestazioni superiori a 60 [Figura 88], risultando anche in questo caso l'esecuzione con le prestazioni migliori di tutto il lavoro di tesi.

In Tabella 10- ***Risultati test t di Student relativi al confronto performance GA con rebound vs GA, senza rebound, per tutti i dataset.*** si può notare come in questo caso le differenze non siano statisticamente significative; come detto ciò può essere dovuto al fatto che nel nostro sistema le *performance* relative ai *dataset* dell'interfaccia "Vol" raggiungono un ottimo di classificazione prima di includere il *rebound*.

### 4.3.3 Confronto *dataset*

Per testare la robustezza della classificazione, abbiamo confrontato i *dataset* T2 vs. T3.

Confronto <i>dataset</i>	p-value
T2 vs T3 (senza <i>rebound</i> )	0.03
T2 vs T3 (con <i>rebound</i> )	0.01

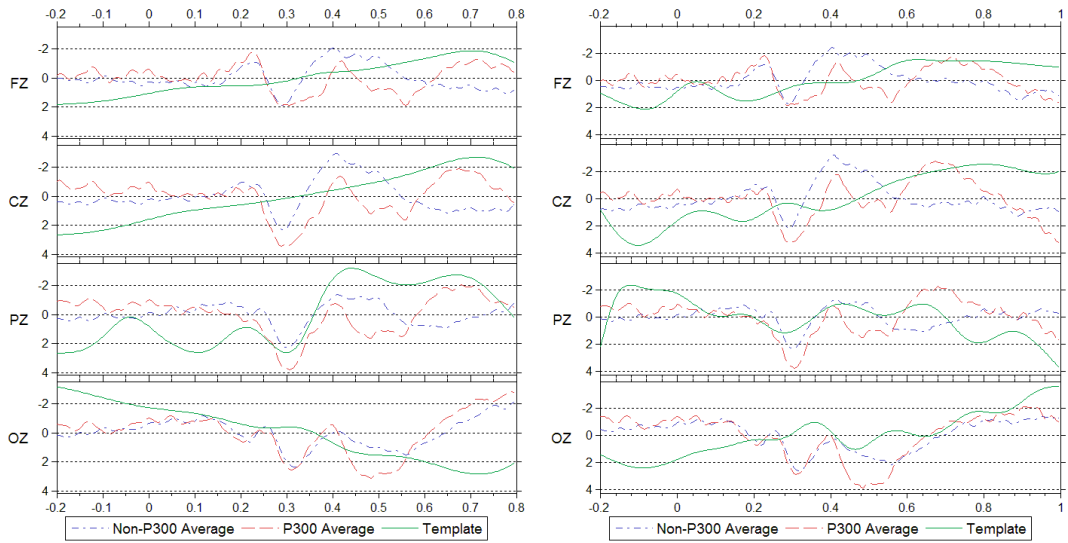
**Tabella 11** - Risultati test *t* di Student relativi al confronto *dataset* per GA.

Nel caso dell'interfaccia “Vol”, le *performance* hanno differenze statisticamente significative modificando il *set* di *testing* [Tabella 11 - **Risultati test *t* di Student relativi al confronto *dataset* per GA.**]; questo potrebbe dipendere dalle differenze delle risposte soggettive nel lungo termine, esaltando le differenze dei segnali *target* tra la fase di *last testing* e la fase di *follow-up*; pertanto è necessario approfondire questo aspetto, con un'analisi mirata ad individuare i *pattern* caratteristici delle risposte all'interfaccia “Vol”, anche per migliorare il sistema di classificazione.

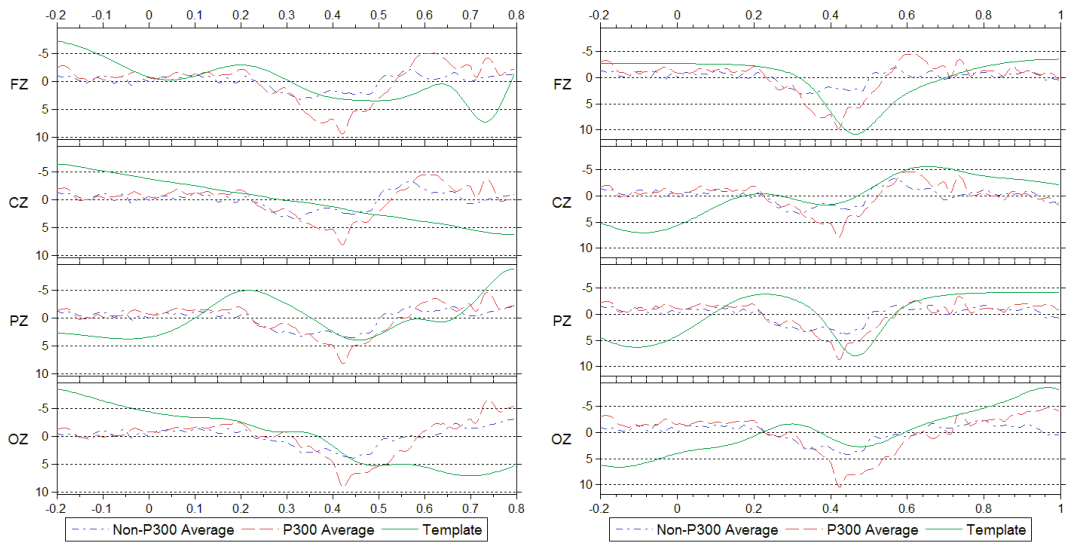
Una parziale conferma viene dal fatto che anche il sistema ICA in questo caso non sembra manifestare capacità di generalizzazione (nel confronto T2 vs T3 il *p-value* è pari a 0.0003).



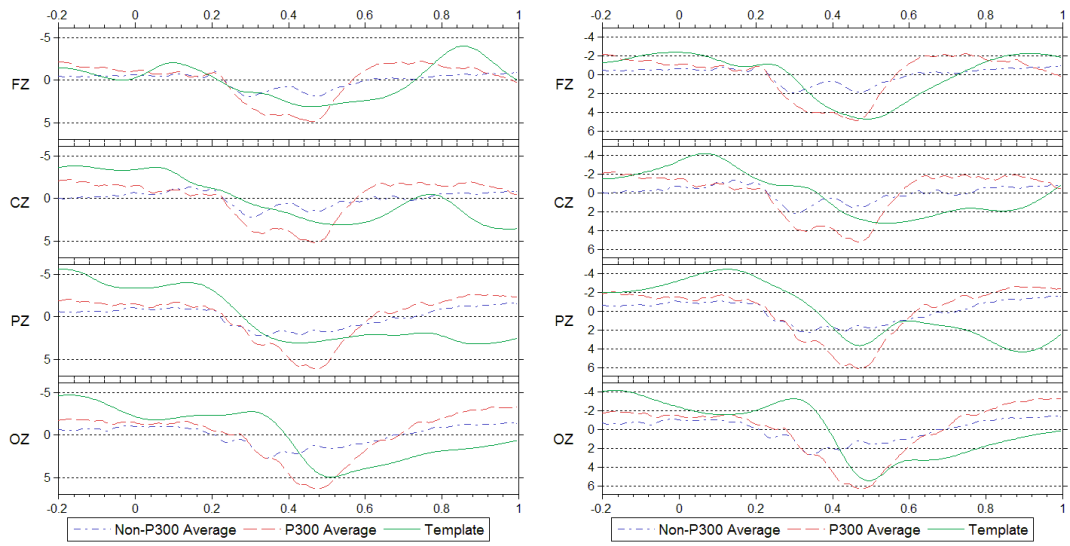
### 4.3.4 *Template*



**Figura 90** - *Template* estratto per il soggetto *sbj07* sul dataset *T2* (senza rebound a sinistra e con rebound a destra).



**Figura 91** - *Template* estratto per il soggetto *sbj14* sul dataset *T3* (senza rebound a sinistra e con rebound a destra).



**Figura 92** - Template medi sui dataset T2 (sinistra) e T3 (destra), con rebound.

## 4.4 Confronto tra le diverse interfacce

Passiamo ora al confronto tra le diverse interfacce per il nostro sistema. Confronteremo i *dataset* T2 e T3, in presenza di *rebound*, poiché suggeriamo possa essere in futuro uno standard consolidato includere il *rebound*, magari la componente completa, in quanto non si sono riscontrati svantaggi e la classificazione sembra avere benefici dal punto di vista delle prestazioni.

Misura F – Interf. “Arrows” – T2	Misura F – Interf. “Arrows” – T3
Media = 53.12	Media = 54.10
Varianza = 116.40	Varianza = 100.42
Valore massimo = 68.69	Valore massimo = 73.12
Valore minimo = 31.60	Valore minimo = 33.47
Misura F – Interf. “Auto” – T2	Misura F – Interf. “Auto” – T3
Media = 54.80	Media = 53.27
Varianza = 109.53	Varianza = 57.92
Valore massimo = 80.56	Valore massimo = 68.61
Valore minimo = 35.39	Valore minimo = 38.63

**Tabella 12** - Confronto tra le interfacce “Arrows” e “Auto” (per il *dataset* T2 con *rebound* a sinistra e per il *dataset* T3 con *rebound* a destra).

Confronto Interfacce	p-value
Interf. “Arrows” vs Interf. “Auto” – T2	0.26
Interf. “Arrows” vs Interf. “Auto” – T3	0.30

**Tabella 13** - Test *t* di Student tra le interfacce “Arrows” ed “Auto” per i *dataset* T2 e T3 con *rebound*.

Come possiamo notare in Tabella 12, le medie sono molto vicine tra loro: l’unica vera differenza è il valore massimo per l’interfaccia “Auto” per il *dataset* T2, che raggiunge un valore elevato se confrontato ai risultati delle altre esecuzioni in tabella.

A conferma di ciò, il test *t* di Student in Tabella 13 conferma che non c’è differenza statisticamente significativa tra i dati, con *p-value* pari a 0.26 nel caso del *dataset* T2 e pari a 0.30 nel caso del *dataset* T3.

<b>Misura F – Interf. “Arrows” – T2</b>	<b>Misura F – Interf. “Arrows” – T3</b>
Media = 53.12	Media = 54.10
Varianza = 116.40	Varianza = 100.42
Valore massimo = 68.69	Valore massimo = 73.12
Valore minimo = 31.60	Valore minimo = 33.47
<b>Misura F – Interf. “Vol” – T2</b>	<b>Misura F – Interf. “Vol” – T3</b>
Media = 65.74	Media = 60.89
Varianza = 128.58	Varianza = 120.18
Valore massimo = 82.95	Valore massimo = 70.94
Valore minimo = 38.26	Valore minimo = 35.50

**Tabella 14** - Confronto tra le interfacce "Arrows" e "Vol" (per il dataset T2 con rebound a sinistra e per il dataset T3 con rebound a destra).

Confronto Interfacce	p-value
Interf. “Arrows” vs Interf. “Vol” – T2	0.04
Interf. “Arrows” vs Interf. “Vol” – T3	0.004

**Tabella 15** - Test t di Student tra le interfacce "Arrows" ed "Vol" per i dataset T2 e T3 con rebound.

Come possiamo notare in

Tabella 14, per l’interfaccia “Vol” abbiamo prestazioni decisamente migliori, soprattutto per il *dataset* T2, dove abbiamo una media di misura F-bilanciata di 65.74 ed un valore massimo di 82.95, sicuramente le prestazioni migliori ottenute in questo lavoro di tesi.

A conferma di ciò, il test t di Student in

Tabella 15 conferma che c’è differenza statisticamente significativa tra i dati, con *p-value* pari a 0.04 nel caso del *dataset* T2 e pari a 0.004 nel caso del *dataset* T3.

<b>Misura F – Interf. “Auto” – T2</b>	<b>Misura F – Interf. “Auto” – T3</b>
Media = 54.80	Media = 53.27
Varianza = 109.53	Varianza = 57.92
Valore massimo = 80.56	Valore massimo = 68.61
Valore minimo = 35.39	Valore minimo = 38.63
<b>Misura F – Interf. “Vol” – T2</b>	<b>Misura F – Interf. “Vol” – T3</b>
Media = 65.74	Media = 60.89
Varianza = 128.58	Varianza = 120.18
Valore massimo = 82.95	Valore massimo = 70.94
Valore minimo = 38.26	Valore minimo = 35.50

**Tabella 16** - Confronto tra le interfacce "Auto" e "Vol" (per il dataset T2 con rebound a sinistra e per il

Confronto Interfacce	p-value
Interf. “Auto” vs Interf. “Vol” – T2	0.0005
Interf. “Auto” vs Interf. “Vol” – T3	0.01

*dataset T3 con rebound a destra).*

**Tabella 17** - Test t di Student tra le interfacce "Auto" ed "Vol" per i dataset T2 e T3 con rebound.

Come possiamo notare in Tabella 16, anche nel confronto con l’interfaccia “Auto”, l’interfaccia “Vol” ha prestazioni decisamente migliori.

A conferma di ciò, il test t di Student in Tabella 15 conferma che c’è differenza statisticamente significativa tra i dati, con p-value pari a 0.0005 nel caso del *dataset* T2 e pari a 0.01 nel caso del *dataset* T3.

## 5. Conclusioni e sviluppi futuri

---

È stato dimostrato che i normali canali di uscita neuromuscolare necessitano di un continuo *feedback* affinché le risposte all'attività cerebrale risultino ottimali: tutte i movimenti o le abilità più complessi necessitano non solo di una fase iniziale di apprendimento, ma anche di una fase di rafforzamento, in cui si ottengono correzioni e raffinamenti (90); quando il *feedback* è assente dall'inizio, il movimento non viene appreso in modo opportuno; quando il *feedback* scompare, l'abilità si deteriora.

Come sostituto dei canali periferici neuromuscolari, anche i sistemi di BCI devono restituire un *feedback* ed avere in ingresso un segnale che tenga conto del costante adattamento dell'attività cerebrale a questo *feedback*. Il successo del sistema di BCI sta nello sviluppo e nel mantenimento di un'abilità nuova, un'abilità che consiste nel controllo di specifici segnali elettrofisiologici piuttosto che di muscoli, segnali che il BCI deve tradurre in uscite che assecondino le intenzioni dell'utente.

Questo adattamento si verifica anche per attività spontanee, come la P300: una volta che la P300 viene usata come mezzo di comunicazione, subisce anch'essa modifiche dovute all'adattamento (91), e il riconoscimento e l'uso produttivo di questo adattamento sarà importante per il successo continuo delle BCI.

L'estensione delle capacità adattative del cervello al controllo dei segnali elettrofisiologici è stata osservata inizialmente in studi sulle applicazioni terapeutiche dei segnali EEG (92); benchè non sia stata dimostrata la possibilità di un controllo bidirezionale (aumento e diminuzione del segnale in specifiche caratteristiche in modo rapido accurato), la semplice ipotesi di questa potenzialità ha spinto ulteriormente lo sviluppo di sistemi di comunicazione basati su EEG.

In questo lavoro di tesi abbiamo messo a confronto le prestazioni di due sistemi di BCI, uno basato sulla decomposizione ICA (85), usato dall'équipe dell'Ospedale

IRCSS S. Camillo; uno basato su algoritmi genetici e classificatore logistico, sviluppato dal Dipartimento di Elettronica e di Biomedica del Politecnico di Milano (89).

I dati sono stati forniti dall'équipe del S. Camillo: dodici soggetti sani hanno preso parte allo studio, approvato dal Comitato Etico dell'Ospedale IRCCS San Camillo.

Tre interfacce, denominate "Arrows", "Auto" e "Vol", sono state presentate ai partecipanti, che dovevano controllare il movimento di un cursore per raggiungere un *target*, a seguito di stimolazione periferica o centrale; concettualmente le tre interfacce implementate possono essere intese come variazioni complesse del paradigma *oddball* classico; le tre interfacce si diversificano, invece, per il tipo di icona o per modalità e tempo di stimolazione per elicitare la P300; le interfacce sono state costruite facendo riferimento a diversi meccanismi attentivi (36): le prime due all'orientamento implicito esogeno, la terza all'orientamento implicito endogeno. L'obiettivo dell'uso di tre diverse interfacce era quello di testare un nuovo tipo di interfaccia per sviluppare una nuova BCI basata sulla P300.

L'interfaccia "Arrows" è un adattamento di quella utilizzata nel paradigma sperimentale originario: le quattro icone poste ai lati dello schermo sono costituite da frecce indicanti le direzioni cardinali (alto, basso, destra, sinistra).

Nelle altre due interfacce, la prima notevole differenza è che si è passati dall'uso di quattro frecce all'utilizzo di icone quadrate con delle raffigurazioni concettuali, le quali sono rappresentazioni di ipotetici bisogni; la differenza, invece, tra le due nuove interfacce riguarda la modalità di orientamento dell'attenzione.

Nell'interfaccia "Auto" l'orientamento dell'attenzione è evocato dal *blinking* delle icone (orientamento esogeno); l'interfaccia "Vol" è del tutto simile all'interfaccia "Auto", con la differenza che cambia la modalità di stimolazione: le icone in questo caso non sono soggette al *blinking* ma restano sempre fisse sullo schermo, e viene presentato un *cue* anticipatorio in posizione centrale (orientamento endogeno).

Uno degli scopi del presente studio è il confronto delle prestazioni delle tre diverse interfacce, tentando di stabilire quale di queste risulti più efficace .

Nelle fasi di raccolta dei risultati, sono state confrontate *performance* relative a diversi *dataset* e con diverse finestre temporali, che includevano o meno componenti ritenute più o meno significative ai fini della classificazione; la componente oggetto del confronto è stato il *rebound* o *slow negative wave 2* (SNW2), un'onda lenta negativa

post-P300, predominante in siti frontali o centrali dello scalpo, ed apparentemente legata a processi attentivi relativi a stimoli nuovi (33).

I risultati ottenuti dal nostro sistema risultano superiori alle prestazioni ottenute dal sistema ICA: per tutti i *dataset* di ogni interfaccia il nostro sistema risulta mediamente superiore, risultando superiore in media per 10 soggetti su 12; la robustezza del metodo è assicurata da una bassa varianza (dalle 4 alle 5 volte inferiore rispetto al metodo ICA), da medie e risultati inter-soggettivi confrontabili al variare di *dataset* e di interfacce, e da differenze statisticamente non significative al variare del *set* di *testing*, che indica anche una buona generalizzazione dell'algoritmo, condizione prevista in fase di progettazione, includendo una cross-validazione a 4 parti.

Includere il *rebound* sembra dare buoni risultati: per tutti i *dataset* di ogni interfaccia la classificazione con il *rebound* risulta essere migliore della classificazione senza *rebound*; per le interfacce “Arrows” ed “Auto” le differenze sono statisticamente significative, al contrario di quanto accade per l'interfaccia “Vol”, probabilmente a causa del fatto che la classificazione per questa interfaccia raggiunge già soluzioni ottime senza *rebound*, difficilmente migliorabili da un contenuto informativo comunque limitato.

A tal proposito, va sottolineato che il *rebound* considerato non include tutta la componente SNW2, a causa delle registrazioni limitate a nostra disposizione: pertanto, l'uso di registrazioni EEG maggiormente rappresentative, che terminino almeno 1.5 s dopo l'invio del segnale visivo, completerebbe lo studio iniziato in questo contesto, quantificando l'utilità del *rebound* tutto nella classificazione di segnali *target* e non *target*.

Per quanto concerne le differenze tra le interfacce, l'interfaccia “Arrows” ha *performance* quasi sovrapponibili all'interfaccia “Auto”; un risultato che conferma le attese è il deciso miglioramento delle prestazioni per l'interfaccia “Vol”; non c'è miglioramento per il sistema ICA: una possibile spiegazione a questa differenza è che il nostro sistema riesce a cogliere con maggiore efficacia le diverse risposte al nuovo approccio di stimolazione visiva proposta dall'équipe del S. Camillo.

Gli sviluppi futuri in questo contesto sono molteplici e trasversali, abbracciando le competenze e gli ambiti più diversi: dal miglioramento nell'acquisizione del segnale, al miglioramento dell'elaborazione e del rapporto segnale/rumore, fino allo sviluppo di nuove interfacce basate sulle recenti scoperte nelle scienze cognitive.



Noi ci limitiamo a proporre alcuni sviluppi che potrebbero essere apportati al sistema da noi esaminato: l'uso di una finestra temporale più ampia completerebbe lo studio relativo al *rebound*, che ha comunque permesso un miglioramento fino al 10% rispetto alle prestazioni iniziali; l'uso di canali EEG in cui le differenze *target* e non *target* risultino più accentuate potrebbe restituire prestazioni migliori; potrebbe essere utile introdurre un'analisi tempo-frequenza, per integrare ai *pattern* temporali *pattern* in frequenza; a tal proposito, andrebbe modificato l'algoritmo genetico in modo da sfruttare i nuovi canali informativi.

Infine, uno sviluppo immediato potrebbe essere implementare una versione in tempo reale del sistema: ciò porterebbe alla realizzazione di un *tool* applicativo in ambito clinico, che potrebbe coadiuvare lo sviluppo di nuovi paradigmi riabilitati e potenzialità comunicative.

# Bibliografia

---

1. **Birbaumer, N.** Breaking the silence: Brain-Computer Interfaces (BCI) for communications and motor control. *Psychophysiology*. 2006, 43, p. 517-32.
2. **Blankertz, B, et al.** Boosting bit rates and error detection for the classification of fast-paced motor commands based on single-trial EEG analysis. *IEEE T Neur Sys Reh*. 2003, Vol. 2, 11, p. 127-31.
3. **Wolpaw, JR, et al.** Brain-computer interfaces for communication and control. *Clin Neurophysiol*. 2002, Vol. 6, 113, p. 767-91.
4. **Dal Seno, B, Matteucci, M e Mainardi, L.** Online detection of P300 and error potentials in a BCI speller. *Comput Intell Neurosci*. 2010.
5. **Pennisi, P e Sarlo, M.** *Indici elettrofisiologici in psicologia*. Padova : Cleup, 1998.
6. **Berger, H.** Uber das electrenkephalogramm des menchen. *Arch Psychiatr Nervenkr*. 1929, 87, p. 527-70.
7. **Walter, WG, et al.** Contingent Negative Variation: an electric sign of sensorimotor association and apectancy in the human brain. *Nature*. 1964, 203, p. 380-84.
8. **Pfurtscheller, G e Neuper, C.** Motor imagery activates primary sensorimotor area in man. *Meurosci Lett*. 1997, 239, p. 65-68.
9. **Altenmuller, EO e Gerloff, C.** Psychophysiology and the EEG. [aut. libro] E Niedermeyer e FH Lopes da Silva. *Electroencephalography: basic principles, clinical applications and related fields*. 4th. Baltimore : Lippincott Williams & Wilkins, 1999, p. 637-55.
10. **McFarland, DJ, et al.** Mu and theta rhythm topographies during motor imagery and actual movement. *Brain Topogr*. 2000a, 3, p. 177-86.
11. *An optimal spatial filtering electrode for Brain-Computer Interface.* **Besio, WG, Kay, SM e Liu, X.** Minneapolis : s.n., 2009. 31st Annual International Conference of the IEEE EMBS.
12. **Donchin, E e Smith, DB.** The contingent negative variation and the late positive wave of the average evoked potential. *Electroen Clin Neuro*. 1970, Vol. 2, 29, p. 201-03.
13. **Donchin, E.** Surprise !....Surprise ? *Psychophysiology*. 1981, 18, p. 493-513.

14. **Sutton, S, et al.** Evoked-potential correlates of stimulus uncertainty. [a cura di] 1187-88. *Science*. 1965, Vol. 700, 150.
15. **Chiappa, KH.** *Evoked potentials in clinical medicine*. 3rd. s.l. : Lippincott Williams & Wilkins, 1997.
16. **Sutton, S e Runchkin, DS.** The late positive complex. Advances and new problems. *Ann NY Acad Sci*. 1984, 425, p. 1-23.
17. **Duncan-Johnson, CC e Donchin, E.** On quantifying surprise: The variation in event-related potentials with subjective probability. *Psychophysiology*. 1977, 14, p. 456-67.
18. **Ruchkin, DS, et al.** Slow wave and P300 in signal detection. *Electroen Clin Neuro*. 1980, Vol. 1-2, 50, p. 35-47.
19. **Snyder, E e Hillyard, SA.** Long latency evoked potentials to irrelevant, deviant stimuli. *Behav Biol*. 1976, Vol. 3, 16, p. 319-31.
20. **Polich, J.** Updating P300: An integrative theory of P3a and P3b. *Clin Neurophysiol*. 2007, 118, p. 2128-48.
21. **Desmedt, JE.** Clinical uses of evoked potentials. *Clin Neurol*. 1984, Vol. 12, 24, p. 1198-210.
22. **Knight, RT.** Decreased response to novel stimuli after prefrontal lesions in man. *Electroencephalogr, Clin Neurophysiol*. 1984, 59, p. 9-20.
23. **Farwell, LA e Donchin, E.** Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephal, Clin Neurophysiol*. 1988, 70, p. 510-23.
24. **Donchin, E, Spencer, KM e Wijesinghe, R.** The mental prosthesis: Assessing the speed of a P300-based brain-computer interface. *IEEE T Neur Sys Reh*. 2000, Vol. 2, 8, p. 174-79.
25. **Bayliss, JD e Ballard, DH.** A virtual reality testbed for brain-computer interface research. *IEEE T Neur Sys Reh*. 2000, Vol. 2, 8, p. 188-90.
26. **Sommer, W e Schweinberger, S.** Operant conditioning of P300. *Biol Psychol*. 1992, Vol. 1, 33, p. 37-49.
27. **Roder, B, et al.** Event-related potentials during auditory and somatosensory discrimination in sighted and blind human subjects. *Cognitive Brain Res*. 1996, Vol. 2, 4, p. 77-93.
28. **Ravden, D e Polich, J.** On P300 measurement stability: Habituation, intra-trial block variation, and ultradian rhythms. *Biol Psychol*. 1999, Vol. 1, 51, p. 59-76.
29. **Tecce, JJ.** Contingent negative variation (CNV) and psychological processes in man. *Psychol Bull*. 1972, Vol. 2, 77, p. 73-108.
30. **Loveless, NE e Sanford, AJ.** Effects of age on the contingent negative variation and preparatory set in a reaction time task. *J Gerontol*. 1974, Vol. 1, 29, p. 52-63.

31. **Curry, SH.** Contingent negative variation and slow waves in a short interstimulus interval go-nogo task situation. *Ann NY Acad Sci.* 1984, 425, p. 171-76.
32. **Verleger, R.** The true P3 is hard to see: Some comments on Kok's paper on degraded stimuli. *Biol Psychol.* 1988, Vol. 1, 27, p. 45-50.
33. **Rohrbaugh, JW, et al.** Bilateral distribution of the O wave. *Ann NY Acad Sci.* 1984, p. 267-70.
34. **Zimmer, H.** Habituation and recovery of a slow negative wave of the event-related brain potential. *Int J Psychophysiol.* 2002, 43, p. 225-35.
35. **Eimer, M.** An ERP study of sustained attention to stimulus eccentricity. *Biol Psychol.* 2000, 52, p. 205-20.
36. **Posner, MI, Snyder, CR e Davidson, BJ.** Attention and the detection of signals. *J Exp Psychol.* 1980, Vol. 2, 109, p. 160-74.
37. **Sutter, EE.** The brain response interface: Communication through visually electrical brain responses. *J Microcomput Appl.* 1992, 15, p. 31-45.
38. **Polich, J.** P300 clinical utility and control of variability. *J Clin Neurophysiol.* 1998, Vol. 1, 15, p. 14-33.
39. **Donoghue, JP.** Connecting cortex to machines: Recent advances in brain interfaces. *Nat Neurosci.* 2002, 5, p. 1085-88.
40. **Chapin, JK, et al.** Realtime control of a robot arm using simultaneously recorded neurons in the motor cortex. *Nat Neurosci.* 1999, 2, p. 664-70.
41. **Carmena, JM, et al.** Learning to control a brain-machine interface for reaching and grasping by primates. *PLoS biology.* 2003, Vol. 2, 1.
42. **Lebedev, MA e Nicolelis, MAL.** Brain-machine interfaces: Past, present, future. *Trends Neurosci.* 2004, 29, p. 536-46.
43. **Leuthardt, EC, et al.** A brain-computer interface using electrocorticographic signals in humans. *J Neural Eng.* 2004, 1, p. 63-71.
44. **Non-invasive brain-computer interface system to operate assistive devices. Cincotti, F, et al.** 2007. Annual International Conference of the IEEE Engineering in Medicine and Biology Society. p. 2532-35.
45. **Miller, NE.** Learning of visceral and glandular responses. *Science.* 1969, Vol. 866, 163, p. 434-45.
46. **Kamiya, J.** *Biofeedback and self-control: An Aldine reader on the regulation of bodily processes and consciousness.* Chicago : Aldine, 1971.
47. **Lutzenberger, W, Roberts, LE e Birbaumer, N.** Memory performance and area-specific self-regulation of slow cortical potentials: Dual-task interference. *Int J Psychophysiol.* 1993, 15, p. 217-26.
48. **Pfurtscheller, G, et al.** The hybrid BCI. *Front Neurosci.* 2010, 2.

49. **Beverina, F, et al.** User adaptive BCIs: SSVEP and P300 based interfaces. *Psychology*. 2003, Vol. 4, 1, p. 331-54.
50. **Schalk, G, et al.** EEG-based communication: Presence of an error potential. *Clin Neurophysiol*. 2000, Vol. 12, 111, p. 2138-44.
51. **Weiskopf, N, et al.** Principles of a brain-computer interface (BCI) based on real-time functional magnetic resonance imaging (fMRI). *IEEE T Bio-Med Eng*. 2004, Vol. 6, 51, p. 966-70.
52. **Coyle, S, et al.** On the suitability of near-infrared (NIR) systems for next-generation brain-computer interfaces. *Physiol Meas*. 2004, Vol. 4, 25, p. 815-22.
53. **Villringer, A e Obrig, H.** Near Infrared Spectroscopy and Imaging. *NeuroImage*. 2007, 34, p. 1416-27.
54. **Birbaumer, N.** Slow cortical potentials: Plasticity, operant control, and behavioral effects. *The Neuroscientist*. 1999, 5, p. 74-78.
55. **Nielsen, KD, Fuentes Cabrera, A e Feix do Nascimento, O.** EEG based BCI - Towards a better control. Brain-Computer Interface research at Aalborg University. *IEEE T Neur Sys Reh*. 2006, Vol. 14, 2.
56. **Mason, SG e Birch, GE.** A brain-controlled switch for asynchronous control applications. *IEEE T Bio-Eng*. 2000, Vol. 10, 47, p. 1297-307.
57. **Xu, N, et al.** BCI Competition 2003 - Data set IIb: enhancing P300 wave detection using ICA-based subspace projections for BCI applications. *IEEE T Bio-Eng*. 2004, 51, p. 1067-72.
58. **Kaper, M, et al.** BCI Competition 2003 - Data set IIb: Support vector machines for the P300 speller paradigm. *IEEE T Bio-Eng*. 2004, Vol. 6, 51, p. 1073-76.
59. **Allison, BZ e Pineda, JA.** ERPs evoked by different matrix sizes: Implications for a brain computer interface (BCI) system. *IEEE T Neur Sys Reh*. 2003, 11, p. 110-13.
60. **Sellers, EW, et al.** A P300 event-related potential brain-computer interface (BCI): The effects of matrix size and inter stimulus interval on performance. *Biol Psychol*. 2006, 73, p. 242-52.
61. *Improving transfer rates in brain computer interfacing: A case study.* **Meinicke, P, Kaper, M e Hoppe, F.** Canada : s.n., 2002. Proceedings NIPS.
62. **Sellers, EW e Donchin, E.** A P300-based brain-computer interface: Initial tests by ALS patients. *Clin Neurophysiol*. 2006, Vol. 3, 117, p. 538-48.
63. **Vaughan, TM, et al.** The Wadsworth BCI Research and Development Program: At home with BCI. *IEE T Neur Sys Reh*. 2006, 14, p. 229-33.
64. *A study of cross-validation and bootstrap for accuracy estimation and model selection.* **Kohavi, R.** 1995. Proceedings of the 14th International Joint Conference on Artificial Intelligence. Vol. 2.

65. **Stone, M.** Cross validation and multinomial prediction. *Biometrika*. 1974, Vol. 3, 61, p. 509-15.
66. **Vagò, E e Kemény, S.** Logistic ridge regression for clinical data analysis (a case study). *Appl Ecol Env Res*. 2006, Vol. 2, 4, p. 171-79.
67. **Agresti, A.** *Categorical data analysis*. New York : John Wiley & Sons, 2002. p. 166-94.
68. **le Cessie, S e van Houwelingen, JC.** Ridge estimators in logistic regression. *Appl Stat*. 1992, Vol. 1, 41, p. 191-201.
69. **Sivanandam, SN e Deepa, SN.** *Introduction to genetic algorithms*. s.l. : Springer, 2007.
70. **Kohavi, R e Provost, F.** Glossary of terms. *Mach Learn*. 1998, 30, p. 271-74.
71. **Schlogl, A, et al.** Evaluation criteria in BCI research. [aut. libro] G Dornhege, et al. *Towards Brain-Computer Interfacing*. s.l. : MIT Press, 2007.
72. **Provost, F e Fawcett, T.** Robust classification for imprecise environments. *Mach Learn*. 2001, Vol. 3, 42, p. 203-31.
73. **Huang, J e Ling, CX.** Using AUC and accuracy in evaluating learning algorithms. *IEEE T Knowled Data Eng*. 2005, Vol. 17, 3, p. 299-310.
74. **Blankertz, B e al., et.** The BCI Competition III: Validating alternative approaches to actual BCI problems. *IEEE T Neur Sys Reh*. 2006, Vol. 14, 2, p. 153-59.
75. **Buttfield, A e al., et.** Towards a robust BCI: error potentials and online learning. *IEEE T Neur Sys Reh*. 2006, Vol. 14, 2, p. 164-68.
76. *Is information transfer rate a suitable performance measure for self-placed brain interface systems?* **Fatourechi, M, et al.** 2006. Proceedings IEEE Int. Symp Signal Processing and Information Technology. p. 212-16.
77. **Cohen, J.** A coefficient of agreement for nominal scales. *Educ Psychol Meas*. 1960, Vol. 20, 1, p. 37-46.
78. **Mason, SG, Bashashati, A e Birch, GE.** Brain-computer interface design for asynchronous control applications: improvements to the LF-ASD asynchronous brain switch. *IEEE T Bio-Eng*. 2004, Vol. 51, 6, p. 985-92.
79. **Mason, SG, et al.** Evaluating the performance of self-paced BCI technology. *Tech Rep*. 2006.
80. **Fatourechi, M, Birch, GE e Ward, RK.** A self-paced brain interface system that uses movement related potentials and changes in the power of brain rhythms. *J Comput Neurosci*. 2007, Vol. 23, 1, p. 21-37.
81. **Huggins, JE, et al.** Detection of event-related potentials for development of a direct brain interface. *J Clin Neurophysiol*. 1999, Vol. 16, 5, p. 448-55.
82. **van Riksbergen, CJ.** *Information Retrieval*. s.l. : Butterworth, 1979.

83. **McFarland, DJ e al., et.** BCI meeting 2005 - Workshop on BCI signal processing feature extraction and translation. *IEEE T Neur Sys Reh.* 2006, Vol. 14, 2, p. 135-38.
84. **Croft, RJ e Barry, RJ.** Removal of ocular artifact from the EEG: a review. [a cura di] 5-19. *Neurophysiol Clin.* 2000, 30.
85. *P300-based brain computer interface: multiple letter keys vs four arrows displays.* **Piccione, F, et al.** Graz : s.n., 2006a. 3rd International Brain-Computer Interface Workshop and Training Course.
86. **Piccione, F, et al.** P300-based brain computer interface: Reliability and performance in healthy and paralysed participants. *Clin Neurophysiol.* 2006b, 117, p. 531-37.
87. **Kohlmorgen, J e Blankertz, B.** Bayesian classification of single-trial event-related potentials in EEG. *Int J Bifurcat Chaos.* 2004, Vol. 14, 2, p. 719-26.
88. **Citi, L, et al.** *Feature selection and classification in brain computer interfaces by a genetic algorithm.* 2004. Late-breaking papers of the Genetic and Evolutionary Computation Conference (GECCO).
89. **Dal Seno, B.** Toward an integrated P300- and ErrP-based brain-computer interface. *PhD.* 2009.
90. **Ghez, C e Krakauer, J.** The organization of movement. [aut. libro] ER Kandel, JH Schwartz e TM Jessell. *Principles of neural science.* New York : McGraw-Hill, 2000, p. 653-73.
91. **Rosenfeld, JP.** Applied psychophysiology and biofeedback of event-related potentials (brain waves): gistorical perspective, review, future directions. *Biofeedback Self-Reg.* 1990, 15, p. 99-120.
92. **Niedermeyer, E.** The normal EEG of the walking adult. [aut. libro] E Niedermeyer e F Lopes da Silva. *Electroencephalography: basic principles, clinical applications, and related fields.* s.l. : Lippincott Williams & Wilkins, 1999, p. 149-73.