Università degli studi di Padova
Dipartimento di Scienze Statistiche
Corso di Laurea Magistrale in

Scienze Statistiche

Tesi di Laurea

Classificazione di dati funzionali: un approccio basato sulla distanza Riemanniana di Rao

Relatore Prof. Livio Finos
Dipartimento di Psicologia dello Sviluppo e della Socializzazione

Laureando Alessio Benatelli
Matricola N 1104027

Anno Accademico 2016/2017

Indice

Elenco delle figure

Elenco delle tabelle

Introduzione

Una delle convinzioni che ho maturato durante il mio percorso formativo è che i metodi della statistica abbiano davvero pochi limiti in termini di possibilità di applicazione. *Tutto è numero* è stato il motto della scuola pitagorica, a significare che tutto può essere quantificato e, ogniqualvolta questo accade, la statistica può essere impiegata per cercare di estrarre *informazione* dalla grande mole di dati grezzi che oggigiorno è sempre più frequente incontrare.

In diverse applicazioni dell'analisi dei dati la descrizione del fenomeno aleatorio di interesse è affidata ad una serie di misurazioni ripetute nel tempo. La variabile casuale oggetto di studio appartiene ad uno spazio funzionale del quale si osserva una porzione finita definita dagli istanti di tempo $t_1, \ldots, t_T$; se si ipotizza che gli intervalli siano equispaziati, si può dire che ciascuna rilevazione dell'i-esima unità statistica proviene da un processo del tipo

$$y_i(t) = f_i(t) + \varepsilon_i, \qquad t = 1, \ldots, T. \tag{1}$$

in cui ε_i rappresenta un errore casuale. Nel caso in cui la numerosità delle misurazioni a disposizione sia elevata è facile che il comportamento della variabile rilevata $y_i(t)$ sia irregolare e difficilmente interpretabile. Sono quindi necessarie tecniche più sofisticate di rappresentazione e gestione delle rilevazioni che possano riassumere le informazioni contenute nella grande mole di dati. Una delle strade percorribili in questa direzione, se le osservazioni sono sufficientemente frequenti, è quella di analizzare il processo tramite **funzioni temporali** sottostanti l'andamento delle stesse. In quest'ottica l'informazione relativa al processo $y_i(t)$ viene considerata continua e potranno essere applicate le procedure e le proprietà derivanti dall'analisi matematica.

Il ruolo giocato della statistica nello studio di questi dati, comunemente chiamati **funzionali**, è legato alla presenza del termine d'errore nella spe-

cificazione del processo (1): poiché la relazione tra la funzione che si cerca di recuperare e i dati a disposizione è disturbata da rumore stocastico, c'è la necessità di strumenti che possano condurre ad una stima consistente del processo generatore dei dati per ogni unità statistica, $f_i(t)$; proprio in questo aspetto risiede la motivazione alle analisi che seguono.

In questa tesi l'attenzione si concentra su un tipo particolare di dato funzionale, il segnale elettroencefalografico. Nel Capitolo 1 verrano introdotte le sue caratteristiche principali e l'area di ricerca dalla quale provengono i dati che saranno analizzati successivamente; su questi dati verranno calcolate quelle che saranno le protagoniste dell'intero lavoro: le matrici di covarianza.

Nel secondo Capitolo saranno presentati gli strumenti e i concetti della teoria Riemanniana che si è interessati a sfruttare per la formalizzazione di modelli statistici *ad-hoc*; si è cercato di considerare le soluzioni che meglio si adattano al contesto applicativo, impiegando le proprietà che potessero aiutare nella manipolazione delle osservazioni considerate.

Il Capitolo 3 è riservato all'esplorazione di una porzione del campione ottenuta tramite il confronto di due metodologie di *visualizzazione dell'informazione*.

Nel quarto Capitolo i modelli descritti precedentemente dal punto di vista teorico verranno integrati in algoritmi di classificazione che possano essere effettivamente utilizzati per una previsione sui dati a disposizione; i risultati di tali previsioni saranno infine confrontati e discussi.

Capitolo 1

Dati EEG

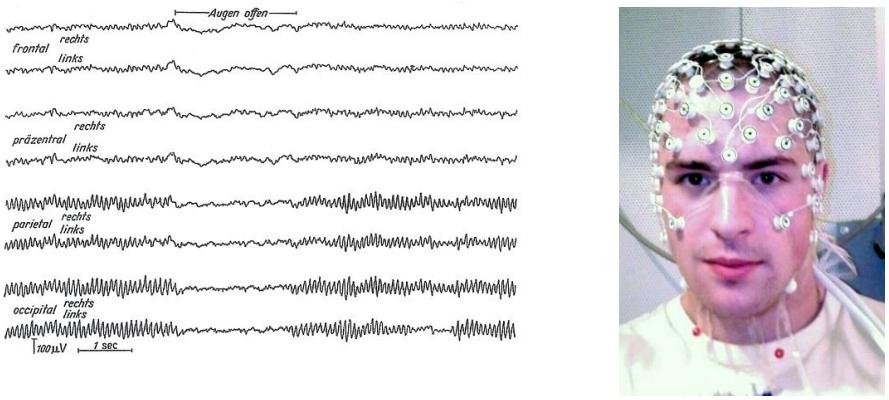

Figura 1.1: **(a)**: Elettroencefalogramma normale: onde a livello parietale e occipitale, *Wikipedia, Spike-and-wave*; **(b)**: Esempio di configurazione degli elettrodi, *Wikipedia, Elettroencefalography*

L'elettroencefalografia (EEG) è la registrazione dell'attività elettrica dell'encefalo. La tecnica è stata inventata nel 1929 da Hans Berger, il quale scoprì che vi era una differenza di potenziale elettrico tra due piccoli dischi di metallo (elettrodi) quando essi sono posti a contatto sulla cute sgrassata del cuoio capelluto [*Wikipedia, Elettroencefalografia*]. In contesto clinico viene chiamato segnale EEG l'insieme di dati relativi all'attività elettrica spontanea prodotta dal cervello in un periodo di tempo limitato; esso viene acquisito tramite una serie di elettrodi (anche detti canali) posizionati in di-

versi punti del cranio [Niedermeyer e Silva, 2005]. Data questa definizione è possibile classificare i segnali EEG come ***dati funzionali multivariati***.

1.1 Brain-Computer Interfaces

Parallelamente alle applicazioni in campo clinico, si sono sviluppati altri filoni di ricerca con lo scopo di utilizzare il segnale EEG per migliorare le condizioni di vita di soggetti con particolari necessità. Una delle innovazioni più interessanti introdotte negli ultimi tempi è quella delle cosiddette Brain-Computer Interfaces (BCIs); queste sono delle interfacce che permettono ai propri utilizzatori di interagire con dei computer esclusivamente mediante l'attività cerebrale, dove questa attività viene misurata tramite elettroencefalografia [Wolpaw e Wolpaw, 2012].
L'idea è quella di interpretare il segnale elettrico prodotto dal cervello e tradurlo in istruzioni che possano essere eseguite da un apposito software. Le BCIs hanno fornito risultati molto promettenti, ad esempio nel permettere la comunicazione a diversi pazienti con paralisi [Wolpaw et al., 1991], nel progettare programmi che reagiscono e si adattano allo stato mentale dell'utilizzatore [Zander e Kothe, 2011] e anche come nuovo dispositivo di controllo nell'ambito videoludico [Congedo et al., 2011].
Nel presente lavoro le applicazioni dei metodi statistici a dati reali riguarderanno una classe particolare di interfacce tra utilizzatore e computer, le **Motor Imagery-based BCIs**. In questo caso gli stati mentali da identificare sono l'immaginazione del movimento di una specifica parte del corpo.

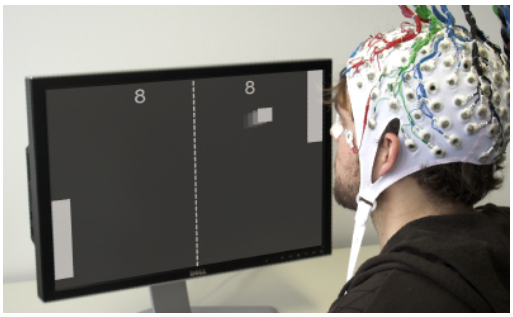

Figura 1.2: Applicazione di una BCI al controllo di un prototipo di videogioco: BrainPong, *Berlin Brain-Computer Interface*

Tuttavia, la maggior parte di queste interfacce sono scarsamente usate fuori dagli ambienti controllati dei laboratori poiché identificare uno stato mentale a partire dal segnale EEG è un processo complesso. Le caratteristiche di tale segnale, poi, non favoriscono il compito: l'EEG subisce pesantemente l'effetto di rumore indesiderato, necessita di lunghi tempi di calibrazione, è generalmente non stazionario e la sua descrizione necessita di un elevato volume di dati (direttamente proporzionale al numero di elettrodi utilizzati in fase di acquisizione) [Lotte et al., 2007]. Inoltre, un'interfaccia settata in un dato contesto ha spesso dato prova di non essere affidabile quando usata su un altro soggetto [Mühl et al., 2015].
In definitiva, la realizzazione di BCIs che possano prevedere in modo efficace necessita di tecniche avanzate di processamento dei segnali e strumenti di *machine learning*.

Il procedimento che viene seguito per tradurre l'EEG da segnale grezzo in una stima dello stato mentale dell'individuo è ottenuta usualmente tramite un approccio di **pattern recognition**, i cui passi sono i seguenti [Lotte, 2014]:

- *feature extraction*, il cui scopo è ottenere una rappresentazione sintetica dell'informazione contenuta nel segnale EEG attraverso alcuni valori rilevanti, chiamati *features* [Bashashati et al., 2007]. Questi valori dovrebbero essere in grado di catturare solo la parte di informazione rilevante nel descrivere lo stato mentale da identificare, escludendo invece il rumore e altre informazioni non rilevanti.

- classificazione, ovvero l'assegnazione di una classe all'insieme di *features* estratte dal segnale. Questa classe corrisponde al tipo di stato mentale identificato.

Applicare il *machine learning* ad un contesto di segnali elettrici significa che, per ogni utilizzatore, il campione di dati a disposizione viene suddiviso in due insiemi distinti, tradizionalmente chiamati *training set* e *test set*. Per la porzione di segnale facente parte del *training set*, in aggiunta alle *features* estratte, si utilizza l'informazione relativa alla classe di appartenenza (il corrispondente stato mentale in questo caso) per capire quali caratteristiche del segnale siano più ricorrenti nelle diverse condizioni. Sfruttando la regolazione

avvenuta sul *training set* l'algoritmo potrà assegnare al segnale appartenente al *test set* la classe ritenuta più probabile, in base ai valori assunti dalle *features* selezionate. Alcuni esempi di regolazione, effettuati in fase di *training*, possono essere la selezione di particolari canali o di bande di frequenza che vengono riconosciuti come particolarmente correlati con una delle classi.

Nella fase di classificazione, per tradurre il segnale grezzo in comandi comprensibili da un software, possono essere utilizzati diversi algoritmi in grado di associare una classe ad un particolare insieme di *features*. Nell'ambito EEG figura tra gli algoritmi più utilizzati, la Linear Discriminant Analysis (LDA), proposta da Fisher (1936); dal punto di vista computazionale essa risulta molto veloce, il che la rende adatta all'utilizzo in BCI che devono operare in tempo reale.

1.1.1 Il ruolo dell'inferenza

Fino alla fine del secolo scorso, nell'ambito delle BCI, i ricercatori si orientavano verso approcci di stampo puramente ingegneristico per rispondere alle esigenze innovative di questo settore. Gli studi si concentravano sulla messa a punto di filtri e macchinari che riuscissero a pulire il più possibile il segnale, in modo che non ci fosse bisogno di un grande impegno computazionale per la classificazione.
Tuttavia la presenza di un errore casuale all'interno del segnale acquisito, come descritto dalla (1), ha reso necessario l'adozione di tecniche inferenziali per riuscire a migliorare i risultati nel campo della stima del processo generatore dei dati.

Il contributo delle matrici di covarianza

L'approccio standard utilizzato nelle classificazioni di segnale EEG consiste nel filtraggio tramite filtro passa-banda, filtraggio spaziale e classificazione (usualmente tramite LDA). L'obiettivo di questo approccio è esaltare le differenze in varianza tra le condizioni considerate. In quest'ottica, però, i valori di covarianza tra i canali, riassunti in apposite matrici, vengono considerati come provenienti dallo spazio Euclideo, senza considerare la curvatura dello

spazio delle matrici simmetriche e definite positive al quale la covarianza appartiene [Barachant et al., 2013].
La linea maggiormente seguita per rappresentare un segnale complesso come l'EEG è quella di operare una sintesi dei dati provenienti dai diversi canali di rilevazione. Nel caso considerato in questo lavoro la sintesi viene ottenuta spostando l'attenzione dai dati grezzi (differenze di potenziale elettrico) alla misura di quanto coppie di canali varino insieme (indice della presenza o meno di una relazione lineare).

Nel concreto, il segnale EEG viene analizzato in segmenti chiamati *trials*, ognuno dei quali contiene una piccola porzione temporale dei dati rilevati dai sensori. Sia $X \in \mathbb{R}^{m \times T}$ uno di questi *trials* contenente i valori centrati di ciascuno degli m canali rispetto alla propria media, dove T indica il numero di osservazioni acquisite per ogni segmento (direttamente proporzionale alla durata della rilevazione). Viene assunto che il segnale EEG di partenza sia stato pre-filtrato in una banda di frequenza considerata di interesse.
Per ogni *trial* centrato X_i è possibile stimare la covarianza del segnale tramite la matrice di covarianza campionaria S_i di dimensione $m \times m$:

$$S_i = \frac{1}{T_i - 1} X_i X_i^\top \tag{1.1}$$

1.2 BCI Competition IV

Le competizioni nell'ambito delle BCIs coinvolgono squadre di ricercatori da tutto il mondo. Il loro scopo è favorire l'avanzamento della ricerca in questo campo e valutare quale sia il miglior approccio di manipolazione del segnale EEG e il miglior algoritmo di classificazione su un insieme di dati che è lo stesso per tutti i partecipanti[1].
I dati che saranno analizzati in questo lavoro provengono da un sottoinsieme di quelli forniti durante la quarta edizione dell'iniziativa, nel 2008. Le tre sessioni considerate, appartenenti al dataset 2b fornito dalla University of Technology di Graz (Austria), avevano come obiettivo quello di classificare correttamente dei dati provenienti da un esperimento di MI-based BCI. Nello specifico la richiesta era di proporre metodologie innovative di analisi

[1] Si veda http://www.bbci.de/competition per maggiori informazioni.

per classificare le osservazioni dell'insieme di valutazione, in modo da massimizzare le performance della previsione. Maggiori dettagli sono dati nelle seguenti Sezioni.

1.2.1 Paradigma sperimentale

L'esperimento in esame consisteva nella misurazione dell'attività cerebrale di 9 individui facenti parte di uno studio pubblicato in Leeb et al. (2007). Questi soggetti, tutti destrorsi e con normali capacità visive, hanno partecipato allo studio dietro compenso. Ognuno è stato fatto sedere su una sedia posta ad una distanza di 1 metro da un monitor, il quale poteva essere posizionato all'altezza ritenuta migliore per poter essere visionato. Per ogni soggetto sono state eseguite diverse sessioni di registrazione del segnale EEG e all'inizio di ognuna un periodo di circa 5 minuti è stato impiegato per tarare gli strumenti di rilevazione; in seguito si sono rilevati i segnali generati dall'encefalo quando veniva fornita l'istruzione di immaginare i movimenti di determinate parti del corpo.
Una sessione di registrazione è costituita da una serie di campioni di osservazioni, chiamati *trials*. Ognuno di questi contiene i valori rilevati dagli elettrodi relativamente ad una specifica richiesta di movimento.

Acquisizione dei dati

Tre canali EEG (C3, Cz e C4) sono stati registrati con una frequenza di campionamento pari a 250 Hz. Prima di essere forniti per l'analisi, questi sono stati filtrati con un filtro passa-banda tra 0.5 e 100 Hz e con un filtro *notch* a 50 Hz per eliminare l'interferenza generata dall'impianto elettrico sul segnale. Il posizionamento degli elettrodi è stato riportato essere leggermente diverso per ogni soggetto (per maggiori dettagli vedere Leeb et al., 2007).

In aggiunta ai canali deputati alla rilevazione del segnale EEG è stato registrato il segnale derivato dall'attività elettrica oculare (elettrooculogramma, EOG); questi canali hanno subito lo stesso tipo di filtraggio descritto in precedenza. Il segnale EOG acquisito in questo modo si dimostra utile nella pulizia del segnale di interesse dal rumore generato da attività non stretta-

mente connesse all'attività del cervello [Fatourechi et al., 2007]; non è stato però utilizzato ai fini della classificazione.

Gli eventi da prevedere nelle fasi successive sono due: il movimento immaginato della mano sinistra (classe 1) e della mano destra (classe 2). Ogni soggetto ha partecipato a 3 sessioni registrate in diverse giornate, ciascuna delle quali costituita da un numero di *trials* variabile da 120 a 160, divisi equamente tra le due classi in esame. Prima di iniziare l'esperimento gli individui hanno eseguito e immaginato diversi movimenti per ogni parte del corpo ed è stata selezionata quella che risultava più facile da immaginare (ad esempio stringere una pallina o tirare un freno).

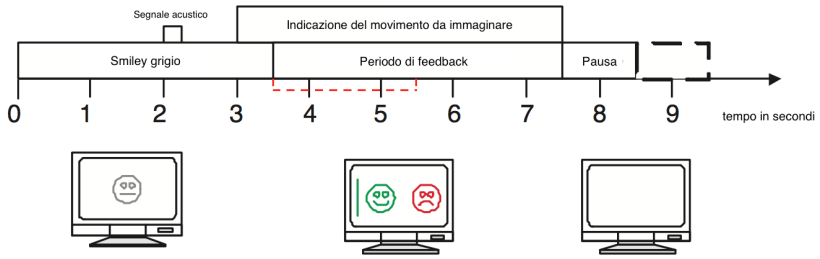

Figura 1.3: Esperimento di MI-based BCI dal quale sono stati ottenuti i dati contenuti nel dataset 2b della BCI competition IV, .

Ogni trial iniziava (secondo 0) con il monitor che mostrava al centro uno *smiley* grigio, al secondo 2 esso produceva un breve tono acustico. Al secondo 3 veniva mostrata un'indicazione visiva (una freccia che puntava a destra o sinistra, in base al tipo di richiesta) per un tempo pari a 4.5 secondi. In questo periodo, in base all'input visualizzato, al soggetto veniva richiesto di muovere lo *smiley* verso il lato destro o sinistro, immaginando il movimento della mano corrispondente. Ogni *trial* veniva seguito da una breve pausa di almeno 1 secondo. Un tempo casuale fino a 1 secondo è stato aggiunto alla pausa per evitare l'adattamento dell'individuo. Durante il periodo di feedback (dal secondo 3.5) lo *smiley* si colorava di verde quando veniva mosso nella direzione corretta o di rosso in caso contrario. Ad ogni individuo è stato chiesto di immaginare il movimento il più a lungo possibile per favorire la successiva classificazione.

1.2.2 Presentazione del dataset

Il file originale contenente tutti i dataset forniti dall'Università di Graz è stato scaricato dal sito della competizione. I comandi utilizzati per importare le sessioni di registrazione e la procedura con la quale sono stati ricavati i dati cercati sono riportati in Appendice A.

Nonostante l'elettroencefalografia sia progettata per registrare l'attività cerebrale, essa è influenzata anche da attività elettrica generata da sorgenti diverse dal cervello. L'attività rilevata che non ha origine cerebrale è chiamata **artefatto** e può essere suddivisa in fisiologico ed extrafisiologico. I primi sono generati dal paziente (da organi o tessuti diversi da quello cerebrale), i secondi derivano da sorgenti esterne all'organismo (ad esempio la strumentazione) [Urigüen e Garcia-Zapirain, 2015]. È stato appurato dagli autori che alcuni dei *trials* acquisiti sono influenzati da artefatti di entrambe le tipologie: queste categorie di *trials* non sono da includere nel processo di classificazione perché introdurrebbero una distorsione e quindi sono stati eliminati subito dopo l'importazione.
Nella Tabella 1.1 si riportano la numerosità campionaria totale per ciascun soggetto, insieme all'indicazione del relativo numero di *trials* segnalati come viziati da artefatti.

Soggetto	Genere	Età	*Trials* acquisiti	Artefatti	Campione effettivo
1	F	21	480	128	352
2	M	25	440	64	376
3	M	21	480	132	348
4	F	23	480	20	460
5	M	24	480	64	416
6	F	22	480	95	385
7	M	27	480	110	370
8	F	32	480	144	336
9	M	26	480	100	380

Tabella 1.1: Esperimento relativo al dataset 2b: dati anagrafici e numerosità campionarie per ogni soggetto.

Formato dei dati

La matrice contenente il segnale grezzo originale e i vettori di intestazione, aperti in precedenza nel software MATLAB, sono stati salvati in un file con estensione `.mat` in modo da poterli esportare (si veda Appendice A). Tramite il pacchetto `R.matlab` e il comando `read.mat` è stato possibile caricare nell'ambiente R i dati di ciascuna sessione per poter procedere nell'elaborazione.

Il primo passo svolto in questa direzione è stato quello di suddividere la matrice contenente l'intera sessione in sotto-matrici, ognuna rappresentante uno specifico *trial*. Nello specifico, seguendo l'approccio adottato in Ang et al., 2012, si è scelto di portare alle fasi successive di analisi la porzione temporale di segnale che inizia 0.5 secondi dopo la visualizzazione dell'indicazione del movimento da immaginare e comprende un totale di 500 rilevazioni per ogni *trial* (corrispondenti a 2 secondi di rilevazione, il periodo evidenziato dalla linea rossa tratteggiata nella Figura 1.3). Questa tecnica consente di focalizzare l'attenzione sulla porzione di segnale in cui è concentrato lo sforzo dei soggetti nell'immaginazione del movimento, limitando al minimo l'interferenza dovuta a fattori estranei dal punto di vista temporale.
La struttura posseduta dai dati di ogni individuo alla fine della fase di importazione è quella di una lista di matrici con dimensioni

$$numero\ di\ canali\ registrati\ (=3)\ \times\ numero\ di\ rilevazioni.$$

La dimensione della lista dipende dal numero di *trials* considerati non affetti da artefatti. Il codice R utilizzato per l'importazione dei dati e le successive preanalisi è disponibile su richiesta all'autore.

1.3 Pre-processing

La fase di pulizia dei dati prima di qualsiasi forma di training dei modelli di classificazione è stata articolata in step distinti, in modo che fossero tenuti in considerazione diversi tipi di distorsioni che un segnale subisce durante la sua acquisizione.

1.3.1 Regressione EOG

Il problema della presenza di rumore è molto rilevante nel caso del segnale EEG. Per essere sufficientemente sicuri che quella rilevata sia solo l'attività elettrica del cervello, e non quella prodotta dal movimento di altri muscoli, la soluzione che tipicamente si dimostra più efficace è quella di calcolare la media di diversi *trials* in modo da depurare il segnale e isolare le componenti veramente collegate all'azione immaginata. Nell'ambito delle BCI progettate a partire da segnale EEG ottenere una media è spesso una strada non percorribile: riprodurre lo stesso processo mentale più volte, per ottenere quella che nelle analisi seguenti sarà trattata come una singola osservazione, ha l'effetto di aumentare drasticamente i tempi di seduta richiesti ad ogni soggetto nelle sessioni di registrazione, senza considerare la frustrazione che questo comporta [Van Vliet, 2006].

Una delle principali fonti di rumore è il movimento degli occhi. Questi vengono erroneamente considerati come attività cerebrale dagli elettrodi del caschetto EEG e, posto che questo tipo di attività non sono usualmente collegate alle funzioni cerebrali di interesse, sono considerate rumore.
Un metodo per ridurre il rumore causato dal movimento degli occhi è costituito da una particolare applicazione della regressione lineare. Tale procedura sfrutta la registrazione dei canali EOG ottenuta in contemporanea a quella dei canali EEG. Il segnale EOG misura quasi esclusivamente l'attività elettrica prodotta dal movimento degli occhi e quindi può essere sottratto dal segnale EEG tramite il calcolo di un appropriato coefficiente di scala. Questo ridurrà l'entità del rumore senza il bisogno di ripetere più volte le misurazioni per farne la media. È una procedura computazionalmente semplice e veloce, eseguibile in tempo reale, che rende virtualmente possibile l'applicazione presso qualsiasi sistema BCI che sperimenta problematiche relative a questa categoria di rumore [Schlogl et al., 2007].

Il metodo descritto in tale articolo ipotizza il seguente modello per il segnale EEG:

$$Y = S + BN \tag{1.2}$$

In questa configurazione:

- Y è il segnale captato dai sensori del caschetto: una matrice in cui ogni colonna corrisponde ad una rilevazione e ogni riga ad un canale EEG,
- S è il segnale EEG pulito da qualsiasi distorsione (una matrice con le stesse dimensioni di Y),
- N rappresenta il rumore EOG compreso nel segnale captato dalla strumentazione con dimensioni (numero di canali EOG acquisiti $\times$ numero di rilevazioni),
- B è una matrice di pesi che descrive le relazione tra le fonti di rumore e i canali EEG acquisiti (numero di canali EEG $\times$ numero di canali EOG).

Quando entrambe le matrici B e N sono note il segnale cercato S può essere ottenuto dalla seguente equazione:

$$S = Y - B\,N \tag{1.3}$$

La matrice B può essere stimata assumendo che S e N (i due segnali rilevati dagli strumenti) siano indipendenti; in quest'ottica il modello (1.2) si riduce ad una regressione lineare semplice dove B rappresenta il coefficiente angolare della retta da stimare, ottenibile come

$$B = S_{YN}\,S_{NN}^{-1}. \tag{1.4}$$

Dove S_{NN} è la matrice di autocovarianza tra i canali EOG e S_{YN} è la crosscovarianza tra i canali EEG ed EOG.

Il problema di questa procedura è dato dai sensori EOG, che raccogliendo parte dell'attività EEG, provocano un aumento della correlazione tra S e N (assunti come indipendenti), portando ad una sovrastima di B. Vari metodi sono stati discussi per porre rimedio a questo fenomeno [Croft e Barry, 2000] ma ognuno di questi mostra delle debolezze. Un altro dettaglio da evidenziare sul metodo riguarda il fatto che interferenze che agiscono sia sui canali EOG che su quelli EEG non possano essere rilevate; la conseguenza è di nuovo un

aumento della correlazione tra i canali che porta nuovamente ad una sovrastima di B.
Seppur consapevoli delle criticità della procedura descritta si è deciso comunque di procedere considerando trascurabile la distorsione introdotta data la sua semplicità a livello computazionale.

1.3.2 Riduzione della dimensionalità

La necessità di avere a disposizione strumenti analitici e di rilevazione del segnale che siano in grado di personalizzarsi rispetto allo specifico individuo considerato ha portato i ricercatori nel campo delle BCI a cercare paradigmi sperimentali che consentissero una rilevazione del segnale quanto più estesa e approfondita. La strada comunemente percorsa per garantire che il segnale acquisito abbia queste caratteristiche è quella di aumentare il numero di sensori impiegati, consentendo una maggiore capacità di captare le diverse "sfumature" delle attività cerebrali umane [Blankertz et al., 2010].
Tuttavia, predisponendo un aumento dei canali di rilevazione del segnale, si induce anche un aumento della mole di dati da trattare per ottenere i risultati desiderati. Come descritto nella Sezione 1.1, la *feature extraction* mira a rappresentare il segnale EEG con un numero ristretto di informazioni rilevanti provenienti dalla rilevazione stessa. La ragione per la quale non è consigliabile fornire direttamente il segnale grezzo agli algoritmi di classificazione si riconduce alla cosiddetta ***maledizione della dimensionalità***. Essa stabilisce che la quantità di dati necessaria per descrivere in modo opportuno le diverse condizioni cresce esponenzialmente con il numero di variabili considerate [Friedman, 1997]. La presenza di un numero elevato di sensori, se non accompagnata da un adeguato aumento della numerosità delle osservazioni di *training* a disposizione, potrebbe addirittura peggiorare le capacità predittive degli strumenti utilizzati. . L'indicazione è di utilizzare dimensioni campionarie almeno 5 volte più elevate rispetto al numero di variabili [Raudys e Jain, 1991].

Nell'utilizzo pratico non è possibile dedicare così tanto tempo alla sola calibrazione della strumentazione, per questo motivo si utilizzano procedure che rappresentano un compromesso, basate su rappresentazioni più compatte

del segnale, senza rinunciare a parte dell'informazione prodotta dall'attività cerebrale.

Filtraggio spaziale

L'informazione spaziale ottenuta durante le sessioni di registrazione del segnale EEG è legata al numero e al posizionamento degli elettrodi sulla superficie del caschetto. Nell'ambito delle BCI si è interessati a comprendere come l'attività cerebrale vari nel momento in cui il soggetto sta pensando di compiere una delle azioni presenti nel set di alternative considerate; per questo motivo è importante conoscere quali sono le aree del cervello che a priori è noto siano coinvolte nei processi mentali considerati. In pratica questo potrebbe significare selezionare specifici canali EEG, concentrandosi sul segnale proveniente da determinate aree del cervello, oppure ottenere una riduzione della dimensionalità del problema considerando combinazioni lineari dei canali contigui.

Non essendo tra gli obiettivi prefissati, in questa tesi ci si limita a fare un accenno al tipo di problematica affrontata, per i dettagli riguardanti la modalità con cui operano le tecniche di filtraggio si rimanda alla letteratura esistente sull'argomento.

Capitolo 2

Metodi statistici per le varietà Riemanniane

La disciplina alla quale appartengono gli strumenti matematici che verranno presentati in questo Capitolo è chiamata **geometria differenziale**. Essa è una branca della matematica che utilizza, tra gli altri, il calcolo integrale e l'algebra lineare per studiare le strutture geometriche nell'ambito di spazi topologici differenziabili. Questi spazi, chiamati varietà, hanno la caratteristica di essere localmente approssimabili da uno spazio Euclideo e comprendono: curve, superfici e più in generale varietà differenziabili n-dimensionali.
La geometria Riemanniana è una sottoclasse di quella differenziale. Essa si occupa di una particolare categoria di varietà che prendono il nome di *varietà Riemanniane*.

Definizione 1. *In geometria differenziale, una varietà Riemanniana è uno spazio topologico liscio*[1] *dotato di una* ***metrica Riemanniana****.*

Definizione 2. *Per metrica Riemanniana si intende una famiglia di prodotti interni che associa uno scalare reale strettamente positivo ad una coppia di vettori non nulli e tangenti ad un punto della varietà.*

Questo strumento generalizza il concetto di prodotto scalare in uno spazio Euclideo ed eredita diverse sue proprietà, alcune delle quali verranno utiliz-

[1] in cui le derivate di qualsiasi ordine esistono e sono continue ovunque.

zate nelle prossime Sezioni.
Una metrica Riemanniana consente di definire diversi concetti geometrici nell'ambito delle varietà Riemanniane, come ad esempio: la lunghezza di una curva, distanza, area e volume, curvatura dello spazio.

2.1 La geometria Riemanniana applicata allo spazio $\mathcal{P}_m$

Tutte le matrici di covarianza empirica delle quali si è trattato nel Capitolo 1 condividono delle proprietà algebriche interessanti. L'insieme contenente tutte le matrici di covarianza fa parte, a sua volta, di uno spazio con il quale condivide alcune di queste caratteristiche.

Definizione 3. *Sia $\mathcal{P}_m$ lo spazio contenente tutte le matrici Y $m \times m$ reali, simmetriche e strettamente definite positive,*

$$Y^\top - Y = 0 \qquad x^\top Yx > 0 \quad \forall x \in \mathbb{R}^m \tag{2.1}$$

Prima di poter introdurre concetti relativi alla misura in tale spazio, bisogna definire la metrica che verrà considerata. Per le applicazioni affrontate in questo lavoro verrà utilizzata la metrica di Rao-Fisher, la quale fornisce allo spazio $\mathcal{P}_m$ la struttura di una varietà Riemanniana *omogenea*[2] [Zanini et al., 2016] ed è definita come segue:

$$ds^2(Y) = \mathrm{tr}[Y^{-1}\, dY]^2 \tag{2.2}$$

Tale metrica per $\mathcal{P}_m$ è una forma quadratica $ds^2(Y)$ che quantifica l'entità dello spazio che separa due oggetti, $Y \in \mathcal{P}_m$ e $(Y + dY) \in \mathcal{P}_m$, dove dY rappresenta uno spostamento infinitesimale. Dalla (2.1) deriva che entrambe le matrici Y e $(Y + dY)$ sono simmetriche e definite positive, perciò anche dY avrà le stesse proprietà. [Said et al., 2015]

[2] per ogni $Y, Z \in \mathcal{P}_m$, esiste una matrice reale e non singolare A di dimensioni $m \times m$, tale che $A^\top Y A = Z$. Una possibile scelta è $A = Y^{-1/2} Z^{-1/2}$.

2.1.1 La distanza di Rao

La metrica di Rao-Fisher definisce una distanza sulla varietà Riemanniana considerata. Essa viene chiamata distanza di Rao, $d : \mathcal{P}_m \times \mathcal{P}_m \to \mathbb{R}_+$, ed è definita come segue [Atkinson e Mitchell, 1981]:

Definizione 4. *Siano $Y, Z \in \mathcal{P}_m$ e sia $c : [0,1] \to \mathcal{P}_m$ una curva differenziabile con $c(0) = Y$ e $c(1) = Z$. La lunghezza $L(c)$ di c è definita da:*

$$L(c) = \int_0^1 ds(c(t))\, dt \tag{2.3}$$

La distanza di Rao è l'estremo inferiore di $L(c)$, preso tra tutte le curve differenziabili c.

Quando è dotato di una metrica come quella di Rao-Fisher, lo spazio $\mathcal{P}_m$ rappresenta una varietà Riemanniana con curvatura negativa [Lenglet et al., 2006 (Teorema 2.2.2, Pagina 428)]. Come conseguenza di questo teorema si evince che la traiettoria localmente più breve tra due punti dello spazio, Y e X, è realizzata da un unica curva γ, conosciuta con il nome di geodetica [Said et al., 2015]. L'equazione di questa curva è la seguente:

$$\gamma(t) = Y^{1/2}(Y^{-1/2}ZY^{-1/2})^t\, Y^{1/2} \tag{2.4}$$

Data l'espressione (2.4) è possibile calcolare $L(\gamma)$ tramite (2.3). L'espressione risultante prende il nome di distanza di Rao[3], $d(Y, Z)$ [Terras, 2012]

$$d^2(Y, Z) = \mathrm{tr}[\log(Y^{-1/2}ZY^{-1/2})]^2 \tag{2.5}$$

Un'ulteriore specificazione della distanza di Rao si trova in Zanini et al., 2016,

$$d^2(Y, Z) = \sum_{i=1}^{m} \log^2(\lambda_i) \tag{2.6}$$

dove $\lambda_i, ..., \lambda_m$ sono gli autovalori della matrice $Y^{-1/2}ZY^{-1/2}$.

[3]Tutti i richiami a funzioni presenti in (2.4) e (2.5) sono da intendersi come la versione matriciale delle funzioni stesse.

Lo spazio $\mathcal{P}_m$, quando è dotato di una metrica Riemanniana come quella di Rao-Fisher, gode di una serie di importanti proprietà geometriche [Terras, 2012]. Tra queste ce ne sono alcune che saranno utili alle analisi del presente lavoro. Nella Sezione 2.1 è stato introdotto il concetto di *spazio omogeneo* nell'ambito delle varietà Riemanniane. Tale caratteristica dello spazio $\mathcal{P}_m$ comporta che la metrica di Rao-Fisher e la distanza di Rao, definiti in esso, siano invarianti nei confronti di trasformazioni isometriche[4] come ad esempio quelle riportate di seguito:

Teorema 1. *Siano $Y, Z \in \mathcal{P}_m$ e sia A una matrice reale, non singolare di dimensioni $m \times m$. Vale:*

$$ds^2(Y) = ds^2(A^\top Y A) \qquad ds^2(Y) = ds^2(Y^{-1}) \tag{2.7}$$

Dove $ds^2(Y)$ è la metrica di Rao-Fisher (2.2)

$$d(Y, Z) = d(A^\top Y A, A^\top Z A) \qquad d(Y, Z) = d(Y^{-1}, Z^{-1}) \tag{2.8}$$

Dove $d : \mathcal{P}_m \times \mathcal{P}_m \to \mathbb{R}_+$ è la distanza di Rao (2.5),(2.6)

Prima di poter trattare la seconda proprietà della metrica di Rao-Fisher, di vitale importanza nelle sezioni seguenti, vanno presentati alcuni strumenti legati al concetto di *baricentro Riemanniano*.

Definizione 5. *Sia π una distribuzione di probabilità su $\mathcal{P}_m$. La funzione di varianza di π è definita come $\mathcal{E}_\pi : \mathcal{P}_m \to \mathbb{R}_+$*

$$\mathcal{E}_\pi = \int_{\mathcal{P}_m} d^2(Y, Z)\, d\pi(Z) \tag{2.9}$$

Definizione 6. *Sia $Y = \{Y_1, \ldots, Y_N\} \in \mathcal{P}_m$ un campione di matrici indipendenti distribuite secondo π, la matrice $\bar{Y}_N \in \mathcal{P}_m$ che minimizza globalmente la funzione $\mathcal{E}_\pi$ è chiamata **baricentro Riemanniano** [Afsari, 2011].*

[4] movimento rigido di un oggetto o di una figura geometrica. Formalmente è una funzione fra due spazi metrici che conserva le distanze.

Dato che lo spazio $\mathcal{P}_m$ dotato della metrica di Rao-Fisher è una varietà Riemanniana con curvatura negativa, vale il seguente teorema:

Teorema 2. *Se π è una distribuzione di probabilità su $\mathcal{P}_m$, allora π ha un unico baricentro Riemanniano $\bar{Y}_\pi$*

Dato che il baricentro $\bar{Y}_N$ minimizza la somma dei quadrati delle distanze all'interno di un insieme Y, esso giocherà un ruolo determinante nel processo di classificazione. Come verrà formalizzato nella Sezione 2.2, $\bar{Y}_N$ può essere considerato come un dato rappresentativo (la media o la moda Riemanniana) di un insieme di matrici di covarianza [Moakher, 2005].

2.2 Distribuzione di Riemann-Gauss

Allo scopo di gestire in modo efficace i dati provenienti dallo spazio delle matrici simmetriche e definite positive, in Said et al., 2015 è stato proposto un modello probabilistico definito in modo rigoroso. Esso consente di rappresentare la variabilità statistica di questa tipologia di dati e di applicare le consuete procedure inferenziali. La distribuzione di interesse è stata progettata per estendere allo spazio $\mathcal{P}_m$, introdotto nella Sezione 2.1, le utili proprietà possedute dalla distribuzione Normale nell'usuale spazio Euclideo. Per questo motivo si parla di *distribuzione di Riemann-Gauss*,

Definizione 7. *La distribuzione di Riemann-Gauss $G(\mu, \sigma)$ è una distribuzione di probabilità su $\mathcal{P}_m$, descritta dalla seguente funzione di densità:*

$$p(Y|\mu,\sigma) = \frac{1}{\zeta(\sigma)}\exp\left\{ -\frac{d^2(Y,\mu)}{2\sigma^2} \right\} \tag{2.10}$$

dove Y, μ sono matrici appartenenti a $\mathcal{P}_m$, $d^2(Y,\mu)$ è la distanza di Rao così come definita in (2.5), (2.6) e $\sigma > 0$ rappresenta il parametro di dispersione in termini di distanza quadratica media da μ.

La distribuzione di Riemann-Gauss fornisce una rappresentazione concreta, dal punto di vista statistico, del concetto geometrico di baricentro Riemanniano. Questa, che costituisce la sua caratteristica più interessante, deriva dai seguenti due teoremi:

Teorema 3. *Siano $Y_1, \ldots, Y_N$ dei campioni indipendenti da una distribuzione di Riemann-Gauss $G(\mu, \sigma)$. La stima di massima verosimiglianza $\hat{\mu}_N$ converge quasi certamente al vero valore del parametro μ quando $N \to \infty$.*

Teorema 4. *In una distribuzione di Riemann-Gauss $G(\mu, \sigma)$ con $\sigma > 0$, $\mu \in \mathcal{P}_m$ è il baricentro Riemanniano della distribuzione.*[5]

Quelle sopra riportate sono dirette conseguenze dell'estensione di proprietà della Normale di Gauss nello spazio Euclideo [Stahl, 2006]. Tale estensione allo spazio $\mathcal{P}_m$ è possibile se questo possiede una curvatura non positiva (come quando è dotata della metrica di Rao-Fisher) [Said et al., 2016].
La conclusione a cui si giunge, mettendo insieme le indicazioni fin qui ricevute, è che la stima di massima verosimiglianza $\hat{\mu}_N$ descrive in maniera statisticamente consistente il vero baricentro Riemanniano. Grazie a questo fatto è stato possibile sviluppare algoritmi che utilizzano efficacemente $\hat{\mu}_N$ a scopo predittivo [Banerjee et al., 2015] e di stima di funzioni di densità [Rosu et al., 2017] per campioni di matrici di covarianza.

Per poter definire una distribuzione di Riemann-Gauss $G(\mu, \sigma)$ tramite la funzione di densità (2.10) è necessario fornire l'esatta espressione del fattore $\zeta(\sigma)$. Lo scopo di questa tipologia di fattori in una densità è quella di normalizzare l'integrale della funzione stessa affinché restituisca un risultato unitario quando calcolato sull'intero spazio campionario. L'espressione di $\zeta(\sigma)$ quindi diventa:

$$\zeta(\mu, \sigma) = \int_{\mathcal{P}_m} \exp\left\{ -\frac{d^2(Y, \mu)}{2\sigma^2} \right\} \tag{2.11}$$

[5] Per la dimostrazione analitica si rimanda all'articolo di Said et al., 2015, Paragrafo III-C

Per ottenere un'espressione analitica dell'integrale indefinito vengono utilizzati i seguenti teoremi presentati e dimostrati in Said et al. (2015):

Teorema 5. *Per ogni $\mu \in \mathcal{P}_m$ e $\sigma > 0$, vale la seguente uguaglianza,*

$$\zeta(\mu, \sigma) = \zeta(I, \sigma) \tag{2.12}$$

dove $I \in \mathcal{P}_m$ è la matrice identità di dimensioni $(m \times m)$.

Teorema 6. *Sia $\zeta(\sigma) = \zeta(I, \sigma)$. Allora,*

$$\zeta(\sigma) = k_m \int_{\mathbb{R}^m} e^{-\frac{r_1^2 + \cdots + r_m^2}{2\sigma^2}} \prod_{i<j} \sinh\left(\frac{|r_i - r_j|}{2}\right) \prod_{i=1}^{m} dr_i \tag{2.13}$$

dove $e^{r_1}, \ldots, e^{r_m}$ sono gli autovalori della matrice $Y \in \mathcal{P}_m$ e

$$k_m = \frac{\pi^{\frac{m^2}{2}} 8^{\frac{m(m-1)}{4}}}{m! \Gamma_m(m/2)} \tag{2.14}$$

con Γ_m che rappresenta la funzione Gamma multivariata [Muirhead, 1982].

Il Teorema 5 conferma che il fattore di normalizzazione $\zeta(\sigma)$ dipende solo da σ, e non dal parametro μ. La valutazione di $\zeta(\sigma)$ è necessaria per la stima degli parametri della distribuzione, in modo particolare per la stima di σ. Questo argomento verrà trattato nel dettaglio nella Sezione seguente.

2.2.1 Inferenza sui parametri della distribuzione

In questa Sezione verrà introdotto l'argomento della stima di massima verosimiglianza dei parametri della distribuzione di Riemann-Gauss e verrà mostrato il ruolo giocato dalla distanza di Rao in questo contesto.
Nel caso di osservazioni $Y_1, \ldots, Y_N$ indipendenti e identicamente distribuite da una $G(\mu, \sigma)$, la funzione di log-verosimiglianza per i parametri μ e σ può essere rappresentata come:

$$\sum_{n=1}^{N} \log p(Y_n | \mu, \sigma) = -N \log \zeta(\sigma) - \frac{1}{2\sigma^2} \sum_{n=1}^{N} d^2(\mu, Y_n) \tag{2.15}$$

Dato che il primo termine a destra del segno di uguaglianza non contiene il parametro μ, la sua stima può essere recuperata massimizzando il secondo

termine. Questo corrisponde a minimizzare la somma di distanze di Rao al quadrato. La funzione obiettivo $\mathcal{E}_N : \mathcal{P}_m \to \mathbb{R}_+$ prende il nome di *funzione di varianza empirica* ed è definita come:

$$\mathcal{E}_N(Y) = \frac{1}{N} \sum_{n=1}^{N} d^2(Y, Y_n) \tag{2.16}$$

Ci si riferisce allo stimatore di massima verosimiglianza di μ con la notazione $\hat{\mu}_N$ e la sua espressione,

$$\hat{\mu}_N = \text{argmin}_Y \ \mathcal{E}_N(Y) \tag{2.17}$$

Il fatto che la varietà Riemanniana considerata sia uno spazio con curvatura negativa assicura che la stima di massima verosimiglianza esista e sia unica. In ogni distribuzione $G(\mu, \sigma)$, $\hat{\mu}_N$, la stima di massima verosimiglianza per μ trovata a partire dall'algoritmo 2, converge quasi certamente verso il vero valore del parametro (si veda Teorema 3).

Stima del parametro σ

Per recuperare $\hat{\sigma}_N$, lo stimatore di massima verosimiglianza per σ, si agisce come segue. Una volta ricavata l'espressione per $\hat{\mu}_N$, la funzione di log-verosimiglianza da massimizzare diventa

$$\ell(\hat{\mu}_N, \sigma) = -N \log \zeta(\sigma) - \frac{1}{2\sigma^2} \sum_{n=1}^{N} d^2(Y_n, \hat{\mu}_N) \tag{2.18}$$

La riparametrizzazione $\eta = g(\sigma) = -1/2\sigma^2$ consente di scrivere

$$\frac{1}{N} \ell(\hat{\mu}_N, \sigma) = \eta \, \frac{1}{N} \sum_{n=1}^{N} d^2(Y_n, \hat{\mu}_N) - \psi(\eta), \tag{2.19}$$

dove $\psi(\eta) = \log \zeta(\sigma)$.

La stima di massima verosimiglianza $\hat{\eta}_N$ è data da

$$\hat{\eta}_N = \text{argmax}_\eta \left\{ \eta \, \frac{1}{N} \sum_{n=1}^{N} d^2(Y_n, \hat{\mu}_N) - \psi(\eta) \right\}. \tag{2.20}$$

Dato che ψ è una funzione strettamente convessa, la sua derivata ψ' è strettamente crescente. In Said et al. (2016) è stato dimostrato che la stima di massima verosimiglianza per η esiste ed è unica in $\mathbb{R}_+$.
$\hat{\eta}_N$ può essere quindi ottenuto come la soluzione di

$$\psi'(\eta) = \mathcal{E}_N(\hat{\mu}_N), \tag{2.21}$$

dove $\mathcal{E}_N$ è la funzione di varianza empirica definita in (2.16)

Il problema di ottimizzazione (2.18) viene quindi affrontato partendo dal calcolo dei valori $\psi(\eta)$ per diversi valori di η tramite integrazione Monte Carlo; la sua stima si ottiene come media campionaria usando, per ciascun valore di η, 10^5 osservazioni generate in modo casuale a partire dalla funzione in (2.13).
Tuttavia le stime Monte Carlo risultano instabili man mano che la dimensione m cresce [Zanini et al., 2016]; per costruire un'approssimazione della funzione $\psi(\eta)$ si è quindi utilizzata una *spline* cubica di lisciamento [Azzalini e Scarpa, 2012, Capitolo 4.4.3]. $\mathcal{E}_N(\hat{\mu}_N)$ è un valore noto nel momento in cui si è ottenuta la stima del baricentro, per cui è sufficiente risolvere l'equazione nei confronti della derivata della funzione lisciata tramite *spline*.
Ottenuto $\hat{\eta}$, grazie alla proprietà di equivarianza della stima di massima verosimiglianza [Azzalini, 2004], è immediato ottenere $\hat{\sigma}$ e sostituire questo valore nella funzione di stima del fattore $\zeta(\hat{\sigma})$. Entrambe le funzioni di stima sono state implementate in R e il codice sviluppato si trova in Appendice B.2.

Nel Capitolo 4, in un'applicazione concreta degli strumenti fin qui presentati, verranno mostrate le stime di massima verosimglianza ottenute tramite la procedura illustrata.

2.3 Misture di Riemann-Gauss

Se dal punto di vista teorico la distribuzione di Riemann-Gauss è perfettamente in grado di descrivere e riassumere la variabilità statistica insita in un campione di matrici di covarianza, questa potrebbe non risultare sufficientemente flessibile nell'utilizzo pratico. Per questa ragione si è cercato di definire uno strumento che fosse abbastanza ricco da poter rappresentare la vera distribuzione di probabilità su $\mathcal{P}_m$, derivante da reali applicazioni sperimentali.
Le misture di distribuzioni parametriche sono largamente studiate e utilizzate in ambito Euclideo. Scopo di questa sezione è di generalizzare queste misture allo spazio $\mathcal{P}_m$, quando questo è dotato di una metrica che permetta di considerarlo come una varietà Riemanniana. Il nome preso da queste distribuzioni è *misture di distribuzioni di Riemann-Gauss nello spazio* $\mathcal{P}_m$.[6]

Definizione 8. *La funzione di densità di una mistura di Riemann-Gauss in* $\mathcal{P}_m$ *è definita come:*

$$p(Y) = \sum_{j=1}^{M} \lambda_j \times p(Y|\mu_j, \sigma_j) \tag{2.22}$$

dove $\lambda_1, \ldots, \lambda_M$ *sono pesi positivi non nulli che soddisfano la condizione* $\lambda_1 + \cdots + \lambda_M = 1$ *e ogni densità* $p(Y|\mu_j, \sigma_j)$ *è del tipo (2.10), con parametri* $\mu_j \in \mathcal{P}_m$ *e* $\sigma_j > 0$. *Il valore* M *è il numero di componenti della mistura*

Si ritiene che una mistura di Riemann-Gauss così definita possa rappresentare qualsiasi distribuzione di probabilità nello spazio $\mathcal{P}_m$, dato che il numero di componenti M può essere incrementato fino a garantire la necessaria flessibilità allo strumento.

[6] Per brevità di esposizione verranno chiamate misture di Riemann-Gauss

2.3.1 Stime di massima verosimiglianza

Siano $Y_1, \ldots, Y_N$ dei campioni indipendenti da una mistura di Riemann-Gauss. In questa Sezione verrà mostrato come sia possibile ottenere le stime di massima verosimiglianza dei parametri della mistura $\theta = \{\lambda_j, \mu_j, \sigma_j; j = 1, \ldots, M\}$ date tali osservazioni.
Questo risultato è ottenuto applicando un nuovo algoritmo di tipo Expectation Maximization (EM) proposto da Said et al. (2015). Questa procedura punta a generalizzare al contesto Riemanniano l'attuale algoritmo EM, utilizzato correntemente per la stima di misture parametriche in uno spazio Euclideo [Titterington et al., 1985]. Il problema di determinare il valore più adatto per il numero di componenti M della mistura, nel caso di osservazioni $Y_1, \ldots, Y_N$, prende il nome di *selezione dell'ordine* e può essere affrontato tramite procedure basate sui criteri di informazione [Leroux, 1992]. Da qui in avanti si ipotizza che M sia noto e fissato.

Descrizione dell'algoritmo EM

Vengono definite le seguenti quantità:

$$\omega_j(Y_n) \propto \lambda_j \times p(Y_n|\mu_j, \sigma_j) \qquad N_j = \sum_{n=1}^{N} \omega_j(Y_n) \tag{2.23}$$

dove la costante di proporzionalità nella prima definizione è scelta in modo che $\omega_1(Y_n) + \cdots + \omega_M(Y_n) = 1$. Dalla (2.23) segue:

$$\sum_{j=1}^{M} N_j = N \tag{2.24}$$

dove N rappresenta la numerosità del campione preso in considerazione.
L'algoritmo EM aggiorna iterativamente $\hat{\theta} = \{(\hat{\lambda}_j, \hat{\mu}_j, \hat{\sigma}_j)\}$ che è un'approssimazione della stima di massima verosimiglianza di θ. La procedura descritta

di seguito viene ripetuta fintanto che la funzione di verosimiglianza congiunta di $\hat{\lambda}_j, \hat{\mu}_j$ e $\hat{\sigma}_j$ aumenta di un valore minimo prefissato φ.

Algoritmo 1: Algoritmo EM di stima della mistura di Riemann-Gauss

1 Aggiornamento di $\hat{\lambda}_j$: basandosi sull'attuale valore di $\hat{\theta}$, viene assegnato a $\hat{\lambda}_j$ il nuovo valore

$$\hat{\lambda}_j^{\text{new}} = N_j(\hat{\theta})/N$$

2 Aggiornamento di $\hat{\mu}_j$: basandosi sull'attuale valore di $\hat{\theta}$, viene calcolato $\hat{\mu}_j^{\text{new}}$ come la matrice che minimizza $\varepsilon_j : \mathcal{P}_m \to \mathbb{R}_+$

$$\varepsilon_j(Y) = \sum_{n=1}^{N} \omega_j(Y_n, \hat{\theta})\, d^2(Y, Y_n)$$

3 Aggiornamento di $\hat{\sigma}_j$: basandosi sull'attuale valore di $\hat{\theta}$, $\hat{\sigma}_j^{\text{new}}$ viene recuperato tramite la parametrizzazione η come soluzione della

$$\psi'(\eta_j) = N_j^{-1}(\hat{\theta}) \times \sum_{n=1}^{N} \omega_j(Y_n, \hat{\theta})\, d^2(\hat{\mu}_j, Y_n)$$

$$\hat{\sigma}_j^{\text{new}} = g^{-1}(\hat{\eta}_j)$$

dove le funzioni ψ e g sono state introdotte nella Sezione 2.2.1

Si noti che l'*attuale valore di* $\hat{\theta} = \{(\hat{\lambda}_j, \hat{\mu}_j, \hat{\sigma}_j)\}$, menzionato nell'algoritmo EM, subisce delle modifiche in ognuno degli step eseguiti. Per esempio, nell'*i-esima* iterazione di aggiornamento di $\hat{\sigma}_j$, gli attuali valori di $\hat{\lambda}_j$ e $\hat{\mu}_j$ sono quelli trovati nel primo e secondo passo del medesimo ciclo dell'algoritmo. Per questa ragione le istruzioni vanno eseguite secondo l'ordine fornito.
Mentre l'aggiornamento di $\hat{\lambda}_j$ e $\hat{\sigma}_j$ sono computazionalmente veloci, la minimizzazione della $\varepsilon_j : \mathcal{P}_m \to \mathbb{R}_+$ richiesta nell'aggiornamento di $\hat{\mu}_j$ necessita di una generalizzazione dell'approccio utilizzato in (2.17) per tenere conto dei pesi $\omega_j(Y_n, \hat{\theta})$. Questo causa un dilatamento dei tempi necessari al calcolo di $\hat{\mu}_j^{\text{new}}$ che è proporzionale alla dimensione del campione considerato.

In Said et al. (2015) è mostrato che l'algoritmo EM di stima delle misture di Riemann-Gauss si riduce all'applicazione ripetuta delle tre regole di aggiornamento enunciate nell'Algoritmo 1. Gli autori hanno agito provando che il passo E viene eseguito calcolando il valore atteso della verosimiglianza congiunta alla fine di ogni iterazione e il passo M è rappresentato dalla procedura di aggiornamento della stima di $\hat{\theta}$.
Nel Capitolo 4 verranno utilizzate le funzioni da me implementate nel linguaggio di programmazione R per ottenere i parametri di interesse introdotti in queste Sezioni. In seguito sarà valutata la bontà della classificazione basata sugli strumenti statistici fin qui definiti.

Capitolo 3

Analisi preliminari

I dati derivanti dalla registrazione di segnali come l'EEG possono diventare molto complessi da gestire se il numero di elettrodi impiegati comincia a crescere. Basta pensare che in una matrice di covarianza calcolata per un'acquisizione di m canali i valori univoci da considerare per ogni *trial* sono $m(m+1)/2$.
L'obiettivo di questo Capitolo è esplorare la distribuzione spaziale dei dati di alcuni dei soggetti dell'esperimento 1.2 per avere un'idea di massima di quali siano le loro caratteristiche. Per poter manipolare i *trials* in modo efficiente sono state sfruttate delle tecniche di visualizzazione che, allo stesso tempo, sono veloci da calcolare e riassumono efficacemente le informazioni contenute in essi.

3.1 Decomposizione di matrici di distanze

Invece di considerare le singole osservazioni (che come detto soffrono del problema della dimensionalità) le tecniche applicate in questa tesi per condurre l'analisi esplorativa si basano sulle distanze tra esse. Nella Sezione 2.1.1, è stata definita la metrica di Rao-Fisher che è in grado di quantificare lo scostamento tra matrici di covarianza e sarà lo strumento sul quale verrà fondata l'analisi.

Più nel dettaglio, queste distanze tra coppie di osservazioni verranno riassunte in strutture in grado di rappresentare i dati provenienti da tutte le sessioni di registrazione di un soggetto. Tali strutture sono a loro volta matrici e sono definite come segue:

Definizione 9. *Una **matrice di distanze** è una struttura che contiene una di misura di dissimilarità calcolata tra coppie di osservazioni di un campione. È una matrice di dimensioni $M \times M$ del tipo,*

$$D_M = \begin{bmatrix} 0 & d(x_1, x_2) & \dots & d(x_1, x_M) \\ d(x_2, x_1) & 0 & & \vdots \\ \vdots & & \ddots & \vdots \\ d(x_M, x_1) & \dots & \dots & 0 \end{bmatrix} \tag{3.1}$$

dove M è la numerosità del campione considerato e $d(x_i, x_j) \geq 0$ è una misura di dissimilarità tra le due osservazioni, con $i, j = 1, \dots, M$.

I valori $d(x_i, x_j)$ rispettano le seguenti condizioni:

$$d(x_i, x_i) = 0, \quad d(x_i, x_j) = d(x_j, x_i) \qquad \forall\, i, j = 1, \dots, M$$

Nel caso in esame le osservazioni (x_i, x_j) menzionate nella Definizione 9 sono coppie di matrici di covarianza dei *trials* EEG. In entrambi gli algoritmi seguenti è stata impiegata come misura di dissimilarità la distanza di Rao (2.5), (2.6).

3.1.1 Multidimensional Scaling

Lo scaling multidimensionale (MDS) è una delle principali tecniche utilizzate nella cosiddetta *visualizzazione dell'informazione*. Esso consiste in una decomposizione di matrici caratterizzate da dimensione elevata M in uno spazio con dimensione $N \ll M$, in modo da conservare il più possibile la similarità tra le variabili coinvolte nella matrice originale. N è un valore che deve essere deciso a priori [Borg e Groenen, 2005].

Nel caso di una matrice di distanze come quella in (3.1), ad ognuno degli M *trial* vengono assegnate le coordinate per le N dimensioni selezionate. L'oggetto prodotto da questa tecnica è quindi una matrice $M \times N$ nella quale *trials* con distanze minime tra loro sono rappresentati con coordinate simili e *trials* dissimili hanno coordinate lontane [Wickelmaier, 2003]. La scelta di $N = 2$ permette di rappresentare gli oggetti in un grafico di dispersione facilmente interpretabile.

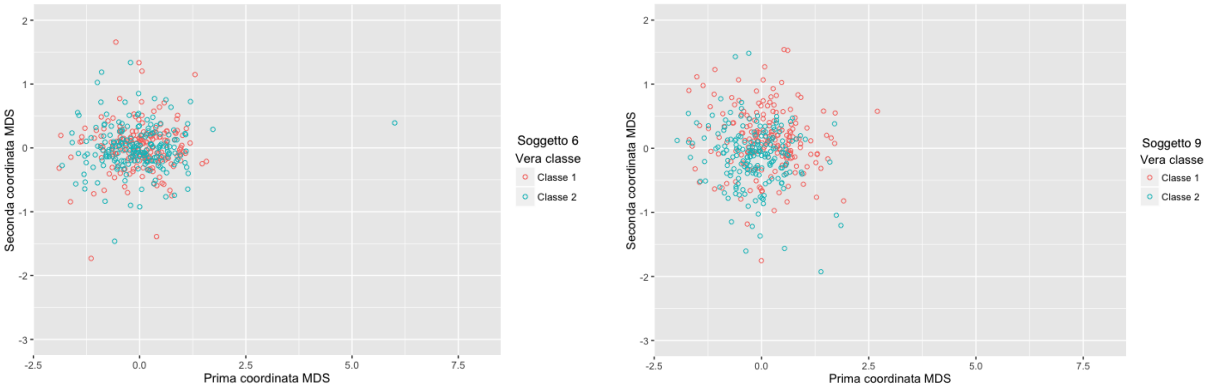

Figura 3.1: MDS con dimensione $N = 2$ eseguito sulla matrice delle distanze tra tutti i *trials* acquisiti, rispettivamente per il soggetto 6 e il sogetto 9.
Classe 1: movimento immaginato della mano sinistra;
Classe 2: movimento immaginato della mano destra.

La Figura 3.1 rappresenta la distribuzione spaziale delle *distanze di Rao tra coppie di trials* di due dei soggetti dell'esperimento 1.2. Come si vede dai punti colorati secondo la classe di appartenenza, non ci sono aree dello spazio occupate solamente da osservazioni di uno degli stati mentali; nel grafico relativo al soggetto 9 si nota che i punti della classe 2 sono leggermente più concentrati verso l'origine degli assi rispetto alla classe 1, ma quello che si percepisce è che a livello di *distanze tra i trials* ci sia tendenza ad una distribuzione simile nelle due condizioni indagate.

3.1.2 t-SNE

t-SNE è un acronimo che significa *t-distributed stochastic neighbor embedding*, ed è una tecnica di riduzione della dimensionalità proposta per la prima volta in Maaten e Hinton (2008). Il contesto di utilizzo e il tipo di risultato restituito sono gli stessi che caratterizzano il MDS; la differenza con quest'ultimo risiede nella metodologia di calcolo utilizzata per arrivare alla riduzione della complessità del dato in input.
L'algoritmo t-SNE presentato nell'articolo originale è composto di due passi principali. Per primo, si costruisce una distribuzione di probabilità per le coppie di variabili della matrice D_M in modo che ai *trials* più simili sia assegnata probabilità alta, e viceversa. Successivamente si definisce in modo analogo una distribuzione per i punti dello spazio con dimensione $N \ll M$ e si minimizza la differenza di Kullback-Leibler tra le due. La definizione di quest'ultima espressione, nella forma introdotta in Bishop (2006), è riportata di seguito:

Definizione 10. *Sia p la distribuzione di probabilità ottenuta sullo spazio con dimensione M e sia q la distribuzione che approssima p nello spazio con dimensione N, la* ***divergenza di Kullback-Leibler*** *stabilisce che la differenza attesa tra p e q ha forma,*

$$D_{\mathrm{KL}}(p|q) = \int_{-\infty}^{+\infty} p(x) \log \frac{p(x)}{q(x)} \, \mathrm{d}x \tag{3.2}$$

Data la sua formulazione, si nota che la quantità $D_{\mathrm{KL}}(p|q)$ non è una misura simmetrica.
La distribuzione q utilizzata nell'articolo per approssimare la distribuzione p delle variabili originali è una t di Student con un grado di libertà. L'applicazione di tale tecnica sulle distanze calcolate per i *trials* dei soggetti 6 e 9 permette di ottenere i seguenti diagrammi di dispersione

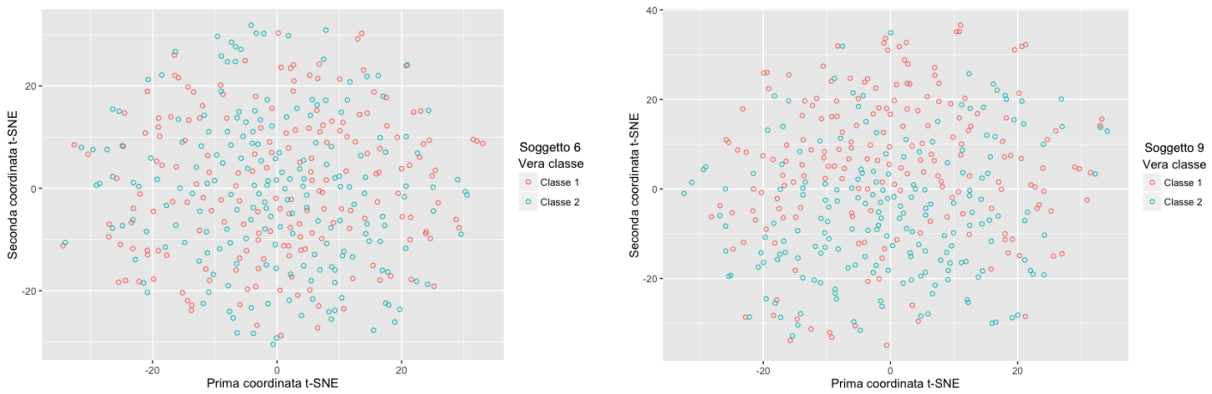

Figura 3.2: t-SNE con dimensione $N = 2$ eseguito sulla matrice delle distanze tra tutti i *trials* acquisiti, rispettivamente per il soggetto 6 e 9.
Classe 1: movimento immaginato della mano sinistra;
Classe 2: movimento immaginato della mano destra.

Anche nella Figura 3.2, come nella 3.1, il primo colpo d'occhio vede le osservazioni delle diverse classi disposte quasi casualmente nell'area del grafico. Nuovamente, si evidenzia che i dati del soggetto 9, confrontati con quelli del soggetto 6, sono caratterizzati da una distribuzione leggermente più definita, con la maggior parte dei dati della classe 2 che occupa la parte inferiore del grafico.

Queste considerazioni di carattere puramente esplorativo saranno riprese dopo l'applicazione dei modelli di classificazione proposti nel Capitolo 4.

Capitolo 4

Un approccio Riemanniano alla classificazione

Nelle seguenti Sezioni verranno utilizzati i concetti e gli strumenti statistici illustrati nel Capitolo 2 per proporre algoritmi di classificazione per dati funzionali basati sulle proprietà della geometria Riemanniana. I risultati in termini di accuratezza delle previsioni verranno poi confrontati e analizzati nella Sezione 4.6.
La procedura scelta per ottenere una misura confrontabile della performance di ogni metodologia presentata è la *10-fold cross-validation*. Il campione effettivo per ogni soggetto è stato diviso in 10 sottogruppi, ognuno dei quali è stato usato come *test set* in un'iterazione del processo di classificazione. L'errore di previsione finale è stato ottenuto come media degli errori derivanti da ognuna delle (10) classificazioni.
Tutti gli algoritmi presentati di seguito sono stati eseguiti sui dati presentati nella Sezione 1.2: il numero di canali acquisiti e, di conseguenza, la dimensione delle matrici di covarianza è $m = 3$; lo spazio topologico al quale verrà applicata la teoria Riemanniana è $\mathcal{P}_3$.
Il numero di classi tra le quali prevedere è fissato a 2, esso si riferisce al movimento immaginato della mano sinistra (classe 1) e della mano destra (classe 2).

4.1 Stima robusta delle matrici di covarianza

La prima operazione svolta sui *trials* pre-processati è stata il calcolo della covarianza campionaria tra i canali. Si è partiti dalla definizione (1.1),

$$S_i = \frac{1}{T_i - 1} X_i X_i^\top,$$

dove $S_i \in \mathbb{R}^{m \times m}$ è la covarianza campionaria di $X_i \in \mathbb{R}^{m \times T_i}$, l'$i$-esimo *trial* centrato rispetto alle medie degli m canali acquisiti, T_i è il corrispondente numero di rilevazioni acquisite. Questo stimatore della quantità di interesse è noto essere non robusto in caso di presenza di valori estremi (*outliers*) all'interno del campione di partenza, soprattutto se le matrici X_i hanno dimensione elevata [Barachant et al., 2013]. Per evitare questo fenomeno è necessario trovare una procedura alternativa che possa produrre stime affidabili. In Ledoit e Wolf (2004) viene introdotto uno stimatore consistente e asintoticamente più accurato di S_i; questo può essere usato in assenza di informazioni a priori sulla vera struttura della matrice di covarianza ed è veloce da calcolare perché ha forma esplicita. $\tilde{S}_i$ effettua uno *shrinkage* dei coefficienti della covarianza campionaria verso una matrice identità $(m \times m)$, come dalla seguente:

Definizione 11. *Siano $m_T = \langle S_i, I \rangle$, e $x_{i,\cdot k}$ la k-esima colonna di X_i,*

$$\tilde{S}_i = \frac{b_i^2}{d_i^2} m_n I + \frac{a_i^2}{d_i^2} S_i, \tag{4.1}$$

dove $d_i^2 = ||S_i - m_T||_2$, $b_i^2 = \min(\frac{1}{T^2} \sum_{k=1}^{n} ||x_{i,\cdot k} x_{i,\cdot k}^\top - S_i||_2, d_n^2)$
e $a_i^2 = d_n^2 - b_n^2$.

Scegliendo i coefficienti in modo da minimizzare una funzione di perdita di tipo quadratico, questo stimatore introduce una distorsione nella stima ma, rispetto a S_i, può essere caratterizzato da una minore varianza quando i pesi sono ottimizzati; in questo modo gli autori ritengono sia possibile ottenere un errore quadratico medio più contenuto quando la dimensione m aumenta considerevolmente.

Il comando R da utilizzare per ottimizzare in modo automatico i pesi e ottenere la stima cercata $\tilde{S}_i$ è `linshrink_cov`(X_i,k), del pacchetto `nlshrink`[1]. X_i è la matrice EEG introdotta in precedenza, con numero di colonne pari al numero di osservazioni del *trial* e numero di righe pari a m, il numero di canali di registrazione e k è un parametro che indica il numero di classi assunte presenti nel campione fornito al comando; nel presente lavoro k è stato impostato a 1 per indicare la presenza di valori da un'unica classe in ogni *trial*. L'oggetto prodotto da questo comando è la stima robusta della matrice di covarianza, semi-definita positiva con dimensioni $(m \times m)$.
Dato che il numero di canali considerati in questa tesi non è alto, l'utilizzo della tecnica di regolarizzazione presentata in questa Sezione non sarebbe stato indispensabile. Tuttavia, per completezza, è stata inserita nel lavoro perché ritenuta di interesse generale.

4.2 *Minimum Distance-to-Mean*

Il primo approccio di classificazione considerato nel presente lavoro si basa unicamente su due concetti: quello di distanza tra matrici e quello di media di covarianze. Per ognuno degli stati mentali contemplati nell'esperimento si utilizza il rispettivo *training set* per ottenere la specificazione del baricentro, successivamente i *trials* del *test set* vengono assegnati alla classe rispetto alla quale risultano meno distanti.

L'utilizzo della distanza di Rao nell'approccio *minimum distance-to-mean* permette di tenere conto delle proprietà dello spazio di cui fanno parte le matrici di covarianza, $\mathcal{P}_m$. Per coerenza con questo approccio anche i baricentri delle classi sono stati calcolati come media Riemanniana delle covarianze; nello specifico si è utilizzato un caso particolare dell'algoritmo presentato in Congedo et al., 2016. La quantità cercata è

$$\hat{\mu}_N = \text{argmin}_Y \frac{1}{N} \sum_{n=1}^{N} d^2(Y, Y_n), \tag{4.2}$$

dove N rappresenta la numerosità campionaria del *training set* di ogni classe e d^2 è la distanza di Rao, definita come in (2.5) e (2.6).

[1] si veda ftp://cran.r-project.org/pub/R/web/packages/nlshrink/nlshrink.pdf

La procedura proposta in tale articolo è un algoritmo iterativo in grado di generalizzare la stima di diverse tipologie di medie di matrici; non essendo questo l'obiettivo della tesi si rimanda alla bibliografia per maggiori dettagli. Gli autori suggeriscono che tale algoritmo può essere adattato per recuperare $\hat{\mu}_N$ in modo efficiente. Di seguito vengono riportati i passi seguiti nell'implementazione in R del codice per ciascuno degli stati mentali in esame (si veda Appendice B).

Algoritmo 2: Media pesata di ordine p

Input: esponente $p \in (-1, 1) \setminus 0$, K pesi positivi $w = \{w_1, \dots, w_K\}$ tali che $\sum_k w_k = 1$ e K matrici dal *training set* di una delle classi considerate $\{\tilde{S}_1^*, \dots, \tilde{S}_K^*\} \in \mathcal{P}_m$, con $* = \text{sgn}(p)$

Output: P, la matrice media pesata di ordine p

Definire P_0 come $(\sum_k w_k \tilde{S}_k^p)^{1/p}$.

Inizializzazione di X come la radice quadrata dell'inversa di P_0 se $p > 0$ o come la sua radice quadrata se $p < 0$.

Porre ζ uguale ad un valore decimale trascurabile come 10^{-8}.

Porre $\varphi = \frac{1}{2}(\varepsilon + 1)^{-1}/|p|$.

repeat

$H \leftarrow \sum_k [w_k (X \tilde{S}_k^* X^\top)]^{|p|}$

$X \leftarrow H^{-\varphi} X$

until $\frac{1}{\sqrt{m}} ||H - I||_F$ [a] $< \zeta$;

return

$$P = Y^\top Y, \quad Y = \begin{cases} X^{-1} & se\ p \in (0, 1] \\ X & se\ p \in [-1, 0) \end{cases}$$

[a] si veda la definizione di norma di Frobenius

La flessibilità dell'algoritmo 2 risiede nel parametro p; in base al valore passato la procedura illustrata calcola diversi tipi di media (aritmetica, armonica, ...). La quantità richiesta dalla (4.2), in questo contesto, prende il nome di **media geometrica** e può essere calcolata ponendo $p \to 0$. L'atto pratico di stima del baricentro è stato ripetuto due volte con valori positivi e negativi di p (valore assoluto nell'ordine di 10^{-4}). Il risultato finale è frutto della media delle matrici P ottenute nei due casi (la grande maggioranza delle volte queste coincidevano fino al quarto decimale con i p scelti).

La velocità di convergenza dell'algoritmo viene controllata per mezzo di $\varepsilon \in [0, 1]$: incrementando ε verso 1 si ottiene una convergenza più veloce, la quale però sottende il caso (irrealistico) di dati senza rumore. Il compromesso raggiunto tra le due situazioni è stato di fissare un valore $\varepsilon = \frac{1}{3}$, questo si traduce in un numero maggiore di iterazioni che riescono a fornire risultati affidabili a partire da dati con rumore potenzialmente rilevante.

Definizione 12. ***Regola di decisione:*** *la classe selezionata k^* è quella che realizza la seguente condizione*

$$k^* = \text{argmin}_k \, d^2(Y_t, \hat{\mu}_N(k)), \tag{4.3}$$

dove Y_t è l'osservazione da classificare, $\hat{\mu}_N(k)$ è la stima del baricentro Riemanniano della classe k e d^2 è la distanza di Rao.

4.2.1 Distribuzione rispetto al baricentro Riemanniano

Parallelamente alla classificazione, le stime univoche dei baricentri Riemanniani per gli stati mentali sono state impiegate in un altro tipo di analisi sui dati. Per ogni matrice del campione è stata calcolata la distanza di Rao dal baricentro stimato per la classe di appartenenza. Si riportano come esempio gli istogrammi ottenuti per il soggetto 1 dell'esperimento.

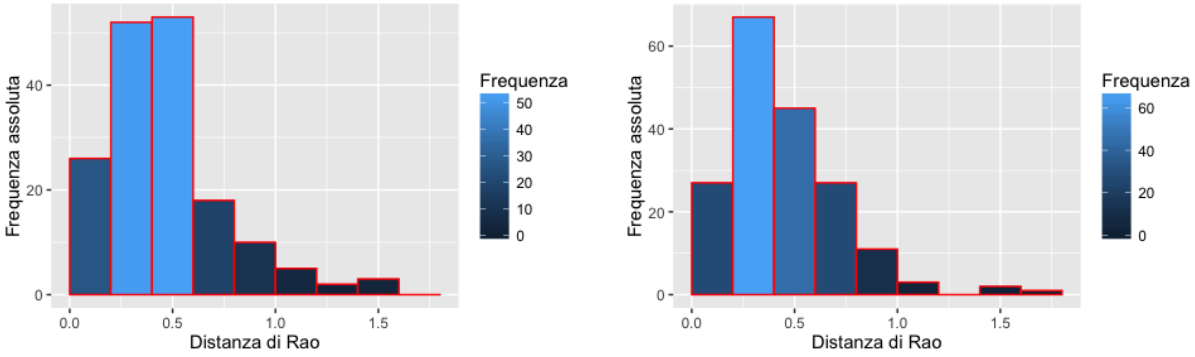

Figura 4.1: Distribuzioni di frequenza assoluta della *distanza di Rao dal baricentro stimato*, rispettivamente per le osservazioni campionarie della classe 1 e della classe 2 per il soggetto 1 dell'esperimento oggetto di studio.

Le considerazioni che seguono sono di carattere qualitativo. La Figura 4.1 evidenzia come la distribuzione delle osservazioni dei campioni si concentri attorno al proprio baricentro (frequenza elevata di osservazioni con distanza inferiore a 1), segno che questa matrice rappresenta propriamente la classe di cui fa parte. Nella coda destra della distribuzione rimane una parte marginale di matrici che sembrano discostarsi dalla media stimata; per cercare di catturare anche questa informazione, nella Sezione 4.4 il modello di classificazione sarà esteso al caso in cui esistono più baricentri per ogni stato mentale. Nella Sezione che segue, invece, sarà trattata una prima formalizzazione probabilistica nel contesto Riemanniano, sulla quale saranno fondate le fasi successive dell'analisi.

4.3 Rapporto di verosimiglianze

In questo approccio, alle informazioni sul baricentro delle classi e sulle relative distanze, si aggiunge la stima della dispersione dei dati campionari. Le osservazioni contenute nel *training set* di ciascuna classe vengono utilizzate per ottenere una stima dei parametri della distribuzione, μ e σ, per entrambi gli stati mentali; per le matrici del *test set* verrà calcolato il rapporto di verosimiglianze tra le due classi ed in base al risultato sarà definita la previsione. Per riunire i parametri menzionati in maniera formalmente puntuale ci si basa sulla distribuzione di Riemann-Gauss presentata nella Sezione 2.2:

$$p(Y|\mu,\sigma) = \frac{1}{\zeta(\sigma)} \exp\left\{ -\frac{d^2(Y,\mu)}{2\sigma^2} \right\}$$

In ogni distribuzione $G(\mu,\sigma)$, $\hat{\mu}_N$, la stima di massima verosimiglianza per μ trovata a partire dall'algoritmo 2, converge quasi certamente verso il vero valore del parametro (si veda Teorema 3).

Il valore di $\hat{\sigma}$ è stato ricavato tramite la procedura descritta nella Sezione 2.2.1; lo stimatore definito nel modo proposto soddisfa la condizione

$$\hat{\sigma}_N = \Phi\left(\frac{1}{N}\sum_{n=1}^{N} d^2(\hat{\mu}_N, Y_n)\right) = \Phi(\mathcal{E}_N(\hat{\mu}_N)), \tag{4.4}$$

dove Φ è la funzione inversa di $\sigma \to \sigma^3 \times \partial \log \zeta(\sigma)/\partial\sigma$.

Le due distribuzioni di Riemann-Gauss stimate vengono impiegate per classificare le osservazioni del *test set* secondo il seguente criterio;

Definizione 13. ***Regola di decisione:*** *sia data la differenza di log-verosimiglianze:*

$$LR \propto \ell(Y_t|\,\hat{\mu}_L, \hat{\sigma}_L) - \ell(Y_t|\,\hat{\mu}_R, \hat{\sigma}_R), \tag{4.5}$$

dove Y_t è l'osservazione da classificare, $(\hat{\mu}_L, \hat{\sigma}_L)$ e $(\hat{\mu}_R, \hat{\sigma}_R)$ sono la stime dei parametri della distribuzione di Riemann-Gauss, rispettivamente della classe 1 e della classe 2. La classe selezionata è la prima se $LR > 0$, la seconda nel caso contrario.

È interessante notare come la Definizione 13 comprenda anche il criterio utilizzato nella 12. A tal proposito nella (4.6) si mostra come è caratterizzato il test LR se si ipotizza la stessa dispersione nelle classi (ipotesi dell'approccio Minimum Distance-to-Mean);

$$\ell(Y_t|\hat{\mu}_L, \hat{\sigma}_N) - \ell(Y_t|\hat{\mu}_R, \hat{\sigma}_N) \propto \left(d^2(Y_t, \hat{\mu}_L) - d^2(Y_t, \hat{\mu}_R)\right) \tag{4.6}$$

I fattori di normalizzazione si elidono e la scelta della classe ritorna ad essere un confronto delle distanze misurate tra l'osservazione del *test set* e i baricentri delle classi.

4.4 Misture di Riemann-Gauss

Seguendo l'indicazione fornita dai dati nella Figura 4.1, è stata presa in esame l'ipotesi che per ogni stato mentale esista più di un baricentro attorno al quale si posizionano le matrici di covarianza del *training set*. Lo strumento statistico che è stato impiegato per descrivere tale ipotesi è la mistura di Riemann-Gauss introdotta nella Sezione 2.3:

$$p(Y) = \sum_{j=1}^{M} \lambda_j \times p(Y|\mu_j, \sigma_j).$$

Per la stima dei parametri si è implementato in R l'algoritmo EM 1 (il codice si trova in Appendice B); l'aggiornamento delle stime $\hat{\mu}_j$ e $\hat{\sigma}_j$ è stato ottenuto adattando l'algoritmo 2 e la procedura presentata nella Sezione 2.18 alla definizione della mistura: i pesi delle componenti, λ_j, hanno preso il posto di quelli fissati a priori nella funzione $\mathcal{E}_N$ (che valevano $\frac{1}{N}$).
Per scegliere M^*, il numero ottimale di distribuzioni di Riemann-Gauss da considerare nella mistura di ogni classe, l'algoritmo 1 è stato inserito all'interno di una funzione che stima iterativamente i parametri, aggiungendo ad ogni ciclo una nuova componente. Il criterio usato al fine di stabilire M^* è il BIC [Schwarz, 1978]:

$$BIC(M) = -2\,\ell\,(\hat{\theta}, M) + k\log(N), \tag{4.7}$$

dove $\ell(\hat{\theta}, M)$ è il valore massimizzato della funzione di log-verosimiglianza della mistura con M componenti, k è il numero totale di parametri contenuti in θ e N è la dimensione del *training set* considerato. Questo porta a definire M^* come

$$M^* = \text{argmin}_M\, BIC(M). \tag{4.8}$$

Tale criterio, utilizzato sui dati a disposizione, ha indicato per tutti i soggetti un valore di $M^* = 2$; questo indica che, come ipotizzato a partire dalla Figura 4.1, le matrici di covarianza oggetto di studio si aggregano nello spazio $\mathcal{P}_3$ attorno ad un paio di centri di massa che hanno la capacità di riassumere l'intero campione.
Una volta stimati tutti i parametri appartenenti alle misture delle classi con-

siderate, è possibile passare alla fase di classificazione delle osservazioni appartenenti al *test set*.

Per ogni classe si è stimata una mistura di Riemann-Gauss, ognuna con il suo set di parametri $\hat{\theta}$. Ipotizzando l'indipendenza tra i due stati mentali è possibile definire un'unica distribuzione per tutte le componenti con la seguente espressione:

$$p(Y_t) \propto \sum_{h=L,R} \sum_{j=1}^{M_h^*} \pi_h \hat{\lambda}_{hj} \times p(Y_t|\hat{\mu}_{hj}, \hat{\sigma}_{hj}). \tag{4.9}$$

in cui Y_t è l'osservazione del *test set*. In tale mistura per la classificazione è stata anche inserita l'indicazione della probabilità a priori delle due classi, sotto forma di proporzione di osservazioni all'interno del *training set* ($\pi_L + \pi_R = 1$). La previsione della classe viene ottenuta tramite un rapporto di verosimiglianze tra le $M_L^* + M_R^*$ componenti della (4.9) che prende la forma del criterio:

Definizione 14. ***Regola di decisione:*** *la classe selezionata è quella di cui fa parte la componente C_{κ^*} che realizza la seguente condizione,*

$$\text{argmin}_\kappa \left\{ -\log \hat{\lambda}_\kappa + \log \zeta(\hat{\sigma}_\kappa) + \frac{d^2(Y_t, \hat{\mu}_\kappa)}{2\hat{\sigma}_\kappa^2} \right\} \tag{4.10}$$

dove Y_t è l'osservazione da classificare e $\kappa = L1, \ldots, LM_L^*,\ R1, \ldots, RM_R^*$ [Said et al., 2015].

4.5 Vettorizzazione delle matrici di covarianza

Tutti i modelli proposti finora godono delle proprietà derivanti dall'utilizzo della metrica Riemanniana; nella presente Sezione sarà introdotta il metodo solitamente usato nel tradizionale contesto Euclideo in modo che, successivamente, si possa ottenere un confronto dei risultati più interessante. La procedura in oggetto prevede di considerare le matrici di covarianza dei *trials* EEG come un normale dataset di variabili esplicative; tali matrici, quadrate e simmetriche, possono essere vettorizzate e, i valori distinti in esse contenuti, possono essere forniti ad un qualsiasi algoritmo di *machine learning* per la classificazione.

L'operatore algebrico utilizzato è il `vech`, esso sfrutta la proprietà di simmetria delle matrici di covarianza per trasformare queste strutture in vettori, tenendo conto solamente dei valori non duplicati della parte triangolare inferiore (o superiore).

Definizione 15. *Per una matrice A simmetrica con dimensioni $(n \times n)$ la vettorizzazione tramite* `vech` *produce un vettore di dimensione $(n(n+1)/2 \times 1)$, ottenuto dalla parte triangolare inferiore della matrice A.*
Ad esempio, per una matrice A con dimensioni (3×3):

$$A = \begin{bmatrix} a & b & c \\ b & d & e \\ c & e & f \end{bmatrix} \quad \text{vech}(A) = \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} \tag{4.11}$$

L'algoritmo usato nel presente lavoro per trattare i vettori di valori ottenuti a partire dalle autocovarianze/crosscovarianze dei canali EEG è la Quadratic Discriminant Analysis (QDA), generalizzazione della LDA introdotta nella Sezione 1.1; tale algoritmo è stato scelto perché permette di considerare il caso in cui la varianza delle due classi sia diversa, mantenendo una complessità computazionale non troppo elevata.

La classificazione eseguita tramite QDA si basa sull'ipotesi che la popolazione di riferimento si distribuisca secondo una distribuzione Normale e opera secondo la seguente definizione:

Definizione 16. *Sia π_k la probabilità a priori della classe k e $p(x|k)$ la densità ipotizzata per l'osservazione x se facesse parte della classe k, allora la distribuzione a posteriori della classe dopo aver osservato x è*

$$p(k|x) = \frac{\pi_k\, p(x|k)}{p(x)} \propto \pi_k\, p(x|k), \tag{4.12}$$

dove la classe scelta k^* è quella per la quale $p(k|x)$ è massima. Questa è conosciuta come la ***regola di classificazione di Bayes****.*

La probabilità a priori π_k è stata calcolata ad ogni iterazione della convalida incrociata come la proporzione effettiva di osservazioni della classe k sulla numerosità totale del campione di *training*. Per gli stessi dati viene stimata anche la distribuzione $p(x|k)$, in questo caso una Normale multivariata. Il comando R utilizzato è `qda`, parte del pacchetto `MASS`.

Definizione 17. ***Regola di decisione:*** *per le osservazioni facenti parte del test set, ad ogni iterazione della convalida incrociata, si calcolano le probabilità a posteriori per ogni classe e si sceglie lo stato mentale che soddisfa la seguente condizione,*

$$k^* = \text{argmax}_k\, \pi_k\, p(Y_t|k), \tag{4.13}$$

dove Y_t è l'osservazione da classificare, π_k è la probabilità a priori della classe k e $p(Y_t|k)$ è la verosimiglianza calcolata per l'osservazione Y_t considerandola come parte della classe k.

4.6 Risultati

In questa Sezione, che rappresenta il cuore dell'analisi, sono raccolti e riassunti i risultati dei tentativi di classificazione effettuati a partire dalle metodologie descritte in precedenza. Verranno anche presentati alcuni esempi specifici ritenuti di interesse per trarre delle considerazioni sul lavoro svolto fin qui.

4.6.1 Confronto delle performance di previsione

Nella Tabella 4.1 sono presentati gli indici riassuntivi che descrivono la bontà delle previsioni effettuate sui 9 soggetti. Essi derivano dalla media delle performance rilevate in ciascuna delle 10 iterazioni della convalida incrociata.

I valori riportati in grassetto si riferiscono alla miglior performance di previsione in assoluto per ogni soggetto; quelli sottolineati indicano il miglior modello tra quelli che utilizzano i concetti della geometria Riemanniana.

Con Sensibilità si intende la proporzione di osservazioni appartenenti alla classe 1 (movimento immaginato della mano sinistra) classificate correttamente, sul numero totale di previsioni per tale classe. Con Specificità si intende la proporzione di osservazioni appartenenti alla classe 2 (movimento immaginato della mano destra) classificate correttamente, sul numero totale di previsioni per tale classe.

All'insieme di modelli testati, è stato aggiunto il Minimum Distance-to-Mean eseguito su medie aritmetiche delle matrici e con distanze Euclidee. Lo scopo di questa aggiunta è di avere un'idea più precisa dell'apporto della geometria Riemanniana all'accuratezza delle previsioni, a partire dalla medesima classe di modello di classificazione.

La cosa più evidente che si rileva è la variabilità dei risultati tra soggetti diversi, soprattutto per quanto riguarda gli ultimi due modelli della Tabella che utilizzano la metrica Euclidea, essi si comportano molto bene in alcuni casi ma non riescono a mantenere lo stesso livello di affidabilità per tutti gli individui. Al contrario, le performance dei modelli Riemanniani risultano nettamente le migliori in almeno 5 casi (nel soggetto 9 lo scarto tra i valori migliori dei due approcci è minimo).

Soggetto	Indice medio	MDRM	LR	Misture	MDEM	QDA
1	Accuratezza	**0.639**	0.615	0.635	0.585	0.579
	Sensibilità	0.638	0.625	0.500	0.633	0.678
	Specificità	0.639	0.611	0.667	0.531	0.444
2	Accuratezza	0.558	0.538	0.587	0.554	**0.729**
	Sensibilità	0.550	0.500	0.444	0.500	0.721
	Specificità	0.547	0.526	0.600	0.611	0.739
3	Accuratezza	0.554	0.521	0.535	**0.579**	0.509
	Sensibilità	0.453	0.353	0.294	0.711	0.329
	Specificità	0.647	0.706	0.765	0.447	0.659
4	Accuratezza	0.876	0.882	0.867	0.869	**0.969**
	Sensibilità	0.870	0.889	0.889	0.868	0.947
	Specificità	0.882	0.684	0.684	0.869	0.975
5	Accuratezza	**0.638**	0.598	0.610	0.550	0.525
	Sensibilità	0.575	0.333	0.333	0.745	0.640
	Specificità	0.680	0.789	0.800	0.355	0.390
6	Accuratezza	**0.608**	0.557	0.568	0.529	0.573
	Sensibilità	0.622	0.667	0.667	0.589	0.634
	Specificità	0.595	0.444	0.444	0.467	0.444
7	Accuratezza	**0.681**	0.673	0.656	0.639	0.506
	Sensibilità	0.644	0.619	0.667	0.689	0.600
	Specificità	0.717	0.684	0.706	0.589	0.378
8	Accuratezza	0.851	0.853	**0.857**	0.830	0.527
	Sensibilità	0.800	0.714	0.737	0.856	0.375
	Specificità	0.906	0.875	0.856	0.809	0.712
9	Accuratezza	0.718	0.735	**0.755**	0.752	0.552
	Sensibilità	0.705	0.685	0.685	0.711	0.476
	Specificità	0.732	0.738	0.789	0.795	0.517

Tabella 4.1: Riassunto dei risultati medi ottenuti dalla classificazione tramite *10-fold cross validation*. I modelli considerati sono: MDRM (Minimum Distance-to-Riemannian-mean, Sezione 4.2); LR (Rapporto di verosimiglianze, Sezione 4.3); Misture di Riemann-Gauss (Sezione 4.4); MDEM (Minimum Distance-to-Euclidean-mean); QDA su matrici di covarianza vettorizzate (Sezione 4.5).

In generale, non risultano esserci grossi squilibri tra sensibilità e specificità nei modelli considerati, solo le misture Riemanniane e il Minimum Distance-to-Euclidean-mean mostrano delle differenze degne di nota: per il primo metodo si riscontra una maggior efficacia nel riconoscere le osservazioni della classe 2, per il secondo una maggior precisione nei confronti della classe 1.

Il confronto diretto tra i due approcci più semplici, basati solo su distanze da un baricentro per classe, vede risaltare le performance del modello Riemanniano rispetto a quello che utilizza la metrica Euclidea. Il Minimum Distance-to-Riemannian-mean risulta anche essere l'approccio che più spesso ottiene i migliori risultati, questa osservazione verrà approfondita nella Sezione 4.6.2.

Il soggetto 4 e il soggetto 8 sono quelli per cui i modelli forniscono le previsioni migliori. L'unica eccezione è rappresentata dalla QDA che nel primo caso produce la migliore previsione, mentre nel secondo non riesce a ripetere la buona performance.

L'unico modello di classificazione Riemanniano a rimanere quasi sempre in ombra è quello del rapporto di verosimiglianze; la formalizzazione probabilistica dei concetti di baricentro e di dispersione per le osservazioni del *training set* non ha portato benefici a livello di previsioni. È probabile che la stima di un maggior numero di parametri sia collegata con una varianza delle previsioni più alta.

Indice medio	MDRM	LR	Misture	MDEM	QDA
Accuratezza	0.680	0.664	0.674	0.654	0.608
Sensibilità	0.651	0.598	0.580	0.701	0.600
Specificità	0.705	0.673	0.701	0.608	0.584

Tabella 4.2: Risultati medi ottenuti dalla classificazione su tutti i soggetti dell'esperimento 1.2. I modelli considerati sono: MDRM (Minimum Distance-to-Riemannian-mean, Sezione 4.2); LR (Rapporto di verosimiglianze, Sezione 4.3); Misture Riemanniane (Sezione 4.4); MDEM (Minimum Distance-to-Euclidean-mean); QDA su matrici di covarianza vettorizzate (Sezione 4.5)

Nella Tabella 4.2 si nota chiaramente come i modelli che utilizzano le proprietà geometriche dello spazio $\mathcal{P}_3$ siano caratterizzati da indici di accuratezza delle previsioni mediamente più alta dei modelli tradizionali. In generale il movimento immaginato della mano destra è stato riconosciuto in maniera più efficace dai modelli Riemanniani (specificità più alta), il movimento immaginato della mano sinistra è stato individuato più efficacemente dal modello Minimum Distance-to-Euclidean-Mean.

4.6.2 Risultati salienti

Per mostrare visivamente le conseguenze derivanti dal cambio di metrica nella classificazione tramite Minimum Distance-to-Mean, si riporta il grafico di dispersione dello stesso campione di osservazioni, rispetto alle due configurazioni considerate.

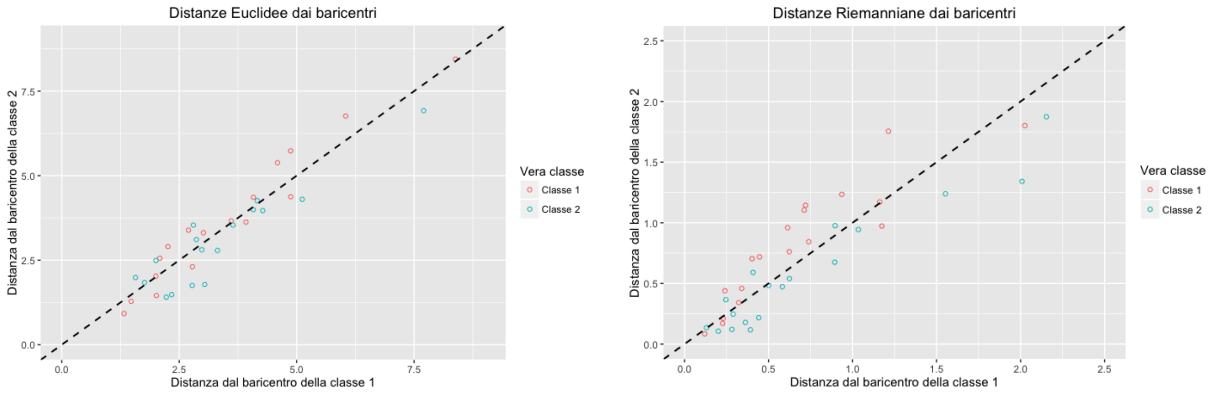

Figura 4.2: Diagramma di dispersione delle osservazioni del *test set* in una iterazione della convalida incrociata per il Soggetto 7. **(a)**: distanze rispetto al baricentro Euclideo; **(b)**: distanze rispetto al baricentro Riemanniano;

Nel primo caso, con la media aritmetica, l'utilizzo esclusivo delle distanze non permette di discriminare sufficientemente le osservazioni dei due stati mentali; i punti colorati sono disposti in modo irregolare e molti di essi non si discostano abbastanza dal margine rappresentato dalla regola di decisione 12, quindi non ci si può attendere un risultato affidabile, in termini generali. Con la metrica Riemanniana la situazione sembra più chiara: le osservazioni sono più distanziate dal margine di decisione e, la maggior parte di esse, risultano più vicine al baricentro della classe corretta.

Qui di seguito sono riportate alcune delle stime ottenute per altri due soggetti dell'esperimento e verranno analizzate nello specifico l'accuratezza delle previsioni del modello Minimum Distance-to-Riemannian-Mean e del modello mistura di Riemann-Gauss con 2 componenti.

Stime per il soggetto 6

- Modello Minimum Distance-to-Riemannian-Mean, baricentri:

$$\hat{\mu}_L = \begin{bmatrix} 32.747 & -25.368 & 2.923 \\ -25.368 & 28.110 & -3.719 \\ 2.923 & -3.719 & 6.271 \end{bmatrix} \quad \hat{\mu}_R = \begin{bmatrix} 31.179 & -24.171 & 3.028 \\ -24.171 & 27.274 & -4.240 \\ 3.028 & -4.240 & 6.655 \end{bmatrix}$$

- Modello Minimum Distance-to-Euclidean-Mean, baricentri:

$$\hat{\mu}_L = \begin{bmatrix} 34.263 & -26.510 & 3.167 \\ -26.510 & 29.368 & -3.975 \\ 3.167 & -3.975 & 6.576 \end{bmatrix} \quad \hat{\mu}_R = \begin{bmatrix} 32.967 & -25.361 & 2.896 \\ -25.361 & 28.933 & -3.989 \\ 2.896 & -3.989 & 7.462 \end{bmatrix}$$

- Modello mistura Riemanniana, pesi delle componenti:

$$\hat{\lambda}_{L1} = 0.043 \qquad \hat{\lambda}_{R1} = 0.010$$
$$\hat{\lambda}_{L2} = 0.957 \qquad \hat{\lambda}_{R2} = 0.990$$

- Modello mistura Riemanniana, baricentri:

$$\hat{\mu}_{L1} = \begin{bmatrix} 33.346 & -26.712 & 3.036 \\ -26.712 & 34.126 & -6.366 \\ 3.036 & -6.366 & 7.708 \end{bmatrix} \quad \hat{\mu}_{R1} = \begin{bmatrix} 59.731 & -57.260 & -33.218 \\ -57.260 & 71.529 & 45.281 \\ -33.218 & 45.281 & 61.804 \end{bmatrix}$$

$$\hat{\mu}_{L2} = \begin{bmatrix} 32.895 & -25.446 & 2.962 \\ -25.446 & 28.027 & -3.675 \\ 2.962 & -3.675 & 6.244 \end{bmatrix} \quad \hat{\mu}_{R2} = \begin{bmatrix} 31.140 & -24.156 & 3.114 \\ -24.156 & 27.237 & -4.349 \\ 3.114 & -4.349 & 6.547 \end{bmatrix}$$

- Modello mistura Riemanniana, dispersione delle componenti:

$$\hat{\sigma}_{L1} = 0.321 \qquad \hat{\sigma}_{R1} = 1.015$$
$$\hat{\sigma}_{L2} = 0.211 \qquad \hat{\sigma}_{R2} = 0.239$$

Stime per il soggetto 9

- Modello Minimum Distance-to-Riemannian-Mean, baricentri:

$$\hat{\mu}_L = \begin{bmatrix} 3.658 & -0.823 & -1.822 \\ -0.823 & 2.761 & 1.797 \\ -1.822 & 1.797 & 6.099 \end{bmatrix} \quad \hat{\mu}_R = \begin{bmatrix} 2.838 & -0.675 & -1.473 \\ -0.675 & 2.755 & 1.751 \\ -1.473 & 1.751 & 6.975 \end{bmatrix}$$

- Modello Minimum Distance-to-Euclidean-Mean, baricentri:

$$\hat{\mu}_L = \begin{bmatrix} 4.086 & -0.852 & -1.943 \\ -0.852 & 2.957 & 1.903 \\ -1.943 & 1.903 & 6.424 \end{bmatrix} \quad \hat{\mu}_R = \begin{bmatrix} 3.048 & -0.698 & -1.545 \\ -0.698 & 2.968 & 1.849 \\ -1.545 & 1.849 & 7.262 \end{bmatrix}$$

- Modello mistura Riemanniana, pesi delle componenti:

$$\hat{\lambda}_{L1} = 0.034 \qquad \hat{\lambda}_{R1} = 0.039$$
$$\hat{\lambda}_{L2} = 0.966 \qquad \hat{\lambda}_{R2} = 0.961$$

- Modello mistura Riemanniana, baricentri:

$$\hat{\mu}_{L1} = \begin{bmatrix} 24.791 & -0.701 & -8.194 \\ -0.701 & 3.731 & 2.555 \\ -8.194 & 2.555 & 9.645 \end{bmatrix} \quad \hat{\mu}_{R1} = \begin{bmatrix} 3.213 & -0.734 & -1.721 \\ -0.734 & 3.014 & 1.836 \\ -1.721 & 1.836 & 6.887 \end{bmatrix}$$

$$\hat{\mu}_{L2} = \begin{bmatrix} 3.577 & -0.818 & -1.797 \\ -0.818 & 2.732 & 1.777 \\ -1.797 & 1.777 & 6.049 \end{bmatrix} \quad \hat{\mu}_{R2} = \begin{bmatrix} 2.833 & -0.671 & -1.466 \\ -0.671 & 2.756 & 1.749 \\ -1.466 & 1.749 & 6.981 \end{bmatrix}$$

- Modello mistura Riemanniana, dispersione delle componenti:

$$\hat{\sigma}_{L1} = 0.787 \qquad \hat{\sigma}_{R1} = 0.389$$
$$\hat{\sigma}_{L2} = 0.272 \qquad \hat{\sigma}_{R2} = 0.247$$

Tutti i valori riportati sono da intendersi come media delle stime fornite dai modelli di *training* in ognuna delle iterazioni della convalida incrociata.

Le medie di classe ottenute nell'ipotesi di un unico baricentro per entrambe non mostrano drammatiche differenze quando sono ottenute come media geometrica (algoritmo 2) o tramite semplice media aritmetica del campione. Questo suggerisce che la differenza nelle performance di previsione dei due modelli Minimum Distance-to-Mean sia dovuta principalmente al tipo di distanza sfruttata.

Spostando l'attenzione ai modelli mistura, emerge che oltre alla componente con peso maggiore (che risulta qualitativamente simile a quella ottenuta nel modello MDRM), l'algoritmo EM produce la stima di un'altra distribuzione di Riemann-Gauss con dispersione più alta e peso molto inferiore rispetto alla prima. Questo fenomeno è compatibile con la tipologia di distribuzione mostrata nella Figura 4.1, in cui era suggerita la presenza di una piccola porzione del campione che il singolo baricentro non era in grado di catturare.

Secondo quanto riportato nella Tabella 4.1, il modello di classificazione basato sulle misture produce la miglior accuratezza per il soggetto 9; i valori dei baricentri, in questo caso, risultano almeno parzialmente diversi nelle due condizioni e la maggior flessibilità della mistura può aver contribuito ad ottenere tale esito.
Nel soggetto 6 i baricentri stimati delle componenti più pesanti sono molto simili tra loro e anche i valori di $\hat{\sigma}_{L2}$ e $\hat{\sigma}_{R2}$ sono vicini; in questo caso la semplicità del modello MDRM (approccio non parametrico di tipo *nearest neighbor*) può essere stata la caratteristica che ha permesso a tale modello di ottenere la migliore performance.

Conclusioni e sviluppi futuri

Il lavoro descritto in questa tesi ha avuto come filo conduttore la geometria Riemanniana; le definizioni e i teoremi recuperati nella fase di ricerca bibliografica sono stati gli strumenti su cui si è basata la successiva costruzione di modelli che potessero essere applicati ad un problema reale di classificazione dell'immaginazione di movimenti.
L'idea che si è cercato di seguire è stata di ripercorrere la proposta metodologica di Said et al. (2015), Congedo et al. (2016), Lotte (2014), Barachant et al. (2013) attraverso la comprensione dei loro articoli e lo sviluppo del codice R basato sul lavoro di Zanini et al. (2016). Grazie alla disponibilità di dati da *trials* EEG è stato anche possibile verificare la validità del metodo proposto nel contesto delle BCIs.
Le considerazioni che si possono ricavare alla fine di questo percorso sono numerose e di diversa natura, per questo motivo si è cercato di riassumere le principali:

- come spesso succede nei problemi di classificazione non esiste un modello che sia sempre superiore agli altri; in base alle caratteristiche del segnale acquisito (impossibile da conoscere a priori) può risultare migliore un approccio non parametrico meno vincolato da strutture probabilistiche o viceversa;

- è stata confermata la validità dell'approccio di *training* personalizzato per ogni soggetto: la variabilità riscontrata a livello di dati dei diversi soggetti è troppo alta per definire modelli per campioni più generali;

- l'approccio della mistura si è confermato il più flessibile nel riassumere campioni caratterizzati da baricentro non unico, anche se questa situa-

zione si è verificata solo per particolari soggetti. Negli altri la maggiore complessità parametrica non ha giocato a favore della correttezza delle previsioni;

- per un utilizzo concreto di tali tecniche c'è la necessità di una maggiore accuratezza nelle previsioni; il pre-processamento del segnale dovrebbe massimizzare le differenze tra la distribuzione spaziale delle osservazioni in modo da facilitare il compito degli algoritmi di classificazione, ma non si è rivelato molto efficace nel caso considerato;

- in ogni caso i risultati ottenuti sono incoraggianti: le indicazioni fornite dagli indici di bontà delle previsioni evidenziano come la geometria Riemanniana, inserita nei modelli per matrici di covarianza, ha la possibilità di essere un fattore determinante per una previsione più efficace degli stati mentali nelle Motor Imagery-based BCIs.

L'aver acquisito maggiore consapevolezza riguardo alle potenzialità e ai limiti della geometria Riemanniana applicata alla classificazione degli stati mentali mi consente di suggerire quelle che, a mio parere, sono le possibilità di sviluppo delle tecniche utilizzate in questa tesi:

- il pre-processamento dell'EEG è una fase del lavoro che, alla luce dei risultati, viene ritienuta fondamentale per una buona classificazione; per questa ragione sarebbe utile l'approfondimento del tema *filtraggio spaziale*, solo accennato nella Sezione 1.3.2 (non era l'obiettivo di questo lavoro);

- la stima del fattore di normalizzazione $\zeta(\sigma)$ risulta l'aspetto computazionalmente più oneroso. Dato l'interesse nel sviluppare un approccio robusto all'aumento del numero di canali, sarebbe utile trovare una procedura più veloce di stima di tale quantità; ad esempio si può indagare la relazione esistente tra il determinante della matrice di distanze Riemanniane e $\zeta(\sigma)$.

Appendice A

Importazione MATLAB

Il file originale contenente tutti i dataset forniti dall'Università di Graz veniva fornito in formato GDF, acronimo di *General Data Format*, un formato standard per segnali biomedici [Schlögl, 2006].

In una prima fase, questo file è stato caricato all'interno del software MATLAB[1] utilizzando il toolbox gratuito BioSig[2]. Il comando MATLAB utilizzato per importare ognuna delle sessioni di registrazione considerate nella tesi è il seguente:

```
[s,h] = sload('nome_file.gdf');
```

In questo modo si producono due variabili, **s** e **h**. La prima contiene il segnale grezzo, sotto forma di un'unica matrice con 6 colonne (le prime 3 corrispondenti ai canali EEG e le ultime tre ai canali EOG) e numero di righe che dipende dalla lunghezza del periodo di registrazione. La seconda è costituita da un insieme di strutture dal quale si possono estrarre le informazioni che consentono di trattare i dati in **s**. In particolare, **h** contiene informazioni relative alla struttura del segnale nel tempo, le più importanti sono:

- `h.EVENT.POS`
- `h.EVENT.TYP`
- `h.Classlabel`

[1] versione R2014b, The MathWorks, Inc., Natick, MA, USA
[2] http://biosig.sourceforge.net/

- `h.ArtifactSelection`

La posizione degli eventi rilevanti ai fini dell'esperimento, intesa come numero di riga nella matrice `s`, è contenuta in `h.EVENT.POS`. La corrispondente descrizione si trova in `h.EVENT.TYP`; queste due variabili sono dei vettori con la stessa dimensione. Tra le tipologie di eventi riportate, quelle rilevanti per l'analisi comprendono l'inizio e la fine di ogni *trial* e la tipologia di movimento richiesta ai soggetti.
I *trials* influenzati da artefatti presentano il valore 1 in `h.ArtifactSelection`.

La matrice contenente il segnale grezzo originale `s` e i vettori di intestazione di interesse sono stati salvati in un file con estensione `.mat` in modo da poterli esportare.

Appendice B

Codice R utilizzato

B.1 *Minimum Distance-to-Mean*

Codice B.1: Distanza Riemanniana di Rao

```
library(powerplus)
library(expm)

distanceRao2 <- function(c,m){
  sum(log(eigen(solve(c)%*%m)$values)^2)
  # matrix.trace(Matpow(logm(Matpow(c,-1,2) %*% m %*% Matpow(c,-1,2)),2)) metodo alternativo
}

distance <- distanceRao2(matrix1,matrix2)
# le due matrici in input sono simmetriche, quadrate e semi-definite positive
```

Codice B.2: Stima della media geometrica di un campione di matrici

```
library(powerplus)
library(expm)

initNegP <- function (c,p,sampdim){  # funzione ausiliaria, inizializzazione di X per valori p
     negativi
  c <- (1/sampdim)*solve(Matpow(c,1,1/p))
}

initPosP <- function (c,p,sampdim){    # inizializzazione per esponenti p positivi
  c <- (1/sampdim)*Matpow(c,1,1/p)
}

hNeg <- function(C,X,sampdim,p){   # funzione ausiliaria, calcolo di H
  h1 <- (1/sampdim)*Matpow((X%*%solve(C)%*%t(X)),1,1/abs(p))
```

```
}

hPos <- function(C,X,sampdim,p){      # calcolo di H per p positivi
  h1 <- (1/sampdim)*Matpow((X%*%C%*%t(X)),1,1/abs(p))
}

powerM <- function(covlist,X,phi,sampdim,p){  # funzione ausiliaria, aggiornamento matrice X
  if (p<0) l1 <- lapply(covlist,hNeg,X,sampdim,p)
  else  l1 <- lapply(covlist,hPos,X,sampdim,p)

  H <- Reduce('+',l1)
  X <- solve(Matpow(H,phi))%*%X
  res <- list("X"=X, "H"=H)
  return(res)
}

MeanP <- function(covlist, zeta, eps, p){      # funzione principale
  phi <- eps/p
  sampdim <- length(covlist)
  if (p<0) l <- lapply(covlist,initNegP,p,sampdim)
    else l <- lapply(covlist,initPosP,p,sampdim)
  p0 <- Reduce('+',l)
  if (p<0) p0 <- Matpow(p0,1,p)
    else p0 <- solve(Matpow(p0,1,p))     # ottengo P0
  X <- expm::sqrtm(p0)     # ottengo la matrice X da usare nel primo ciclo
  m <- dim(covlist[[1]])[1]
  repeat{
    res <- powerM(covlist,X,phi,sampdim,p)
    X <- res$X
    print(norm((res$H-diag(m))))
    if ((1/sqrt(m)*norm((res$H-diag(m)),"F"))<zeta) break  # condizione di uscita dal loop
  }
  X <- res$X
  res <- t(X)%*%X  # ottengo la matrice baricentro con ordine p
  return(res)
}
MNeg <- MeanP(covlist, zeta=1e-8, eps=1/3, p=-1e-4)
# chiamata della funzione principale, covlist lista di matrici di covarianza delle quali voglio
    calcolare la media geometrica
MPos <- MeanP(covlist, zeta=1e-8, eps=1/3, p=1e-4) # stima per p positivo

m <- dim(covlist[[1]])[1]
x <- array(c(MNeg,MPos), dim=c(m,m,2))
FinalMean <- apply(x,c(1,2),mean)    # ottengo la stima finale come media aritmetica delle due
    matrici calcolate in precedenza

# lo stesso procedimento va applicato per tutte le classi sui rispettivi campioni di training
```

B.2 Distribuzione di Riemann-Gauss

Codice B.3: Stima del fattore di normalizzazione della distribuzione di Riemann-Gauss

```
library(MASS)

gamma_p <- function(x,p){  # funzione ausiliaria, calcolo Gamma multivariata
  tmp <- 1
  for (j in 1:p) tmp <- tmp*gamma(x+(1-j)/2)
  res <- pi^(p*(p-1)/4)*tmp
  return(res)
}


zetaMC <- function(eta,m,N){ # stima del fattore di normalizzazione tramite integrazione Monte Carlo
  mu <- rep(0,m)
  Sigma <- diag(-1/eta,m)
  set.seed(456)  # stesso seme per stabilizzare errore MC
  k <- mvrnorm(N,mu,Sigma)
  tmp <- rep(1,N)
  for (i in 1:(m-1))
    for (j in (i+1):m)
      tmp <- tmp*(sinh(abs(k[,i]-k[,j])/2))
  omegap <- ((((pi^((m^2)/2))*8^((m*(m-1))/4)))/(factorial(m)*gamma_p(m/2,m)))
  res <- omegap*mean(tmp)
  res <- res*(2*pi)^(m/2)*(det(Sigma))^(1/2)
  return(res)
}

zetaMC(eta,m,N) # chiamata alla funzione: eta <- -1/2*sigma^2, m numero di canali EEG considerati e N
      numero distribuzioni MC generate
```

Codice B.4: Stima del parametro di dispersione per la distribuzione di Riemann-Gauss

```
compt <- seq(1,2000)^2  # simulazione di zeta(eta) per un campione numeroso
eta <- -10^6/compt
zeta <- rep(NA,2000)
zeta <- sapply(eta,zetaMC,m=3,100000)


psi_smooth <- smooth.spline(eta,log(zeta),all.knots=T) # lisciamento della funzione ottenuta
logzetaprime <- predict(psi_smooth,der=1) # derivata della funzione psi
etahat <- logzetaprime$x[which.min(abs(dist-logzetaprime$y))] # criterio di stima di eta: il valore
     della derivata di psi che piu si avvicina alla varianza empirica calcolata nel baricentro
sigma_hat <- sqrt(-1/2*etahat) # ritorno alla parametrizzazione sigma (invarianza SMV)
```

B.3 Mistura di Riemann-Gauss

Codice B.5: Calcolo della densità di Riemann-Gauss

```
riemannGauss <- function(x,bar,sig,p){
  (1/zetaMC(-1/2*sig^2,p,10000))*exp(-(distanceRao2(x,bar)/(2*sig^2)))
}
```

Codice B.6: Funzione ausiliaria - gestione della matrice dei pesi della mistura

```
pi_k <- function(cov,bar,sigma,w){  # in input baricentri, dispersione e pesi da aggiornare
  I <- length(cov)
  K <- length(bar)
  p <- dim(cov[[1]])[1]
  pi_k <- matrix(NA,I,K)
  Nk <- rep(NA,K)
  Nn <- rep(NA,I)
  for (k in 1:K){  # inizializzo matrice pi_k
    for(i in 1:I) pi_k[i,k] <- w[k]*riemannGauss(cov[[i]],bar[[k]],sigma[k],p)
  }
  Nn <- apply(pi_k,1,sum)

  for (k in 1:K){
    for(i in 1:I) pi_k[i,k] <- (1/Nn[i])*pi_k[i,k]
  }
    return(pi_k)
  }
```

Codice B.7: Algoritmo EM di stima dei parametri della mistura di Riemann-Gauss

```
mixest <- function(covlist,K,Nmax){ # in input numero di componenti e iterazioni massime
  iter <- 1
  N <- length(covlist)
  Nmu <- numeric()
  p <- dim(covlist[[1]])[1]
  wj <- rep(1/K,K)
  barhat <- list()
  sigmahat <- rep(0,K)
  res <- list()
  for(i in 1:K){   # inizializzazione parametri della mistura
    barhat[[i]] <- covlist[[sample(1:N,1)]]
    d <- lapply(covlist,distanceRao2,barhat[[i]])
    dl <- 1/N*Reduce('+',d)
    sigmahat[i] <- sqrt(-1/logzetaprime$x[which.min(abs(distance-logzetaprime$y))])
  }
  tolwj <- rep(1,K)
  tolsig <- rep(1,K)
  while (any(tolwj>10e-5) | any(tolsig>10e-5))
  {  # condizione di uscita dal ciclo
```

```
    wj_new <- numeric()
    pik <- pi_k(covlist,barhat,sigmahat,wj)
    for(i in 1:K)   Nmu[i] <- sum(pik[,i])
    for(i in 1:K)   wj_new[i] <- Nmu[i]/sum(Nmu)
    pik <- pi_k(covlist,barhat,sigmahat,wj_new)
    for(i in 1:K)   Nmu[i] <- sum(pik[,i])

    barhat_new <- list()
    sigmahat_new <- numeric()
    for(i in 1:K) {
    barhat_new[[i]] <- MeanPik(covlist,i,Nmu,pik) # chiamata alla funzione ausiliaria che stima il
         baricentro della componente generalizzando i pesi del B.2
    d <- lapply(covlist,distanceRao2,barhat_new[[i]])
    distance <- 1/N*Reduce('+',d)
    sigmahat_new[i] <- sqrt(-1/logzetaprime$x[which.min(abs(distance/2-logzetaprime$y))])
    }   # aggiornamento delle stime

    tolwj <- 100*(abs((wj_new-wj)/wj))
    tolsig <- 100*(abs(sigmahat_new-sigmahat)/sigmahat)
    print(tolwj)
    print(tolsig)

    wj <- wj_new
    barhat <- barhat_new
    sigmahat <- sigmahat_new
    print(wj)
    print(barhat)
    print(sigmahat)
    iter <- iter+1
    if (any(wj<10e-4)) break  # interrompi se una componente ha un peso troppo piccolo (risultato
         comune con piu di 2 componenti)
    if (iter>Nmax) break
  }
  res$wj <- wj
  res$barhat <- barhat
  res$sigmahat <- sigmahat
  return(res)
}

res <- mixest(covlist,k,1000)
```

Bibliografia

Afsari, Bijan (2011). Riemannian Lp center of mass: existence, uniqueness, and convexity. In: *Proceedings of the American Mathematical Society* 139.2, 655–673.

Ang, Kai Keng, Chin, Zheng Yang, Wang, Chuanchu, Guan, Cuntai e Zhang, Haihong (2012). Filter Bank Common Spatial Pattern Algorithm on BCI Competition IV Datasets 2a and 2b. In: *Frontiers in Neuroscience* 6, 39. ISSN: 1662-453X. DOI: 10.3389/fnins.2012.00039.

Atkinson, Colin e Mitchell, Ann FS (1981). Rao's distance measure. In: *Sankhyā: The Indian Journal of Statistics, Series A*, 345–365.

Azzalini, Adelchi (2004). Inferenza statistica: una presentazione basata sul concetto di verosimiglianza. Springer Science & Business Media.

Azzalini, Adelchi e Scarpa, Bruno (2012). Data analysis and data mining: An introduction. OUP USA.

Banerjee, Monami, Chakraborty, Rudrasis, Ofori, Edward, Vaillancourt, David e Vemuri, Baba C (2015). «Nonlinear regression on riemannian manifolds and its applications to neuro-image analysis». In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 719–727.

Barachant, Alexandre, Bonnet, Stéphane, Congedo, Marco e Jutten, Christian (2013). Classification of covariance matrices using a Riemannian-based kernel for BCI applications. In: *Neurocomputing* 112, 172–178.

Bashashati, Ali, Fatourechi, Mehrdad, Ward, Rabab K e Birch, Gary E (2007). A survey of signal processing algorithms in brain–computer interfaces based on electrical brain signals. In: *Journal of Neural engineering* 4.2, R32.

Berlin Brain-Computer Interface. URL: http://www.bbci.de.

Bishop, Christopher M. (2006). Pattern Recognition and Machine Learning (Information Science and Statistics). Secaucus, NJ, USA: Springer-Verlag New York, Inc. ISBN: 0387310738.

Blankertz, Benjamin, Sannelli, Claudia, Halder, Sebastian, Hammer, Eva M, Kübler, Andrea, Müller, Klaus-Robert, Curio, Gabriel e Dickhaus, Thorsten (2010). Neurophysiological predictor of SMR-based BCI performance. In: *Neuroimage* 51.4, 1303–1309.

Borg, Ingwer e Groenen, Patrick JF (2005). Modern multidimensional scaling: Theory and applications. Springer Science & Business Media.

Congedo, Marco, Goyat, Matthieu, Tarrin, Nicolas, Ionescu, Gelu, Varnet, Léo, Rivet, Bertrand, Phlypo, Ronald, Jrad, Nisrine, Acquadro, Michael e Jutten, Christian (2011). «" Brain Invaders": a prototype of an open-source P300-based video game working with the OpenViBE platform». In: *5th International Brain-Computer Interface Conference 2011 (BCI 2011)*, pp. 280–283.

Congedo, Marco, Phlypo, Ronald e Barachant, Alexandre (2016). «A fixed-point algorithm for estimating power means of positive definite matrices». In: *Signal Processing Conference (EUSIPCO), 2016 24th European*. IEEE, pp. 2106–2110.

Croft, RJ e Barry, RJ (2000). Removal of ocular artifact from the EEG: a review. In:

Fatourechi, Mehrdad, Bashashati, Ali, Ward, Rabab K e Birch, Gary E (2007). EMG and EOG artifacts in brain computer interface systems: A survey. In: *Clinical neurophysiology* 118.3, 480–494.

Fisher, Ronald A (1936). The use of multiple measurements in taxonomic problems. In: *Annals of eugenics* 7.2, 179–188.

Friedman, Jerome H (1997). On bias, variance, 0/1—loss, and the curse-of-dimensionality. In: *Data mining and knowledge discovery* 1.1, 55–77.

Ledoit, Olivier e Wolf, Michael (2004). A well-conditioned estimator for large-dimensional covariance matrices. In: *Journal of multivariate analysis* 88.2, 365–411.

Leeb, Robert, Lee, Felix, Keinrath, Claudia, Scherer, Reinhold, Bischof, Horst e Pfurtscheller, Gert (2007). Brain–computer communication: motivation,

aim, and impact of exploring a virtual apartment. In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 15.4, 473–482.

Lenglet, Christophe, Rousson, Mikaël, Deriche, Rachid e Faugeras, Olivier (2006). Statistics on the manifold of multivariate normal distributions: Theory and application to diffusion tensor MRI processing. In: *Journal of Mathematical Imaging and Vision* 25.3, 423–444.

Leroux, Brian G et al. (1992). Consistent estimation of a mixing distribution. In: *The Annals of Statistics* 20.3, 1350–1360.

Lotte, Fabien (2014). «A tutorial on EEG signal-processing techniques for mental-state recognition in brain–computer interfaces». In: *Guide to Brain-Computer Music Interfacing*. Springer, pp. 133–161.

Lotte, Fabien, Congedo, Marco, Lécuyer, Anatole, Lamarche, Fabrice e Arnaldi, Bruno (2007). A review of classification algorithms for EEG-based brain–computer interfaces. In: *Journal of neural engineering* 4.2, R1.

Maaten, Laurens van der e Hinton, Geoffrey (2008). Visualizing data using t-SNE. In: *Journal of Machine Learning Research* 9.Nov, 2579–2605.

Moakher, Maher (2005). A differential geometric approach to the geometric mean of symmetric positive-definite matrices. In: *SIAM Journal on Matrix Analysis and Applications* 26.3, 735–747.

Mühl, Christian, Jeunet, Camille e Lotte, Fabien (2015). EEG-based workload estimation across affective contexts. In: *Using Neurophysiological Signals that Reflect Cognitive or Affective State*, 227.

Muirhead, Robb J (1982). Aspects of multivariate statistical analysis. In: *JOHN WILEY & SONS, INC., 605 THIRD AVE., NEW YORK, NY 10158, USA, 1982, 656.*

Niedermeyer, Ernst e Silva, FH Lopes da (2005). Electroencephalography: basic principles, clinical applications, and related fields. Lippincott Williams & Wilkins.

Raudys, Sarunas J, Jain, Anil K et al. (1991). Small sample size effects in statistical pattern recognition: Recommendations for practitioners. In: *IEEE Transactions on pattern analysis and machine intelligence* 13.3, 252–264.

Rosu, Roxana, Donias, Marc, Bombrun, Lionel, Said, Salem, Regniers, Olivier e Da Costa, Jean-Pierre (2017). Structure Tensor Riemannian Statistical

Models for CBIR and Classification of Remote Sensing Images. In: *IEEE Transactions on Geoscience and Remote Sensing* 55.1, 248–260.

Said, Salem, Bombrun, Lionel, Berthoumieu, Yannick e Manton, Jonathan (2015). Riemannian Gaussian distributions on the space of symmetric positive definite matrices. In: *arXiv preprint arXiv:1507.01760*.

Said, Salem, Hajri, Hatem, Bombrun, Lionel e Vemuri, Baba C (2016). Gaussian distributions on Riemannian symmetric spaces: statistical learning with structured covariance matrices. In: *arXiv preprint arXiv:1607.06929*.

Schlögl, Alois (2006). GDF-a general dataformat for biosignals. In: *arXiv preprint cs/0608052*.

Schlogl, Alois, Keinrath, Claudia, Zimmermann, Doris, Scherer, Reinhold, Leeb, Robert e Pfurtscheller, Gert (2007). A fully automated correction method of EOG artifacts in EEG recordings. In: *Clinical neurophysiology* 118.1, 98–104.

Schwarz, Gideon et al. (1978). Estimating the dimension of a model. In: *The annals of statistics* 6.2, 461–464.

Stahl, Saul (2006). The evolution of the normal distribution. In: *Mathematics magazine* 79.2, 96–113.

Terras, Audrey (2012). Harmonic analysis on symmetric spaces and applications II. Springer Science & Business Media.

Titterington, DM, Smith, AFM e Makov, UE (1985). (Statistical Analysis of Finite Mixture Distributions.) John Wiley and Sons. In: *New York*.

Urigüen, Jose Antonio e Garcia-Zapirain, Begoña (2015). EEG artifact removal—state-of-the-art and guidelines. In: *Journal of neural engineering* 12.3, 031001.

Van Vliet, Marijn (2006). Effectiveness of Automatic EOG Regression. In: *University of Twente, Netherlands*.

Wickelmaier, Florian (2003). An introduction to MDS. In: *Sound Quality Research Unit, Aalborg University, Denmark* 46.

Wikipedia, Elettroencefalografia. URL: https://it.wikipedia.org/wiki/Elettroencefalografia.

Wikipedia, Elettroencefalography. URL: https://commons.wikimedia.org/w/index.php?curid=10827060.

Wikipedia, Spike-and-wave. URL: https://it.wikipedia.org/w/index.php?curid=338741.

Wolpaw, Jonathan e Wolpaw, Elizabeth Winter (2012). Brain-computer interfaces: principles and practice. OUP USA.

Wolpaw, Jonathan R, McFarland, Dennis J, Neat, Gregory W e Forneris, Catherine A (1991). An EEG-based brain-computer interface for cursor control. In: *Electroencephalography and clinical neurophysiology* 78.3, 252–259.

Zander, Thorsten O e Kothe, Christian (2011). Towards passive brain–computer interfaces: applying brain–computer interface technology to human–machine systems in general. In: *Journal of neural engineering* 8.2, 025005.

Zanini, Paolo, Congedo, Marco, Jutten, Christian, Said, Salem e Berthoumieu, Yannick (2016). «Parameters estimate of Riemannian Gaussian distribution in the manifold of covariance matrices». In: *9th IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM 2016)*. Rio de Janeiro, Brazil. URL: https://hal.inria.fr/hal-01325055.