UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Sede Amministrativa: Università degli Studi di Padova

Dipartimento di Scienze Statistiche
Corso di Dottorato di Ricerca in Scienze Statistiche
Ciclo XXX

Classification Approaches in Neuroscience: A Geometrical Point of View

Coordinatore del Corso: Prof. Nicola Sartori

Supervisore: Prof. Livio Finos

Co-supervisore: Prof. Bruno Scarpa

Dottorando: Ehsan Kharati Koopaei

06/10/2017

Abstract

Functional magnetic resonance images (fMRI) are brain scan images by MRI machine which are taken functionally cross the time. Several studies have investigated methods analyzing such images (or actually the drawn data from them) and is interestingly growing up. For examples models can predict the behaviours and actions of people based on their brain pattern, which can be useful in many fields. We do the classification study and prediction of fMRI data and develop some approaches and some modifications on them which have not been used in such classification problems. The proposed approaches were assessed by comparing the classification error rates in a real fMRI data study and the merits of our proposed methods are shown. In addition, many programming codes for reading from fMRI scans and codes for using classification approaches are provided to manipulate fMRI data in practice. The codes, can be gathered later as a package in R.

Also, there is a steadily growing interest in analyzing functional data which can often exploit Riemannian geometry. As a prototypical example of these kind of data, we will consider the functional data rising from an electroencephalography (EEG) signal in Brain-Computer interface (BCI) which translates the brain signals to the commands in the machine. It can be used for people with physical inability and movement problems or even in video games, which has had increased interest. To do that, a classification study on EEG signals has been proposed, while the data in hand to be classified are matrices. A multiplicative algorithm (MPM), which is a fast and efficient algorithm, was developed to compute the power means for matrices which is the crucial step in our proposed approaches for classification. In addition, some simulation studies were used to examine the performance of MPM against existing algorithms and the behavior of different power means in terms of accuracy are compared in our classifications, which had not been discovered previously. We will show that it is difficult to have a guess

to find the optimal power mean to have higher accuracy depending on the multivariate distribution of available data. Then, an approach which is combination of power means is also developed to have the benefit of all to improve the classification performance. All the codes related to the fast MPM algorithms and the codes for manipulating EEG signals in classification are written in MATLAB and can be developed later as a toolbox.

Sommario

Le immagini da risonanza magnetica funzionale (functional magnetic resonance image - fMRI) sono immagini di scansioni cerebrali effettuate tramite la macchina MRI prese come funzione del tempo. Negli ultimi anni sta crescendo l'interesse sull'analisi di queste immagini, o meglio dei dati da loro estratti. L'obiettivo di questo tipo di analisi, applicabile in molti ambiti diversi, è quello di stimare e prevedere i comportamenti e le azioni delle persone a partire dai loro pattern cerebrali. Il nostro lavoro si basa sulla classificazione e previsione dei dati fMRI e sullo sviluppo di nuove tecniche che non sono mai state applicate a questi problemi di classificazione. La validazione delle tecniche proposte è stata effettuata tramite il confronto degli errori di misclassificazione su dati fMRI provenienti da studi reali. Inoltre, vengono forniti i codici di lettura dalle immagini fMRI ed quelli per applicare le tecniche di classificazione proposte per la manipolazione dei dati fMRI. In futuro i codici potranno essere organizzati per la creazione di un pacchetto R.

L'interesse nell'analisi di dati funzionali che utilizzano la geometria riemanniana è in costante crescita. Un prototipo di questi dati consiste nei dati funzionali generati dal segnale EEG nell'interfaccia Brain-Computer (BCI), la quale traduce i segnali cerebrali ai comandi nella macchina. Il BCI può essere utilizzato da persone con inabilità fisiche e problemi motori o persino, con crescente interesse, nell'ambito dei video giochi. A questo scopo, abbiamo proposto uno studio di classificazione dei segnali EEG i cui dati sono raccolti in matrici. Abbiamo sviluppato un algoritmo moltiplicativo (MPM) veloce ed efficiente nel calcolare le medie di potenza di matrici, punto cruciale dei metodi proposti per la classificazione. In alcuni studi di simulazione abbiamo esaminato le performance del MPM rispetto a quelle di algoritmi già esistenti. Abbiamo inoltre comparato il coportamento di diverse medie di potenza in termini di accuratezza delle classificazioni, cosa che non era stato mai fatta fino ad ora. Abbiamo verificato la difficolt di scegliere la potenza associata con la migliore accuratezza del modello poichè

questa dipende dalla distribuzione multivariata dei dati. Inoltre abbiamo sviluppato un approccio basato sulla combinazione di medie di potenza per poter beneficiare e per migliorare le performance di classificazione. Tutti i codici relativi all' algoritmo MPM veloce e quelli per la manipolazione dei segnali EEG nella classificazione sono scritti in MATLAB e possono essere sviluppati successivamente per la creazione di un pacchetto.

To my dad and my mom
my uniformly most powerful tests.

Acknowledgements

First, I would like to give my special thanks to whom contributed to helping me grow up: my dad and mom. However, I have lost the warm support of my father over the past six years, but, if I did not have encouragement from my mom to start this difficult period of my life as a PhD, I would not have been able to be at this position, almost at the graduation. Beside them, I was the last kid between five more brothers and one lovely sister. During whole of my educational and research life, I always had one of them as an expert of my problems beside myself to help me solve my homework and teach me what I did not understand in lectures. Now, they have earned their PhD and they are professors in universities in different fields such as biology, mechanical engineering, electrical engineering, statistics, religious studies and genetic engineering! I am the last chain of this long queue.

Of cours, Livio has played a major role in guiding me in this work. This was my first experience working with a non-Iranian supervisor, and I was very lucky to meet him. He is patient, smart, always full of ideas and working beside you! not behind!

I also would like to thank Marco Congedo and Bruno Scarpa. In my productive 3-month study visit in Grenoble in France, Marco really helped to open my eyes to the correct method of publishing papers in scientific world and I am still learning from those lessons. Bruno was like a father in research life, to set you in the right way to be efficient in the work. I don't remember I left his office without finding the idea for my problems; never! I am also grateful for my brother Mahmood, who is now an associate professor in statistics in Iran. Regardless of his own busy schedule, he was always available for me and greatly helped me in my PhD the past three years. I would also like to show my appreciation for Monica and Nicola, our PhD coordinators. I always found them as a strong supporter of us, and I always felt free to share all my concerns with them.

I want to give my special thanks to Patrizia Piacentini, the PhD secretary in our department, who is a very kind and patient lady who was always helping me to overcome my concerns, even with entrance visa problems, before we met in Padova. Lastly, I would like to thank my lovely colleagues in XXX cycle , which is where I had my best experiences and developed many wonder friendships.

Contents

List of Figures

List of Tables

Introduction

Overview

The digital era that has begun in the last decades stimulates the production of a paramount quantity of data. Because of this revolution, the work of the statisticians changed due to the economic and powerful tools that she/he can use. Most of all, the work of statistics has changed because of the kind of data that is requested to analyzed. This does not simply mean the well-known characteristic of the new-generation dataset that are usually "big". The "big data" issue is well recognised by the statistics community, and many efforts have been made to study such problems. More interestingly, the digital era stimulates the production of more complex data. For example, functional data is among the most well formalized and studied one. Functional data is very common in various fields, like, signals, stock market, imaging, traffic, internet, etc. Such amount of data, stimulates the research in the geometry and statistics fields. However, in many practical analyses that use statistical tools, researchers are encountering with a kind of huge amount of data which is called big data in scientific term in the sense that the number of variables are big, much larger than the observations.
Functional magnetic resonance images (fMRI) are the brain scan images by MRI machine which are taken functionally over the time. Analyzing such images (or actually the drawn data from) is interestingly growing up. For example, studies have developed models capable of modeling and predicting people behaviors and actions based on their brain pattern, which can be useful in many fields. During a fMRI experiment, a series of brain images are taken while the subject is doing a task in specific time segments (see Figure 1). Each image comprises roughly 100,000 voxels, i.e., a cubic unit in the 3D brain volume, as shown in Figure 2. Each brain volume comprises three slices: coronal, sagittal, and axial. In addition, each voxel has its own time series for each subject, as shown in Figure 3.

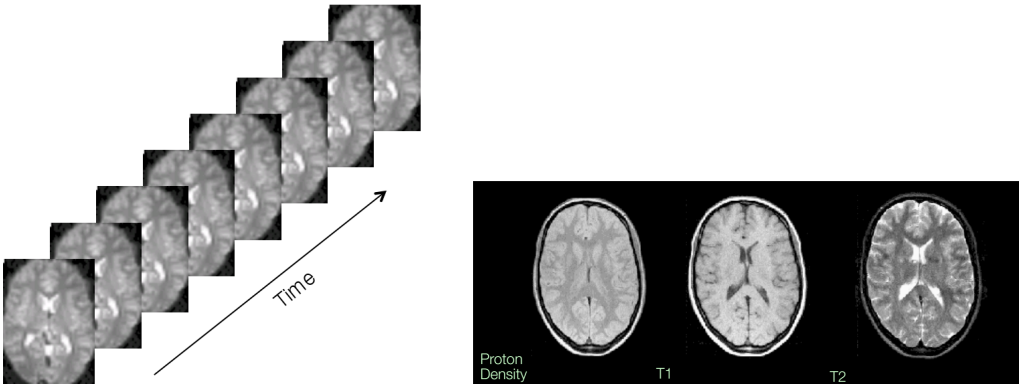


FIGURE 1: fMRI (left) and MRI (right) images of the brain of a case study.

Brain imaging can be used to show different type of issues by using captured neuron signals. Typically, there are two major issues in studying brain images: spacial resolution which can be used to make inference on different parts of brain which are activated during a stimuli in a fix time point and temporal resolution which is related to images on different time points during a stimuli. The latter is our main focus in this research. Facing a stimuli, neurons in different voxels are active; consequently, they access to oxygen. BOLD (Blood Oxygenation Level Dependent) is the usual method used to interpret neuronal activities and it measures the ratio of oxygenated to deoxygenated hemoglobin in the blood. Note that it does not directly measure neuronal activity.

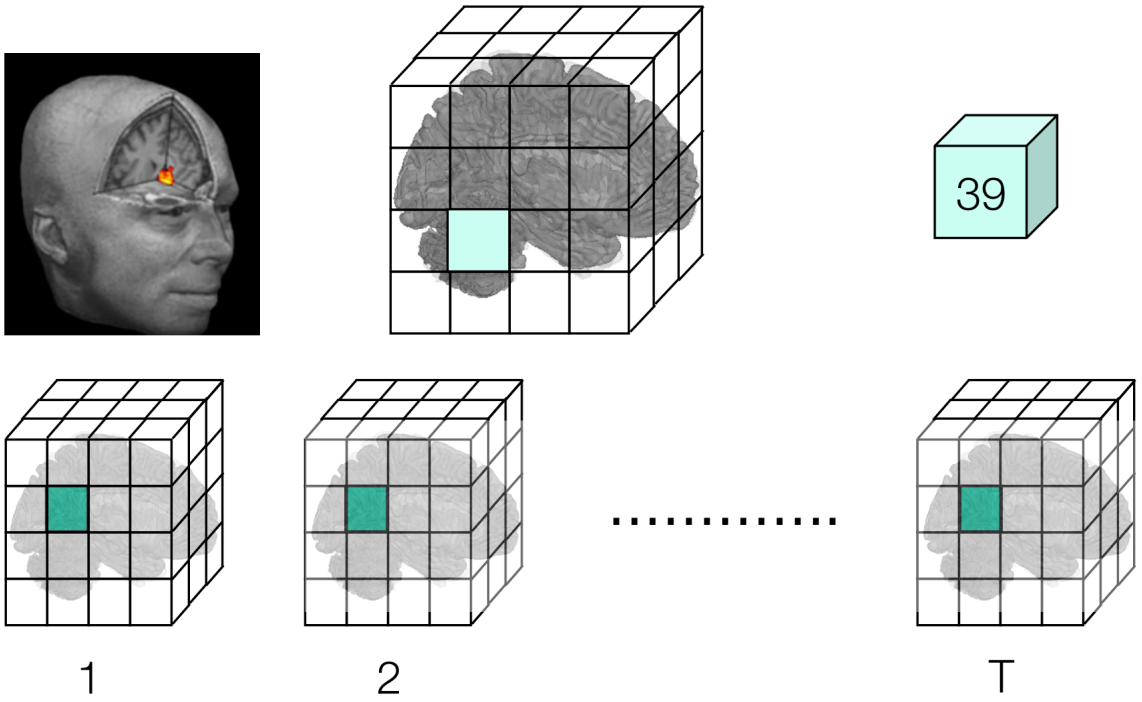


FIGURE 2: Schematic of voxels (over time) in a brain.

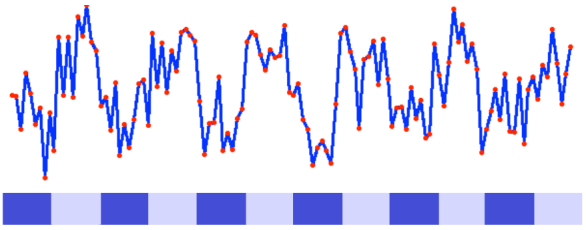


FIGURE 3: Time series of computed BOLD for voxels of a fMRI over a segment of time for a case study brain.

An fMRI experiment may contain many subjects while there are different runs for a specific stimuli. Moreover, each run consists of a series of brain volumes which are made up by multiple slices, and each slice contains many voxels. Consequently, fMRI data is a big data problem. There are many studies on fMRI in the literature of the different science fields. For example Spiridon and Kanwisher (2002), Cox and Savoy (2003), Tsao *et al.* (2006), Hung *et al.* (2005), Kiani *et al.* (2007) and Brants *et al.* (2011) studied the multivariate pattern analysis of brain images (fMRI) of how to categorize representation of ventral temporal cortex in the brain. More references about the other aspects on this topic might be found in Haxby *et al.* (2011) who proposed a so-called method, hyperalignment, to study the behavior of fMRI during some stimuli from a statistical perspective of view. The first chapter of this thesis proposes some predictions on the brain behaviors under stimuli based on the data rising from fMRI in the high dimensional perspective. Consequently, some classification techniques and approaches are developed to make the classifier model more powerful to distinguish different groups within a possible minor error rate. In this way, Procrustean problem are used to reach our goal. Procrustes, basically, is a least-square problem to transform a given matrix A to a given matrix B by T such that trace(E^TE) is minimized and $T^TT = I$, where $E = B - AT$ (Schönemann, 1966). This concept is discussed in greater detail in section 1.3. Procrustean problem is used in generalized Procrustean (GP) analysis (Devrim, 2003). GP finds a transformation matrix for each matrix data point to align them to the true and unknown common coordinates such that all the errors together are minimized, and this is shown in Section 1.3.2. An extension of the GP approach is also proposed which enhances the classification accuracy. Then, by doing statistical classification using the logistic regression by Lasso and Elastic net (see section 1.4.1) one can find the involved part of the brain of new subject during a stimulus and re-obtain the brain image to see those parts in the brain. Our results show that the accuracy of our classification approach is higher than existing ones. Moreover, by using our method, the brain image (the fMRI) can be captured again, ignoring the uninvolved regions during

the stimuli. Whereas, this is the missing link in the fMRI classification methods previously presented. Specifically, the limitation of GP is that the solution is not unique. For example, given one solution (map), any possible reshafling of its transformation matrix columns (voxels), i.e. multiplying T by any orthogonal matrix Q is still a valid solution (see 1.3.3). Therefore, the spatial coherence is lost. On the other side, the sequential application of Procrustes rotation does not reach the global minimum imposed by GP. As a matter of fact, to the best of our knowledge, all the proposed methods rely on sequential application of Procrustes rotation (e.g. Haxby *et al.* (2011)), while GP has been never used. In this thesis, GP is applied and an additional constrain is imposed that makes the solution unique (sections 1.3.2 and 1.3.3). The constrain is defined to enhance the interpretability of the solution (map/image). As a further advantage, the application to real data shows that the proposed method also enhances the classification accuracy.

Electroencephalogram (EEG)

Manipulating functional data in machine learning studies is highlighted in many practical researches, increasingly as big data problems, such as brain-computer interface (BCI). The main goal of BCI is translating brain signals to commands in the machine. It can be used for people with physical inability and movement problems and has attracted increased interest for use in video games (Barachant *et al.*, 2012). EEG signals which show the brain activity are the main focused data in BCI. EEG often obtained on short-time segments called trials such that each of them can be presented as a matrix with number of electrodes in the row and the epoch (time period) duration in the column (Barachant *et al.*, 2013); see Figure 4. Electroencephalographic data, with number of electrodes 19–64, number of time points between 200 and 1000 and more than 500 trails for one subject are treated as a kind of big data in statistics (Congedo, 2013). In BCI, the brain signals need to be classified depending on what the subject imagines or wants to do. Thus, a big data classification problem is encountered, and the classification problem of functional data rising from EEG signals in BCI is considered. In brief, each observation to be classified is the brain activity (i.e. multiple electrodes) over a fixed period of time. Therefore, each observation is a matrix. In the rest of the thesis, we make use of such setting as leading example (see Figure 2.10) to present the classification problem and the method proposed in this work. More details about the nature of the data are given in sub-section 2.1.1.

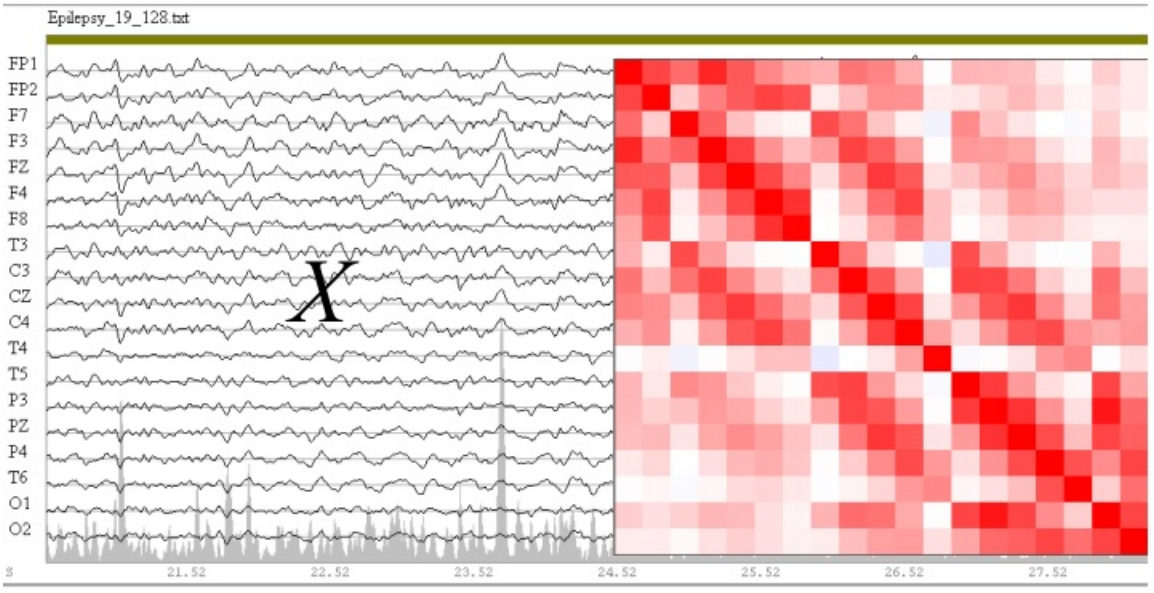


FIGURE 4: EEG signals as a trail in a case study (Congedo *et al.*, 2013).

While searching for appropriate models to describe and analyze functional data, the concept of covariance matrix sometimes is raised naturally or can be driven from the raw data depending on the problem in hand (Congedo *et al.*, 2013). Thus, the classification problem moves from observed covariance matrices to sample covariance matrices (i.e. symmetric positive definite - SPD - matrices). Statistical analysis of covariance matrices are arised in many applications as well as in BCI classification as a functional data problem. Estimating the average of available sample covariance matrices is a crucial step in such classification problems. Assuming m random vector samples $(V_{1i}, \ldots, V_{ni}), i = 1, \ldots, m$; from Wishart distribution, causes to the arithmetic mean as the estimation of population mean, which coincides with the MLE. This estimator can be presented using Euclidean distance in Frechet varational approach (see Section 2.3.3.1). However, working on the sample covariance matrices, the Euclidean space does not provide optimal properties. Skovgaard (1984) shows that when the data arise from a multivariate normal distribution, the Riemannian mean of SPD matrices provides some optimal properties. We consider the Riemannian manifold of SPD matrices. This manifold in coincidence with Riemannian geometry techniques are well adopted in BCI classification, and they provide a rich framework to manipulate in this context (Barachant *et al.*, 2012). In addition, estimating means of the data points lying on the Riemannian manifold of SPD matrices has proved of great utility in applications requiring interpolation, extrapolation, smoothing, signal detection and classification (Barachant *et al.*, 2012; Congedo *et al.*, 2017). Lim and Pálfia (2012) introduced the concept of power means for SPD matrices such as real positive numbers case. As an extention of the univariate case, power means

with exponent p in the interval $[-1, 1]$ interpolate in between the harmonic mean when $p = -1$ and the arithmetic mean when $p = 1$, while the geometric (Cartan or Karcher) mean arises when $p \to 0$ (Congedo *et al.*, 2017). to compute the power means, a general fixed point algorithm (MPM) is provided, and its convergence rate for $p = \pm 0.5$ deteriorates very little with the number and dimension of points given as input. Along the whole continuum, MPM is also robust with respect to the dispersion of the points on the manifold (noise) which is much more so than the gradient descent algorithm (Lim and Pálfia, 2012) usually employed to estimate the geometric mean. Thus, MPM is an efficient algorithm for the whole family of power means, including the geometric mean, which by MPM can be approximated with a desired precision by interpolating two solutions obtained with a small $\pm p$ value. Another motivation to use power means and their combinations is the convergence problems of available algorithms to estimate the geometric mean. The most popular algorithm for computing the geometric mean, which is the one currently employed in most applications, is a Riemannian gradient descent flow with fixed step size Afsari *et al.* (2013); Jeuris *et al.* (2012). The convergence rate of this algorithm deteriorates rapidly as the dispersion of points on the manifold decreases and it does not converge at all in some cases. The algorithm proposed in Zhang (2014) has high complexity per iteration and slow convergence rate and for a review of available algorithms for estimating the geometric mean see Congedo *et al.* (2015); Jeuris *et al.* (2012). A multiplicative algoithm is proposed for for estimating power means (MPM), which it can be used to estimate the geometric mean, as well. For a complete discussion on the benefits of MPM compared to other available algorithms in different cases, see Congedo *et al.* (2017). Geometric mean is the most used one in practice for such classifications so far, while, we will see that it might not be the best estimator of the mean population depending on the data distribution. Indeed, this optimality strongly relies on the distributional assumption that cannot be verified in practise (specially in the multivariate framework). To provide an intuition of this fact, a simple example is drawn from the univariate case. The arithmetic mean is the best estimator of the mean population when the data follow a normal distribution. However if, for example, the data were from a lognormal distribution i.e. $X_1, \ldots, X_n \sim LG(\mu, \sigma^2)$, the MLE of parameter e^{μ} is the geometric mean of the observation $(\prod_{i=1}^{n} X_i)^{1/n}$ and not the arithmetic mean. This has direct consequences on current classification problems: the centers of the classes should be the geometric mean and the distance to be used is euclidean with log transformed data. When this example is extended to the multivariate setting, the lack of adequate tools to judge the fit of the data to a given multivariate distribution makes the problem even more difficult to be dealt. The analysis of the real

data of our motivating example (Figure 5) shows that different subjects present different best (in terms of accurcy in the classification) power means and that a pattern to select the optimal power mean among subjects cannot be drawn. After providing an adequate theoretical background, sub-section 2.4.2.2 (and Figure 5) presents with better details of these considerations.

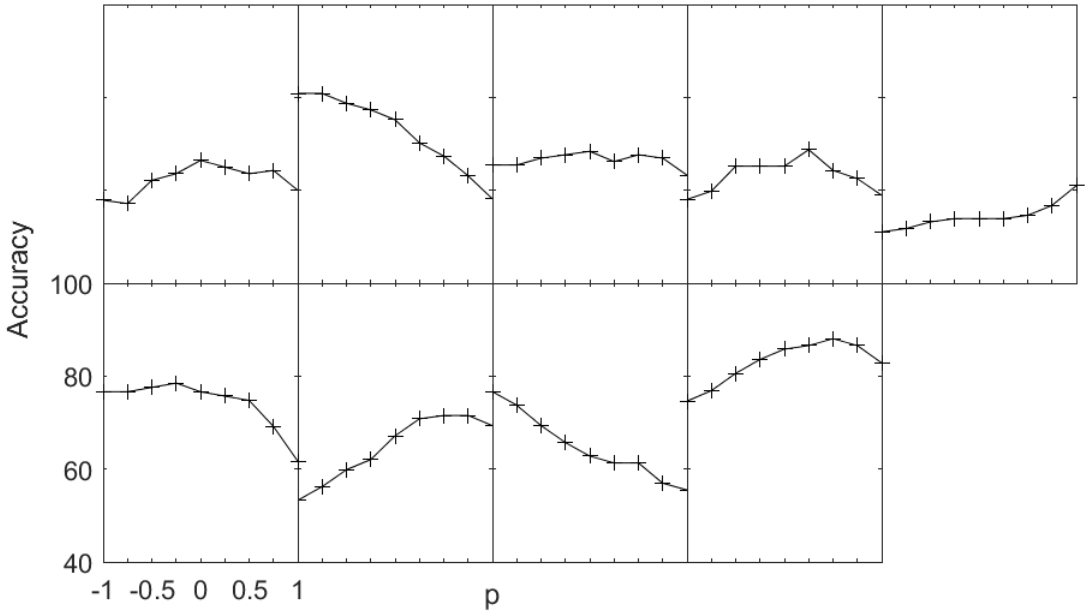


FIGURE 5: Classification accuracy on 9 subjects for the classes 3 vs 4 in Motorimagery task using MDM algorithm for the training and test sets in size of 288 trials. Power means with $p \in \{\pm 1, \pm 0.75, \pm 0.50, \pm 0.25, 0\}$ are estimated by MPM algorithm.

Main contributions of the thesis

For fMRI, by using the proposed approach to classify and predict brain activity for a new case in the presence of a stimuli, more powerful performance is obtained compared to other approaches in the sense that a lower misclassification errors obtain using the proposed approach. Furthermore, some proposed approaches (i.e. generalized Procrustes and its modification) have not been used so far in such fMRI classification study, however some modifications that enhance the accuracy of classification are supplied. In addition, with our method, after classifying the brain fMRI of a new subject to the true class, one will be able to re-capture the brain images to see involved parts. It is noteworthy, by previous approaches that there might be several brain images that minimise the error of the orthogonal transformation in Procrustes problem. This is very important from neuroscience point of view and it can be used in any problems that predict brains

are important. Many programming codes for reading from fMRI scans and codes for using classification approaches are provided to manipulate the fMRI data in practice. The codes are developed in R and are available on request to the author and can be gathered later as a package in R.

In addition, a classification study is proposed on EEG signals while the available data are matrices. In addition, a multiplicative algorithm (MPM) which is a fast and efficient algorithm was developed to compute the power means (for a set of values of parameter p in section 2.3.3) for matrices which is the crucial step in our proposed approaches for classification. A motivation to use power means is the convergence problems of available algorithms for estimating the geometric mean. In some simulation studies , the performance of MPM is examined and compared against existing algorithms. In addition, the behavior of different power means are compared in terms of accuracy in our classifications, which has not been discovered so far in such studies. We will show that it is difficult to have a guess to find the optimal power mean that provides the highest accuracy depending on the multivariate distribution of data in hand. Then, an approach that is a combination of power means was developed to have the benefit of all to improve the classification performance. As a result, the combination method is shown to be a very general approach and to be more powerful for different sample sizes of the training set and an accuracy almost close to the accuracy of optimal power mean can be obtained while a pattern to select the optimal power mean among subjects cannot be drawn in advance. All the codes related to the fast MPM algorithms and the codes for manipulating EEG signals in classification are written in MATLAB and can be later developed as a package.

Chapter 1 explains the classification and analysis related to fMRI and chapter 2 discusses the analysis and classification of EEG signals in a BCI problem. The chapter contains two main sections. Section 2.3 explains some basic details is needed to know to work with EEG data and some algorithms that they will be used in our classification approaches, and section 2.4 is for developing our classification approaches on EEG signals.

Chapter 1

Functional Magnetic Resonance Image Analysis

1.1 Introduction to fMRI

Functional magnetic resonance imaging (fMRI) showing the brain activity , and it focuses on the detected activities related to the blood flow. This measurement is based on the fact that cerebral blood flow and neuronal activation are associated. As a matter of fact when a region in the brain is working, blood flow to that area also increases. (Huettel *et al.*, 2004; Logothetis *et al.*, 2001). This measurement can be done, as it is so far, based on the blood oxygen level dependence which called in brief BOLD. Due to the energy used by brain cells, by imaging the change in blood flow (hemodynamic response), this is a type of specialized brain and body scan used to map neural activity (but not directly) in the brain or spinal cord of humans or other animals. BOLD measure is quite often took down by noise from various sources; therefore, statistical tools are needed to extract the underlying signal. In practice, the brain activation can be graphically captured by color-coding the strength of activation across the brain or the specific region studied (Figure 1.1). The technique can localize activity to within millimeters and using standard techniques, no better than within a window of a few seconds. fMRI is used more in the research world; however, it is used to a lesser extent, in the clinical world. Newer methods which improve both spatial and time resolution are being researched, and these largely use biomarkers other than the BOLD signal (Langleben and Moriarty, 2013), because the brain does not store glucose, its primary source of energy. When neurons become active, getting them back to their original state of polarisation requires actively pumping ions across the neuronal cell membranes in both directions. The energy for those ion pumps is mainly produced from glucose.

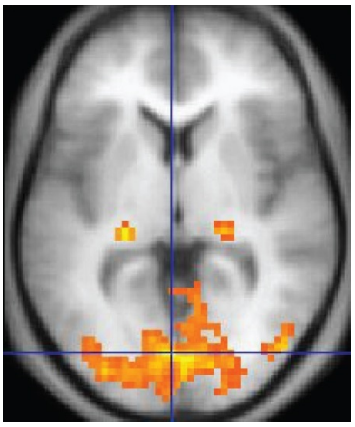


FIGURE 1.1: An fMRI image with yellow areas showing increased activity compared with a control condition. (http://cnx.org/contents/FPtK1zmh@8.25:fEI3C8Ot@10/Preface)

More blood flows in to transport more glucose, also bringing in more oxygen in the form of oxygenated hemoglobin molecules in red blood cells. This is from both a higher rate of blood flow and an expansion of blood vessels. Oxygen is carried by the hemoglobin molecule in red blood cells. The blood-flow change is localized to within 2 or 3 mm of where the neural activity is, and the brought-in oxygen is usually more than the oxygen consumed in burning glucose (it is not yet settled whether most glucose consumption is oxidative), and this causes a net decrease in deoxygenated hemoglobin (dHb) in the blood vessels of that area of the brain (Huettel *et al.*, 2004). When neurons become active, local blood flow to those brain regions increases, and oxygen-rich (oxygenated) blood displaces oxygen-depleted (deoxygenated) blood around 2 seconds later. This rises to a peak over 46 seconds before falling back to the original level (and typically undershooting slightly).

1.2 Data Structure

fMRI are obtained cross the time during a specific task (stimuli). Furthermore, each voxel has its own activity during the stimulus which is measured by BOLD. Consider the generic data model $X \in \mathcal{R}^{t \times p}$ where t indicates the time points and p is the number of voxels. Due to the huge number of voxels, each matrix X uses a large part of RAM (or hard disk) while manipulating the data in the computer such as doing any kind of algebra operation on the matrices. Based on our experience, this is sometimes around 15 GB which is too much only for doing operation on a single matrix. Instead, the number of columns is reduced by averaging the more correlated voxels in a process which we call pixelizing. Pixelizing can be performed up to have a certain number of voxels which

can be used in practice. All the related codes for pixelizing and also for reading fMRI scans are written in R and are easy to use.

1.3 Procrustes Approach

The Procrustes problem is basically a mathematical problem to transform a given matrix A to a given matrix B by T such that $\text{tr}(E^T E)$ minimized and $T^T = I$, where $E = B - AT$. In practice, A is usually the currently observed matrix, and T is calculated. Schönemann (1966) proposed a general solution for this problem. Mathematically speaking, it is a least-square problem to find a transformation matrix T such that,

$$AT = B + E \tag{1.1}$$

with respect to

$$\text{tr}(E^T E) \quad \text{minimized} \tag{1.2}$$

and

$$T^T T = I. \tag{1.3}$$

Both A and B are in the same dimension and not necessarily square. Mathematically, (1.1) introduces the main focused model, (1.3) is the *side condition* and (1.2) is our *criterion* in this least square problem. To solve the model in (1.1), the Equation (1.2) can be written as (Schönemann, 1966),

$$H_1 = tr(E^T E) = tr(T^T A^T AT - 2T^T A^T B + B^T B). \tag{1.4}$$

Also, by reforming the side condition (1.3), like

$$H_2 = tr\left(L\left(T^T T - I\right)\right), \tag{1.5}$$

where L is the matrix of Lagrange coefficients, the usual Lagrange optimization problem in matrix form is established

$$H = H_1 + H_2. \tag{1.6}$$

Now, the task is partial derivating H with respect to the T (in a matrix form; see Dwyer and MacPhail (1948)) and then find the extremum values:

$$\frac{\partial H}{\partial T} = 2A^T AT - 2A^T B + T(L + L^T). \tag{1.7}$$

For more simplicity, in (1.7), set $P = A^T A$, $S = A^T B$ and $Q = 2(L^T + L)$, so, by setting (1.7) to zero, the following equation must be solved to obtain the extremum of H_1,

$$S = PT + TQ, \tag{1.8}$$
$$Q = T^T S - T^T PT. \tag{1.9}$$

It is quite clear that P and Q are symmetric matrices. Therefore, in (1.9), $T^T PT$ and $T^T S$ are held symmetric; i.e.

$$T^T S = S^T T \tag{1.10}$$

The fact that T is orthonormal (1.3) and using (1.10) leads to obtain $S = TS^T T$ and consequently

$$SS^T = TS^T ST^T. \tag{1.11}$$

As Schönemann (1966) mentioned, from now we work on the known symmetric matrices SS^T and $S^T S$ which have the same spectral decomposition (latent roots); (Schönemann *et al.*, 1965)). Therefore, consider the following spectral decomposition

$$S^T S = VDV^T, \tag{1.12}$$
$$SS^T = WDW^T, \tag{1.13}$$

where W and V are the corresponding matrix of eigenvectors and D is a diagonal matrix of eigenvalues. It is well known that $W^T W = WW^T = V^T V = VV^T = I$. Starting from (1.11), one obtains

$$WDW^T = TVDV^T T^T, \tag{1.14}$$

which leads to $W = TV$ and then, consequently,

$$T = WV^T, \tag{1.15}$$

showing that the transformation T minimizes (1.2).

1.3.0.1 Uniqueness of T

The Eckart-Young matrix decomposition (Eckart and Young, 1936) can help to show the uniqueness of T. Some more discussions on this decomposition in approaches and usages can be found in Johnson (1963) and Schönemann *et al.* (1965). In particular,

considering (1.4) and (1.15), we have

$$H_1 = tr(E^T E) = tr(T^T P T - 2T^T S + B^T B) \quad (1.16)$$
$$= tr(P + B^T B) - 2tr(T^T S), \quad (1.17)$$

the latter equation is due to the orthogonality of T. The scalar ν can be defined as the following,

$$\nu = tr(T^T S) = tr(V W^T S) \quad (1.18)$$
$$= tr(V W^T W D^{1/2} V^T) \quad (1.19)$$
$$= tr(W W^T D^{1/2} V^T V) \quad (1.20)$$
$$= tr(D^{1/2}), \quad (1.21)$$

which the last equality is possible by cyclic permutation inside the trace function, since trace function is invariant under cyclic permutations of input matrices. In group theory, a cyclic permutation is a permutation of the elements of some set x which maps the elements of some subset s of x to each other in a cyclic fashion that starts permutation from an element and finishes the permutation to it , while fixing all other elements of x (that is, mapping to themselves). If s has k elements, the cycle is called a k-cycle. D is already introduced in (1.11). Equation (1.17) shows that minimizing H_1 is equivalent to maximizing ν. Therefore, ν is maximized if all diagonal elements in $D^{1/2}$ are non-negative. Once they are chosen, the orientation of W can be obtained by

$$S = W D^{1/2} V^T, \quad (1.22)$$

which is the Eckart-Young decomposition and it is used in the (1.18). This guarantees the uniqueness of T in the case of distinct eigenvalues (Schönemann, 1966).

There are two concerns in this approach. First, when multiple zero occurs in eigenvalues, since, orthogonal eigenvectors occur for distinct eigenvalues. To handle this, the projection matrix should be $T = [T_r \quad T_0]$, where T_r is the correspond matrix of eigenvectors of the nonzero eigenvalues and $T_0 = NG$. N is the null space of T_r i.e. $T_r^T N = 0$, while N is orthogonalised by G using the Gram-Schmidt approach. Thus,

$$T_r^T T_r = I, \quad T_0^T T_0 = I, \quad T_r^T T_0 = 0.$$

The second concern is more computational, and due to the available data, $D^{1/2} = W^T S V$ has some negative diagonal elements. To handle this, rotate W with an arbitrary

projection matrix until all the elements of $D^{1/2}$ are nonnegative. By calling the final rotation as W^*, then $T = W^*V'$ minimises Trace$(E'E)$. To get all the elements of $D^{1/2}$ nonnegative, for having a fast loop, a choice of the arbitrary projection matrix can be a diagonal one, with elements -1 to correspond to the negative diagonal elements of $D^{1/2}$ and 0 as the rest.

1.3.1 Hyperliagnment Approach

Haxby *et al.* (2011) introduced the so-called approach hyperalignment. The idea in hyperalignment is to use Procrustean transformation repetitively. In particular, first the voxel spaces for two matrices (subjects) were brought into optimal alignment. Then, a third subjects voxel space was brought into optimal alignment with the mean trajectory for the first two subjects and proceeded by successively bringing the voxel space of the remaining subjects into alignment with the mean trajectory of response vectors from previous subjects. In a second iteration , the voxel space of each individual subject was brought into alignment with the group mean trajectory from the first iteration and recalculated the group mean vector trajectory. The third and final step recalculated the orthogonal matrix that brought the voxel space of each subject into optimal alignment with the final group mean vector trajectory. The orthogonal matrix for each subject was then treated as the hyperalignment parameters of the subjects that were used to transform data from independent experiments into the common space (Haxby *et al.*, 2011).

1.3.2 Generalized Procrustes (GP)

Let $X_1, \dots X_n$ be $t \times p$ matrix observations in fMRI data for n subjects. The GP idea provides the least squares for more than two matrices. Indeed, one is looking for the transformation matrices T_i to satisfy (Devrim, 2003)

$$min \sum_{i=1}^{n} \sum_{j=i+1}^{n} tr(X_iT_i - X_jT_j)^T(X_iT_i - X_jT_j). \tag{1.23}$$

In the GP idea, there is an unknown matrix Z, also called the consensus matrix. Z represents the matrix that all the matrices in hand are going to align to its spaces in the common true coordinate system (Goodall, 1991; Devrim, 2003), as shown in Figure 1.2. Therefore, this problem can be seen as looking for the unknown matrix Z, as follows:

$$Z + E_i = X_iT_i = \hat{A}_i, \quad i = 1, \dots, n; \tag{1.24}$$

where E_i represents the error matrix with normal distribution, i.e.

$$E_i \sim N(0_{t\times p}, \Sigma_{tp\times tp}). \tag{1.25}$$

We mention this distributional assumption on E_i, to give a cue and head line for possible future works like considering some priors on parameters to challenge more with the likelihood. Thus, Equation (1.23) can be written as

$$min \sum_{i=1}^{n} \sum_{j=i+1}^{n} tr(\hat{A}_i - \hat{A}_j)^T(\hat{A}_i - \hat{A}_j) = min \sum_{i=1}^{n} \sum_{j=i+1}^{n} \left\|\hat{A}_i - \hat{A}_j\right\|^2, \tag{1.26}$$

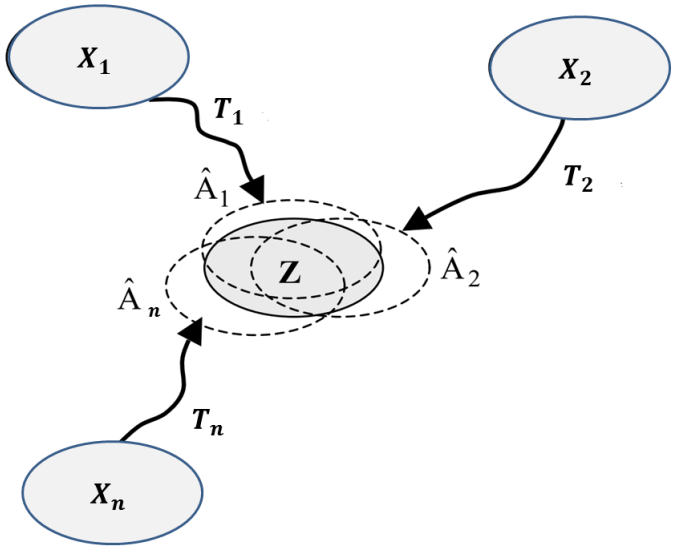


FIGURE 1.2: Schematic of GP (Crosilla and Beinat, 2002)

and then by defining the centroid as

$$C = \frac{1}{n} \sum_{i=1}^{n} \hat{A}_i, \tag{1.27}$$

Equation (1.26) is equivalent with

$$min \sum_{i=1}^{n} \left\|\hat{A}_i - C\right\|^2 = min \sum_{i=1}^{n} tr(\hat{A}_i - C)^T(\hat{A}_i - C), \tag{1.28}$$

(Kristof and Wingersky, 1971; Borg and Groenen, 2005). As a result, GP can be performed through minimizing the Equation (1.28) , which is easier than implementing (1.26). The iterative solutions for (1.26) can be found in Gower (1975) and Ten Berge (1977), and for (1.28) the algorithm is as follows:

Algorithm for GP
INPUT: $X_1, \ldots, X_n$ input matrices.
An initial value for C.
OUTPUT: T_i, and $\hat{Z}$ the estimation of consensus matrix.
REPEAT
Obtain the transformation parameter (T_i) for each of X_i with respect to C as a Procrustes problem.
Update C by (1.27).
Till
C stabilization at a certain precision.

As a matter of fact, with respect to the minimization condition, the final C determines the true coordinates in which all the X_is are aligned. Therefore, C can be seen as the least square estimate for Z (Crosilla and Beinat, 2002), i.e.,

$$\hat{Z} = C = \frac{1}{n} \sum_{i=1}^{n} \hat{A}_i. \tag{1.29}$$

Devrim (2003) proposed GP procedure in different cases of dependency of time points in each X_is and between, which, will show its affect on parameter Σ in (1.25).

1.3.3 Modification on GP

GP does not provide a unique solution as the transformation matrix (T). For example, given one solution (map), any possible reshafling of columns of its transformation matrix (voxels), i.e. multiplying T by any orthogonal matrix Q, is still a valid solution. Therefore, spatial coherence is lost. On the other side, the sequential application of Procrustes rotation does not reach the global minimum imposed by GP. As a matter of fact, to the best of our knowledge, all the proposed methods of Procrustes rotation rely on sequential application (e.g. Haxby *et al.* (2011)). In particular, for every matrix Q

with suitable dimension such that $Q^TQ = I$, (1.28) can still be minimized, i.e.

$$min \sum_{i=1}^{n} tr(X_iT_iQ - CQ)^T(X_iT_iQ - CQ), \tag{1.30}$$

$$= min \sum_{i=1}^{n} trQ^T(X_iT_i - C)^T(X_iT_i - C)Q, \tag{1.31}$$

$$= min \sum_{i=1}^{n} tr(X_iT_i - C)^T(X_iT_i - C)QQ^T, \tag{1.32}$$

$$= min \sum_{i=1}^{n} tr(X_iT_i - C)^T(X_iT_i - C), \tag{1.33}$$

because of orthogonality and cyclic permutation in the trace function. Therefore, the final transformation is not unique and one can rotate the matrices in hand several times while still maintaining minimal condition on the trace of error matrix. This is crucial for capturing back the brain image. Since once for a new subject the active voxels are obtained after classification in a specific task, one might be interested in having the brain image with active regions, consequently, having a unique rotation is necessary. Let $\bar{X}$ denote the mean of the observed data in hand i.e. $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$. It is reasonable to consider that $\bar{X}$ and C in (1.27) are expected to be close to each other. Thus, we find the orthogonal transformation Q such that

$$tr(CQ - \bar{X})^T(CQ - \bar{X}), \tag{1.34}$$

is minimized. Hence, the least square parameters as the transformation matrix Q are obtained to minimize the trace of error matrix with respect to C. In fact, another Procrustes problem this time between C and $\bar{X}$ can be solved, and the final transformation matrix which is applied on the data is

$$T_{i_M} = T_iQ. \tag{1.35}$$

According to (1.34), one more criterion is added to the problem, i.e. minimizing the trace of another error matrix, this time between $\bar{X}$ and C. In addition, this is also expected to decrease the misclassification error as shown in the real data study, see 1.5.

1.4 Classifier Model

Because of structure of the data in hand (see section 1.2) and the expected possible linearity among voxels (columns of our data matrices), the logistic regression was used as the classification approach. The matrix of the variables was constructed with voxels (p) in columns and the time points (t) in the row considering all subjects (n). Hence,

$$y_{ij} = \begin{cases} 0 & \text{having stimuli for the i-th subject in j-th time point} \\ 1 & \text{otherwise.} \end{cases} \tag{1.36}$$

Thus, $y = [y_{ij}]_{tn\times 1}$ and $X_{tn\times p}$ represent response and matrix of variables, respectively. Voxels in a fMRI study are roughly 500,000. However, after pixelizing (see section 1.2), the number of columns is reduced , but the number of variables (voxels) may still be larger than the number of observations. Therefore, this leads to model selection approaches such as lasso to be used in logistic regression. Also, regardless of the number of variables, since in our case one wants to find non zero coefficients (active voxels) during the stimuli, to re-obtain the brain image again with active regions, the model section becomes highlighted. Moreover, because of the existence of possible collinearity among variables, shrinkage methods such as ridge regression may be useful, see Hastie *et al.* (2008). The so-called elastic net method that is a convex combination of lasso and ridge was used in this research.

1.4.1 Lasso, Ridge and Elastic net

Lasso regularization was used to provide model selection. Lasso regularization causes some unimportant coefficients to be exactly zero. Lasso shrinks the regression coefficients by applying a penalty on their size; therefore, coefficients in this approach minimize a penalized residual. Generally speaking, shrinkage methods are more continuous and do not suffer as much from high variability (Hastie *et al.*, 2008).

$$\max_{\beta_0,\beta}\{\sum_{i=1}^{n}[y_i(\beta_0+\beta^T x_i)-\log(1+e^{\beta_0+\beta^T x_i})]+\lambda\sum_{j=1}^{p}|\beta_j|\} \tag{1.37}$$

The penalty part which is known as L_1 penalty can be written as $\sum_{j=1}^{p}|\beta_j| < t$, and makes the size constraint explicit on the parameters. There is a one-to-one correspondence between the parameters λ and t, and because of the nature of the penalty, for sufficient small value of t, some coefficients are set to be exactly zero. If $t > t_0 = \sum_{j=1}^{p}|\hat{\beta}_j|$, then Lasso estimates are the usual least square estimations, i.e., $|\hat{\beta}_j|$ are least square

estimates. On the other side, by $t = t_0/2$, the least square estimates are shrunk by an average of 50%. Usually, the parameter λ is chosen by cross validation to have a model which has the minimal mean cross validated error (in logistic, the deviance). Ridge regression is also useful when there are many correlated variables in the model and their coefficients can become poorly determined and exhibit high variance. In ridge, the penalty term is the L_2 penalty ($\lambda \sum_{j=1}^{p} \beta_j^2$). λ is a complexity parameter that controls the amount of shrinkage and large values of λ cause greater amounts of shrinkage. Using L_2 penalty, coefficients can be shrunk toward zero (and each other) but not exactly zero (Hastie *et al.*, 2008). To have both benefits of lasso and ridge, the elastic net penalty introduced by Zou and Hastie (2005) may be applied to balance the ridge and lasso penalties (L_1 and L_2 norms, respectively). The elastic net penalty is a convex combination of lasso and ridge penalties:

$$\sum_{j=1}^{p} \alpha|\beta_j| + (1-\alpha)\beta_j^2. \tag{1.38}$$

Elastic net selects variables like lasso, and shrinks together the coefficients of correlated predictors like ridge, as shown in Figure 1.3. The parameter α can be chosen by cross validation, again.

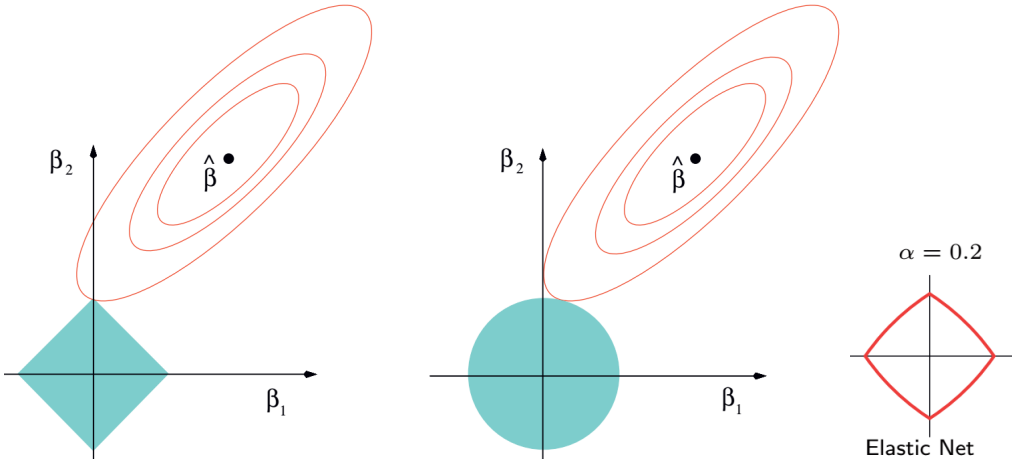


FIGURE 1.3: The constraint region for ridge regression (right side) is the disk $\beta_1^2+\beta_2^2 \leq t$, while the constraint region for lasso is the diamond (left side) $|\beta_1| + |\beta_2| \leq t$. Both methods find the first point where the elliptical contours hit the constraint region. Unlike the disk, the diamond has corners; if the solution occurs at a corner, then it has one parameter β_j equal to zero. Although it is not visually clear, the elastic net has sharp (non-differentiable) corners (Hastie *et al.*, 2008)

1.5 Real data study

The task is covert verb generation which is already considered in the work by Gorgolewski *et al.* (2013) and 10 subjects were asked to think of a verb complementing a noun visually presented to them. The following instructions were used: When a word appears it will be a noun. Think of what you can do with it and then imagine saying "With that I can ..." or "That I can ...". A block design with 30 s activation and 30 s rest blocks was employed (each scan takes 2.5 s $*$ 12 scan = 30 s). During the activation blocks, 10 nouns were presented for 1 s , and each were followed by a fixation cross during which subject had to generate the response. More details about the data can be found in Gorgolewski *et al.* (2013). Then, the brain fMRI are provided during the presence of stimulus and rest time. This is repeated for having 168 time points in general. As mentioned in Section 1.2, for each subject, the data from fMRI were gathered in a matrix with voxels in columns and time points in rows. The following tables are the classification error rates for four cases, data transformed by T_{i_M} in (1.35) in modification on generalized Procrustes (GPQ), by T_i in (1.15) in GP, by hyperalignment in 1.3.1 and not transformed data (Raw). Considering all the subjects, a test set of time points of size 40 is chosen and the rest is used as the training set. Then, once the nonzero coefficients of the voxels were obtained in the training set by elastic net penalty, the test set was used to find the error of misclassification. This procedure is repeated by 100 times re-sampling, and the average (standard deviation) of errors and ACU were calculated in Table 1.1. In addition, a box plot for all approaches is provided in Figure 1.4 to see the difference of methods visually.

	Methods			
	Raw	Hyper.	GP	GPQ
Total error	15.11(1.38)	15.088(1.485)	7.728(1.239)	6.667(1.223)
AUC	0.917(0.011)	0.925(0.012)	0.978(0.005)	0.982(0.01)

TABLE 1.1: Average (standard deviation) of total error and AUC on test set of size 40 repeated in 100 times.

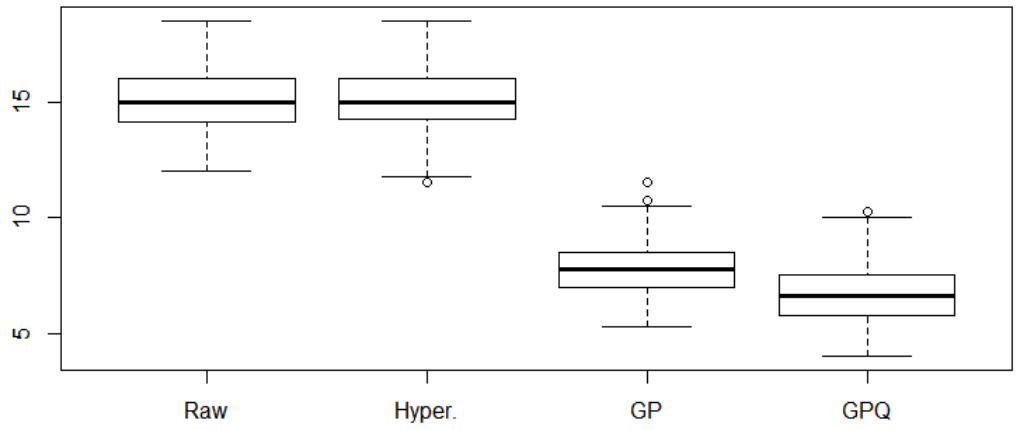


FIGURE 1.4: Box plot for total errors in 100 times re-sampling: raw data, hyperalignment, GP and GPQ methods.

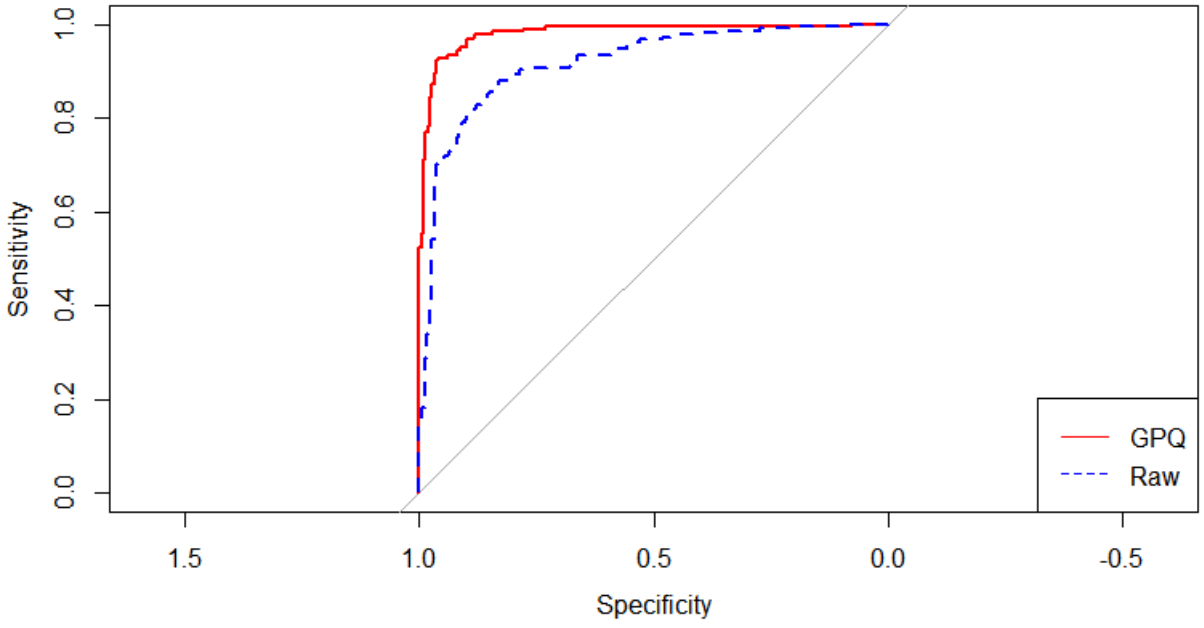


FIGURE 1.5: AUC plots for GPQ and Raw approaches.

Table 1.1 shows that, first of all, applying Procrustes problem is useful in classification. Furthermore, generalized Procrustes and our modification also will result to make the classifier model more powerful to distinguish different groups with smaller error rates and higher values of AUC, which GP has not been used in such classification studies thus far due to our knowledge; as shown in Figure 1.5.

1.6 Conclusion

To the best of our knowledge, all the proposed methods for rotating the data matrices rely on sequential application of Procrustes rotation and the sequential application of Procrustes rotation does not reach the global minimum imposed by GP, while GP has been never used in such classification studies of fMRI data. The limitation of GP is that the solution is not unique. Moreover, as mentioned, the spatial coherence is lost using GP on the data. In this thesis, GP was applied, and an additional constrain was imposed to make the solution unique. The criterion is defined to enhance the interpretability of the solution (map/image). As a further advantage, the application to real data shows that it also enhances the classification accuracy. Furthermore, the related codes for reading from MRI scans and writing the data as an image again all are provided in R. Also, the programming codes for Procrustes problem, modification on GP and hyperalignment are written in R. All together, the codes are available and will be provided as package as easily as possible for any usages.

Chapter 2

Electroencephalographic Signals

2.1 Introduction to EEG signals

2.1.1 EEG Signals

The main goal of brain computer interface (BCI) is translating the brain signals to the commands in the machine. BCI can be used for people with physical inability and movement problems or even with a focus on video games, which as shown growing interest (Barachant *et al.*, 2012). EEG signals which show the brain activity are the focus of the data obtained from BCI. EEG often obtained on short-time segments called trials such that, each of them can be presented as a matrix with number of electrodes in the row and the epoch duration in the column (Barachant *et al.*, 2013). Electroencephalographic data, with number of electrodes 19–64, number of time points like 200–1000 and more than 500 trails for one subject are treated as a kind of big data in statistics (Congedo, 2013). In BCI, the brain signals need to be classified depending on what the subject imagines or desires to achieve. Thus, this classification is basically a big data classification problem. Raw EEG are known to have a poor spatial resolution, since they are acquired with multiple electrodes covering the whole scalp which contains a considerable amount of spatial information. Usually in practice, spatial filtering is required to represent the data in a different space, possessing some desirable statistical property (Blankertz *et al.*, 2008; Congedo, 2013). Instead of using EEG signals, the corresponding covariance matrices of the data are considered, in which the diagonal elements are the variance of electrodes and the off-diagonal elements are their covariances. By this approach, spatial information are contained in the covariance matrices, and no more spatial filtering is needed (Barachant *et al.*, 2012). For event-related potentials data (ERP-based BCI) the spatial structure contained in that covariance matrix of a trial does not hold sufficient information for classification. In the other words, in this case,

that covariance matrix does not contain any temporal information at all. To overcome the problem, Congedo *et al.* (2013) proposed a modification of the usual covariance matrices for some kind of signals including ERP which is the main focused data in this work; see Section 2.2. In the rest of the thesis, covariance matrix means the latter extension and is used for all classifications (Section 2.2).

As a leading example (Figure 2.10), consider the experiment to produce motorimagery data by BCI Competition 2008 Graz data set A[1]. In this experiment, the subjects are asked to seat comfortably and look at the in-front monitor. At beginning of the trial ($t = 0s$), on the black color screen, a cross is appeared. After $t = 2s$, an arrow is shown while its direction to the left, right, down and up notifies the subject to imagine moving their left hand, right hand, foot and tongue, which constructs four classes for classification. The subject performs the motor imagery task till $t = 6s$ when the cross disappears from the screen. Then, 22 electrodes collect the EEG signals, and this trial is stored in the matrix, with 22 rows and many time points (more than 1000) in the columns. For a trial, which can belong to any of four classes, the covariance matrix is obtained as the working data in hand. This data set consists of EEG data from 9 subjects. The cue-based BCI paradigm consisted of four different motor imagery tasks, namely the imagination of movement of the left hand (class 1), right hand (class 2), both feet (class 3), and tongue (class 4). Two sessions on different days were recorded for each subject, and Each session comprised of 6 runs separated by short breaks. One run consists of 48 trials (12 for each of the four possible classes), yielding a total of 288 trials per subject. However, it may be possible some trials were removed as artifact records, and two classes (3 vs 4) were considered for classification.

2.2 Data model

The main thing in a BCI task is classifying single trials. In the first step, a generic model of the available data were specified and presented in such studies. Suppose $x(t) \in \mathcal{R}^N$ is the EEG data vector with N electrodes at a discrete sample time t with zero mean. Let $X_k \in \mathcal{R}^{N\times T}$ be a trial, as a finite time-interval realization and one of the T samples belonging to the class $k \in \{1, \ldots, K\}$. Each trial data is assumed to have zero mean since there is a usual band-pass filtering (Congedo, 2013). Thus, the well known sample covarince matrix of a sample trial belonging to the class k is given by

$$C_k = 1/(T-1)(X_k X_k^T) \tag{2.1}$$

[1]http://www.bbci.de/competition/iv/

2.2.1 Covariance Matrix for Motor Imagery (MI)

In the sample covariance matrix in (2.1), the diagonal elements present the variance of the signal at each electrode while the off-diagonal elements present the covarinace among all pairs of electrodes. (2.1) contains only spacial information which is sufficient for classification of MI data since MI trials for different classes generate different spatial patterns which is completely infixed in sample covariance matrix (2.1) according to its structure (Pfurtscheller and Da Silva, 1999; Congedo *et al.*, 2013). Thus, there is no extension for MI data to obtain the related covariance matrices as the working data; therefore,

$$C_k^{MI} = C_k. \tag{2.2}$$

In MI-based BCI, the only pre-processing step is filtering the data band-pass (e.g., 8–30 Hz); as shown in Congedo *et al.* (2013).

2.2.2 Covariance Matrix for Event-Related Potentials (ERPs) case

In ERP-based BCI (which is related to P300 data in our research; , as described in Section 2.4.3), the usual sample covariance matrix is not efficient, since the spacial structure of covariance matrix of a single trial does not contain sufficient information for classification. Indeed, in case, (2.1) does not contain temporal information (Congedo *et al.*, 2013). The reason is clear, because with a random jumble in samples of a trial X_k, the sample covariance matrix in (2.1) is not changed, nonetheless, ERPs have a specific time signature and it makes distinguished an ERP from another or an ERP from the absence of the ERP. Therefore, this is the required information (extracting and embedding) in a covariance matrix. To overcome this problem, consider a bunch of training trials $X_k; k \in \{1, \ldots, K\}$, while each class corresponds to a different ERP, also, a no-ERP class is usually added. For example, in P300-based BCI, one class is the target class, containing a P300, and the other is the non-target class which provides two classes ($K = 2$); see section 2.4.3 and the beginning introduction part. Now, a so-called super trial can be made (Congedo *et al.*, 2013):

$$X_k^{ERP} = \begin{bmatrix} \bar{X}_{(1)} \\ \bar{X}_{(2)} \\ \vdots \\ \bar{X}_{(K)} \\ X_k \end{bmatrix} \in \mathcal{R}^{N(K+1)\times T}, \tag{2.3}$$

where $\bar{X}_{(1)}, \ldots, \bar{X}_{(K)}$ are so-called temporal prototypes which are the average of the training trials on the previous session of the user or a data base of other users for each classes. These prototypes are computed for all classes, and the index (k) in the parentheses emphasizes the difference with index of X_k, which shows the k-th training class. The covarince matrix for the super-trail X_k^{ERP}, which is a block matrix, can be obtained as

$$C_k^{ERP} = 1/(T-1)\left(X_k^{ERP}\left(X_k^{ERP}\right)^T\right) \tag{2.4}$$

$$= 1/(T-1)\begin{bmatrix} \bar{X}.\bar{X}.^T & \left(X_k\bar{X}.^T\right)^T \\ X_k\bar{X}.^T & X_kX_k^T \end{bmatrix} \in \mathcal{R}^{N(K+1)\times N(K+1)}, \tag{2.5}$$

where

$$\bar{X}.\bar{X}.^T = \begin{bmatrix} \bar{X}_{(1)}\bar{X}_{(1)}^T & \ldots & \bar{X}_{(1)}\bar{X}_{(K)}^T \\ \vdots & \ddots & \vdots \\ \bar{X}_{(K)}\bar{X}_{(1)}^T & \ldots & \bar{X}_{(Z)}\bar{X}_{(K)}^T \end{bmatrix} \in \mathcal{R}^{NK\times NK}, \tag{2.6}$$

and

$$X_k\bar{X}.^T = \left[X_kX_{(1)}^T \ldots X_kX_{(K)}\right] \in \mathcal{R}^{N\times NK}. \tag{2.7}$$

More precisely, in (2.5), the $N \times N$ block $X_kX_k^T$ establishes the covarince matrix in (2.1) ,which contains only spacial information as discussed previously.

The $N \times N$ diagonal blocks of $\bar{X}.\bar{X}.^T$ in (2.6) represent the covariance matrices of K temporal prototypes, and its $N \times N$ off-diagonal elements represent the covariance between their pairs. Obviously, all these blocks are based on the fixed prototypes, and do not change from trials to trials, so they do not share useful information for classification. In (2.7), the $N \times N$ blocks hold the covariance between the trial X_k (corresponds to the class k) and K temporal prototypes; indeed, these blocks are temporal covariances, which was our concern in recent discussion. In addition, shuffling at random in samples of trials has some affects on the covarinace now. When the covariance of the trail and prototype with the same class in a block is large then there is relevant information regarding the covariance structure for classification. Also, a usual 1–16 Hz band-pass filtering is required as a pre-process step; however, the precise value of band-pass is not vital for ERP classification problems based on the lab experiments claimed by Congedo *et al.* (2013). We must say that what researchers are facing often, is the case of presence and absence of ERP, like, P300-based BCI. In that case, the following two classes are obtained: TARGET trials, which are when P300 is presented, and NON-TARGET trials, which are when P300 are not presented. Consequently, in P300-based BCI, the

super trial in (2.3) is simplified as the following

$$X_k^{P300} = \begin{bmatrix} \bar{X}_{(+)} \\ X_k \end{bmatrix} \in \mathcal{R}^{2N \times T}, \tag{2.8}$$

where $\bar{X}_{(+)}$ is the prototype of TARGET class (presence of P300) and the class index is $k \in \{+, -\}$. The $+$ and $-$ represent the TARGET and NON-TARGET classes, respectively. Thus, the covariance matrix of the super trial (2.8) changes to a simpler block matrix (Congedo *et al.*, 2013)

$$\begin{aligned} C_k^{P300} &= 1/(T-1)\left[X_k^{P300}\left(X_k^{P300}\right)^T\right] &(2.9)\\ &= 1/(T-1)\begin{bmatrix} \bar{X}_{(+)}\bar{X}_{(+)}^T & \bar{X}_{(+)}X_k^T \\ X_k\bar{X}_{(+)}^T & X_kX_k^T \end{bmatrix} \in \mathcal{R}^{2N\times 2N}. &(2.10)\end{aligned}$$

As shown in (2.6), $\bar{X}_{(+)}\bar{X}_{(+)}^T$ is based on the fixed prototypes and does not change from trial to trial; consequently, it is not useful for classification. Similar to (2.7), $X_k\bar{X}_{(+)}^T$ which is the temporal covariance, is sufficient for classifying TARGET and NON-TARGET classes. Notice that temporal covariance is large if the trial belongs to the TARGET class, while the temporal covariance is small if the trial does not belong to the TARGET class. Also, as discussed, X_kX_k has little information for classification. In the rest of the thesis by covarince matrix, we mean the latter extension in different type of the data in hand. Also, all the related programming codes in MATLAB to obtain such covariance matrices are written and are easily used in hand now. The extended covariance matrix for another type of data, steady-state svoked potentials, (SSEP), which is not our goal in this thesis, can be found in Congedo *et al.* (2013) .

2.3 Fixed point algorithms for estimating power means of positive definite matrices

Estimating means of data points lying on the Riemannian manifold of symmetric positive-definite (SPD) matrices has proved of great utility in applications requiring interpolation, extrapolation, smoothing, signal detection and classification. The power means of SPD matrices with exponent p in the interval $[-1, 1]$ interpolate in between the Harmonic mean ($p = -1$) and the Arithmetic mean ($p = 1$), while the Geometric (Cartan/Karcher) mean, which is the one currently employed in most applications, corresponds to their limit evaluated at 0. In this article we treat the problem of estimating

power means along the continuum $p \in (-1, 1)$ given noisy observed measurement. We provide a general fixed point algorithm (MPM; see 2.3.4) and we show that its convergence rate for $p = \pm 0.5$ deteriorates very little with the number and dimension of points given as input. Along the whole continuum, MPM is also robust with respect to the dispersion of the points on the manifold (noise), much more so than the gradient descent algorithm usually employed to estimate the geometric mean. Thus, MPM is an efficient algorithm for the whole family of power means, including the geometric mean, which by MPM can be approximated with a desired precision by interpolating two solutions obtained with a small $\pm p$ value. Finally, we show the appeal of power means through the classification of brain-computer interface event-related potentials data.

2.3.1 Introduction

The study of means (centers of mass) for a set of symmetric positive definite (SPD) matrices has recently attracted much attention, driven by practical problems in radar data processing, image and speech processing, computer vision, shape and movement analysis, medical imaging (especially diffusion magnetic resonance imaging and brain-computer interface), sensor networks, elasticity, numerical analysis and machine learning e.g., (Arsigny *et al.*, 2007; Arnaudon *et al.*, 2013; Barachant *et al.*, 2012, 2013; Congedo, 2013; Kalunga *et al.*, 2016; Faraki *et al.*, 2015; Fillard *et al.*, 2005; Fletcher, 2013; Li and Wong, 2013; Li *et al.*, 2012; Moakher, 2006; Zhang *et al.*, 2016). In many applications the observed data can be conveniently summarized by SPD matrices, for example, some form of their covariance matrix in the time, frequency or time-frequency domain, or autocorrelation matrices. In others, SPD matrices arise naturally as kernels, tensors (or slice of), density matrices, elements of a search space, etc. Averaging such SPD matrices is a ubiquitous task. In signal processing we find it in a wide variety of datadriven algorithms allowing spatial filters, blind source separation, beamformers and inverse solutions. While robust estimation of covariance matrices and related quantities is a long-standing topic of research, only recently an information/differential geometry perspective has been considered (Bhatia, 2009; Sra, 2016; Chebbi and Moakher, 2012; Moakher and Zéraï, 2011; Moakher, 2005; Bhatia and Holbrook, 2006; Nakamura, 2009; Georgiou, 2007; Jiang *et al.*, 2012). Once observations are represented as SPD matrices, they may be treated as points on a smooth Riemannian manifold in which the fundamental geometrical notion of distance between two points and the center of mass among a number of points are naturally defined (Bhatia, 2009). In turn, these notions allow useful operations such as interpolation, smoothing, filtering, approximation, averaging, signal detection and classification. In classification problems a simple Riemannian

classifier based on a minimum distance to mean (MDM) procedure (Barachant *et al.*, 2012) has been tested with success on electroencephalographic data, in several kinds of brain-computer interfaces (Barachant *et al.*, 2012, 2013; Congedo, 2013; Kalunga *et al.*, 2016) and in the analysis of sleep stages (Li and Wong, 2013; Li *et al.*, 2012), as well as on motion capture data for the classification of body movements (Zhang *et al.*, 2016). A similar method has been used for clustering in the context of video-based face and scene recognition (Faraki *et al.*, 2015) and in radar detection (Arnaudon *et al.*, 2013). These examples demonstrate that simple machine learning algorithms, which are known to allow poor performance using the Euclidean metric, can be easily translated into equivalent Riemannian classifiers using an appropriate metric, obtaining excellent performance, Among the several means one may define from an information geometry point of view, so far the geometric mean (sometimes referred to as Karcher, Cartan or Frchet mean) has been the most studied and the most used in practical applications. It is the natural definition of mean when the Fisher-Rao metric is applied to multivariate Gaussian distributions (Nakamura, 2009; Georgiou, 2007), but also arises naturally from a pure geometrical and algebraic perspective without making assumptions on the data distribution (Bhatia, 2009). It happens that the geometric mean satisfies a number of desirable invariances, including congruence invariance, self-duality, joint homogeneity and the determinant identity (Congedo *et al.*, 2015). The simultaneous verification of all these properties is hard to find for means based on other metrics, such as the arithmetic, harmonic and log-Euclidean mean, thus the geometric mean of SPD matrices is not just important in practice, but a fundamental mathematical object per se. For positive numbers the arithmetic, geometric and harmonic mean are all members of the family of power means, also known as Holder or generalized mean. Given a set of K positive numbers $\{x_1, \ldots, x_K\}$ and K associated weights $\{w_1, \ldots, w_K\}$ satisfying $\sum w_k = 1$, the w-weighted power mean of order p of $\{x_1, \ldots, x_K\}$ is

$$g = (\sum_{k=1}^{K} w_k x^p)^{1/p} \tag{2.11}$$

power mean interpolates continuously between Harmonic mean ($p = -1$) and Arithmetic mean ($p = 1$) in the continuum $p \in [-1, 1]$ while the limit $p \to 0$ allows the Geometric mean. This generality of power means is appealing from a signal processing perspective; in a typical engineering scenario the sensor measurement is affected by additive noise and varying p one can find an optimal mean depending on the signal-to-noise-ratio (SNR), as we will show.

Recently Lim and Pálfia (2012) extended the concept of power means of positive numbers

to SPD matrices for the continuum $p \in [-1, 1]$, with the case $p = -1$ being the matrix harmonic mean, $p = 1$ the matrix arithmetic mean and the limit to zero from both sides allowing the matrix geometric mean we have discussed (see also (Lawson and Lim, 2013, 2014; Pálfia, 2016). So far power means of SPD matrices have not been applied in signal processing. Also, only a "naive" fixed-point algorithm has been proposed for their estimation (Lim and Pálfia, 2012) and its convergence behavior is unsatisfactory. In this research we report a fixed-point algorithm for computing power means of SPD matrices along the interval $p \in (-1, 1)\backslash\{0\}$. This algorithm has been recently presented in (Congedo *et al.*, 2017) and therein we have named it MPM (multiplicative power means). We then demonstrate a procedure to use MPM for approximating the geometric mean with a desired precision. By means of simulation we show that the MPM displays better convergence properties as compared to alternatives used for the geometric mean, with equal or lesser computational complexity. We also show that it offers a better estimation of the geometric mean as compared to the standard gradient descent algorithm. Then, we show the advantage of considering the whole family of power means, instead of the sole geometric mean as it is customary, in classification problems, by analyzing a data set of 38 subjects related to brain-computer interface event-related potentials.

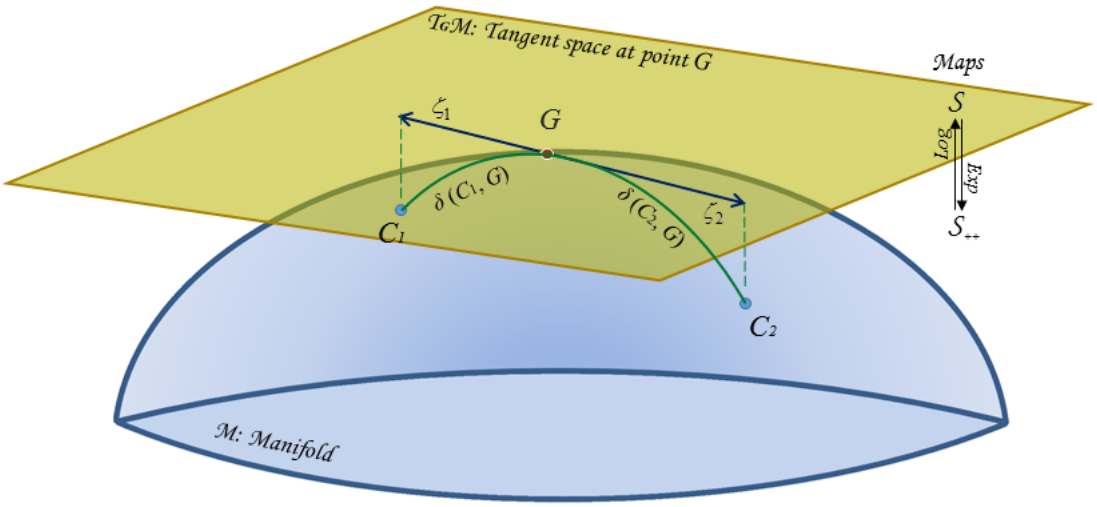
2.3.2 The Manifold of Symmetric Positive-Definite Matrices

In differential geometry, a smooth manifold is a topological space that is locally similar to the Euclidean space and has a globally defined differential structure. A smooth Riemannian manifold $\mathcal{M}$ is equipped with an inner product on the tangent space defined at each point and varies smoothly from point to point. The tangent space $\mathcal{T}_G\mathcal{M}$ at point G is the vector space containing the tangent vectors to all curves on $\mathcal{M}$ passing through G. For the manifold $\mathcal{M}$ of SPD matrices S_{++} , this is the space S of symmetric matrices. (Figure 2.1). For any two tangent vectors ζ_1 and ζ_2, the inner product given by the Fisher-Rao metric at any base-point G is desired (Bhatia, 2009):

$$\langle \zeta_1, \zeta_2 \rangle_G = \mathrm{tr}(G^{-1}\zeta_1 G^{-1}\zeta_2). \tag{2.12}$$

2.3.2.1 The Geodesic

The SPD manifold has non-positive curvature and is complete (Bhatia, 2009); for any two points C_1 and C_2 on $\mathcal{M}$, a unique path on $\mathcal{M}$ of minimal length (at constant velocity) connecting the two points always exists. The path is named the geodesic, and



H

FIGURE 2.1: Schematic representation of the SPD manifold, the geometric mean G of two points and the tangent space at G. Consider two points (e.g., two covariance matrices) C_1 and C_2 on $\mathcal{M}$. The geometric mean of these points is the midpoint on the geodesic connecting C_1 and C_2, i.e., it minimizes the sum of the two squared distances $\delta_1(C_1, G) + \delta_2(C2, G)$. Now construct the tangent space $\mathcal{T}_G\mathcal{M}$ at G. There exists one and only one tangent vector ζ_1 (respectively ζ_2) departing from G and arriving at the projection of C_1 (respectively C_2) from the manifold onto the tangent space; we see that the geodesics on $\mathcal{M}$ through G are transformed into straight lines in the tangent space and that therein distances are mapped logarithmically; the map from the manifold (symmetric positive definite matrices S_{++}) to the tangent space (symmetric matrices S) is of logarithmic nature. Furthermore, the inverse map from the tangent space to the manifold is of exponential nature. See Bhatia (2009) for details on these maps.

the points along it have analytical expressions given by

$$C_1 \#_t C_2 = C_1^{1/2}(C_1^{-1/2} C_2 C_1^{-1/2})^t C_1^{1/2}, \quad t \in [0, 1]. \tag{2.13}$$

By changing t we are moving over the geodesic connecting two points. For example, $t = 0$ corresponds to the C_1 location, $t = 1$ corresponds to the C_2 location, and $t = 1/2$ corresponds to the geometric mean of the two points (Figure 2.1). As a special case, $I \#_t C = C^t$ and geodesic equation (2.13) verifies $C_1 \#_t C_2 = C_2 \#_{1-t} C_1$ and $(C_1 \#_t C_2)^{-1} = C_1^{-1} \#_t C_2^{-1}$. The points along the geodesic can be understood as the t-weighted geometric means of C_1 and C_2 according to the Riemannian metric, in analogy with the weighted mean according to the Euclidean metric given by $(1 - t)C_1 + tC_2$, which still results in a SPD matrix, but, greater than $C_1 \#_t C_2$ in the Loewner order sense (Pálfia, 2016).

2.3.2.2 The Distance

For two matrices (points) C_1 and C_2 of dimension $N \times N$ on $\mathcal{M}$, the Riemannian distance is defined as the length of the geodesic in (2.13) and is given by (Bhatia, 2009),

$$\delta(C_1, C_2) = \left\| \mathrm{Ln}(C_1^{-1/2} C_2 C_1^{-1/2}) \right\|_F = \sqrt{\mathrm{tr}(\mathrm{Ln}^2(\Lambda))} = \sqrt{\sum_{i=1}^{N} \mathrm{Ln}^2(\lambda_i)}, \tag{2.14}$$

where Λ is the diagonal matrix holding the N eigenvalues $\lambda_1, \ldots, \lambda_N$ of matrix $C_1^{-1/2} C_2 C_1^{-1/2}$ or of similar matrix $C_1^{-1} C_2$. Some key features of Riemannian distance are listed in Congedo *et al.* (2015). Both symmetry and positivity are obvious properties and the next proposition mentions the two invariance properties that are useful in signal processing. For any invertible matrix with suitable dimension B,

$$\text{Congruence} \qquad \delta(BC_1B^T, BC_2B^T) = \delta(C_1, C_2), \tag{2.15}$$

$$\text{Self-Duality} \qquad \delta(C_1^{-1}, C_2^{-1}) = \delta(C_1, C_2). \tag{2.16}$$

2.3.3 Means of Matrices

The study of means (centers of mass) for a set of SPD matrices has recently attracted much attention, driven by practical problems in radar data processing, image and speech processing, computer vision, shape and movement analysis, medical imaging (especially diffusion magnetic resonance imaging and brain-computer interface), sensor networks, elasticity, numerical analysis and machine learning. In many applications, the observed data can be conveniently summarized by SPD matrices, for example, some form of their covariance matrix in the time, frequency or time-frequency domains. In others, SPD matrices arise naturally as kernels, tensors (or slice of) density matrices, elements of a search space, etc. Averaging such SPD matrices is a ubiquitous task, and the averaging can be obtained in a wide variety of data driven by signal processing algorithms such as spatial filters, blind source separation, beamformers and inverse solutions.

2.3.3.1 Frechet's variational approach

Let $C = \{C_1, \ldots, C_K\}$ be a set of SPD matrices and $w = \{w_1, \ldots, w_K\}$ be a set of K associated positive weights verifying $\sum_k w_k = 1$. Typically, in signal processing, the elements of C are noisy data points (e.g. recordings, observations, etc.) or quantities derived thereof. Following the Frechet's variational approach, the center of mass G of set C, given a distance function d, is the point G minimizing the dispersion of points, that

is, $\sum_k w_k d^2(G, Ck)$. This definition applies in general. For instance, the w-weighted arithmetic and harmonic means are defined, respectively, as

$$G_{\mathcal{A}}(C; w) = \underset{G}{\operatorname{argmin}} \sum\nolimits_k w_k \left\| C_k - G \right\|_F^2 = \sum\nolimits_k w_k C_k, \tag{2.17}$$

$$G_{\mathcal{H}}(C; w) = \underset{G}{\operatorname{argmin}} \sum\nolimits_k w_k \left\| C_k^{-1} - G^{-1} \right\|_F^2 = \left(\sum\nolimits_k w_k C_k^{-1} \right)^{-1}, \tag{2.18}$$

in which, $\| \cdot \|_F$ is the Frobenius norm.

2.3.3.2 The Geometric Mean of a Matrix Set

Following the same idea, the geometric mean of SPD matrices can be defined (Bhatia, 2009). On the manifold $\mathcal{M}$, the w-weighted geometric mean $G_{\mathcal{G}}(C; w)$ is the point realizing the minimum of $\sum_k w_k \delta^2(C_k, G)$ with respect to G, where the Riemannian distance function δ acting on $\mathcal{M}$ has been defined in definition (2.14). Indeed, the geometric mean G is the unique point on $\mathcal{M}$ such that the following non-linear matrix equation is satisfied (Moakher, 2005):

$$\sum\nolimits_k w_k \mathrm{Ln}(G^{-1/2} C_k G^{-1/2}) = 0. \tag{2.19}$$

In general, for $K > 2$ equation (2.19) dose not have closed form solution and needs to be estimated by iterative algorithms. For $K = 2$, as mentioned in Section 2.3.2.1, the geometric mean is equal to $C_1 \#_{1/2} C_2$ (shortly indicated by $C_1 \# C_2$; see (2.13) and Figure 2.1). Furthermore, it is the unique solution of the Riccati equation $(C_1 \# C_2) C_2^{-1} (C_1 \# C_2) = C_1$ (Arnaudon *et al.*, 2013) and is equal to $B^{-1} D_1^{1/2} D_2^{1/2} B^{-T}$ for any joint diagonalizer B of C_1 and C_2, that is, any B satisfying $B C_1 B^T = D_1$ and $B C_2 B^T = D_2$, with D_1 and D_2 being invertible diagonal matrices Congedo *et al.* (2015). The geometric mean satisfies all 10 properties of means postulated in the seminal work (Ando *et al.*, 2004). Also, straightforward from (2.15) for any invertible matrix B with suitable dimension,

$$\text{Congruence} \quad G_{\mathcal{G}}\left(B C_1 B^T, \ldots, B C_K B^T; w\right) = B G_{\mathcal{G}}\left(C; w\right) B^T, \tag{2.20}$$

$$\text{Self-Duality} \quad G_{\mathcal{G}}^{-1}(C_1^{-1}, \ldots, C_K^{-1}; w) = G_{\mathcal{G}}(C; w). \tag{2.21}$$

2.3.3.3 Power Mean

Given a set of K positive numbers $\{x_1, \ldots, x_K\}$ and K associated weights $\{w_1, \ldots, w_K\}$ satisfying $\sum_k w_k = 1$ following the Frechet's variational approach, it is well known that the power mean in real number case can be defined as $M_p = \underset{x}{\operatorname{argmin}} \sum_k w_k |x_k^p - x^p|^2$. This fact leads M_p being as a unique positive solution of the equation $x = \sum_k w_k x^{1-p} x_k$.

The matrix analogue form can be obtained as (Lim and Pálfia, 2012)

$$X = \sum_k w_k (X \#_p C_k), \tag{2.22}$$

where $C = \{C_1, \ldots, C_k\}$ and w_k are arbitrary weights. This matrix equation has unique SPD solution $G_{\mathcal{P}}(C; w; p)$ (called matrix power mean) for $p \in (0, 1]$ (Lim and Pálfia, 2012). By defining $G_{\mathcal{P}}(C; w; p) = G_{\mathcal{P}}^{-1}(C^{-1}; w; -p)$ for $p \in [-1, 0)$, power mean interpolates continuously between harmonic mean ($p = -1$) and arithmetic mean ($p = 1$) in the continuum $p \in [-1, 1]$ while the limit $p \to 0$ produces the geometric mean. It has been shown that if all C_k input matrices commute, then,

$$G_{\mathcal{P}}(C; w; p) = (\sum_k w_k C_k^p)^{1/p}, \tag{2.23}$$

(Lim and Pálfia, 2012), which is the straightforward extension of real numbers case. For any pair (G, C_k) in $\mathcal{M}$, $G \#_p C_k$ with $p \in [0, 1]$ is the mean of G and C_k weighted by p. Since $G \#_p C_k = C_k \#_{1-p} G$ we see that a power mean is the arithmetic mean of the input matrices dragged along the geodesic toward the desired mean by an arc-length equal to $1 - p$. Briefly, the power means over the continuum $[-1, 1]$ can be presented as the following,

$$\begin{cases} G_{\mathcal{P}}(C; w; p = 1) & = & G_{\mathcal{A}}(C; w), \\ G_{\mathcal{P}}(C; w; p \in (0, 1)) & = & \sum_k w_k (G_{\mathcal{P}} \#_p C_k), \\ G_{\mathcal{P}}(C; w; p = 0) & = & G_{\mathcal{G}}(C; w), \\ G_{\mathcal{P}}(C; w; p \in (-1, 0)) & = & G_{\mathcal{P}}^{-1}(C^{-1}; w; -p), \\ G_{\mathcal{P}}(C; w; p = -1) & = & G_{\mathcal{H}}(C; w), \end{cases} \tag{2.24}$$

$C^{-1} = \{C_1^{-1}, \ldots, C_K^{-1}\}$, $G_{\mathcal{G}}(C; w)$ is the geometric mean of Section 2.3.3.2 and $G_A(C; w)$ and $G_H(C; w)$ are the arithmetic mean and the harmonic mean in (2.17) and (2.18), respectively. $G_{\mathcal{P}}(C; w; p)$ is named the w-weighted power mean of order p (Lim and Pálfia, 2012; Pálfia, 2016). As per (2.24), the pair of power means obtained at opposite values of p around zero are duplicates of each other; for a negative value of p, the mean is defined as the inverse of the mean for p as applied on the inverted input matrices C^{-1}. Thus, the power means family encompasses and generalizes all Pythagorean means encountered thus far. All of them enjoy the congruence invariance as found in the geometric mean (2.20), but their duality, expressed in the fourth line of (2.24), coincides with the self-duality property (2.21) only for $p = 0$. The numerous properties of the power means can be found in Lim and Pálfia (2012) and a recent extension of this already quite general mathematical object has been proposed in Pálfia (2016).

2.3.4 Algorithm For Power Means

Suppose P_0 is used as an initial value to determine the power mean in the iterative equation (2.22). Once the value of p is fixed, it corresponds to a certain point on the geodesic connecting each C_k and P_0. Then, the arithmetic mean of these points on the geodesics is considered as the new starting value based on the (2.22), and this procedure continues till the power mean is established, as shown in Figure2.2.

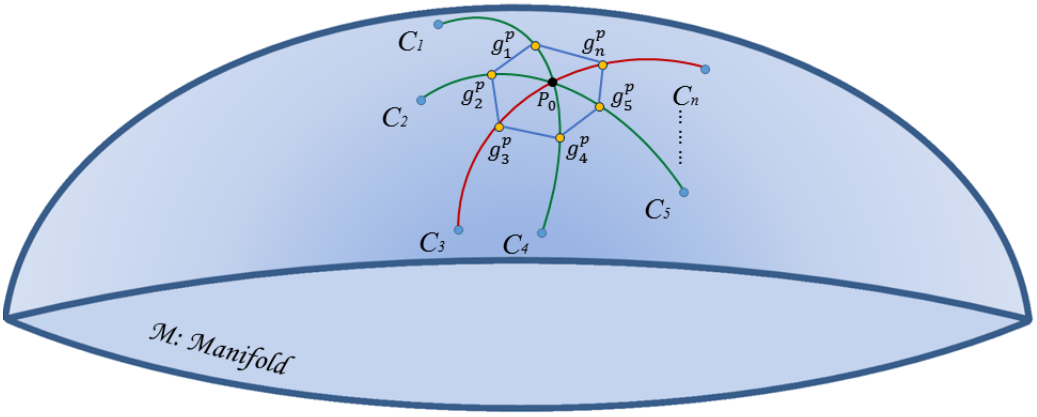


FIGURE 2.2: The schematic procedure of estimating power means in (2.22). Suppose P_0 as the initial value for this iterative equation. By fixing the order of power mean as p, we are at the point $g_k^p = P_0 \#_p C_k$ on the geodesic connecting C_k and P_0 for $k = 1, \ldots, K$. Then, the arithmetic mean of g_k^p's is computed and it is considered as the new starting point in (2.22). Again, the arithmetic mean of new g_k^p's in the second iteration is calculated and this procedure continues till the power mean is obtained up to a given precision.

We sought a general algorithm for computing the w-weighted power mean of order p, with $p \in (-1, 1) \backslash \{0\}$. We are also interested in an effective algorithm for estimating the geometric mean, the third line in (2.24). The most popular algorithm for computing the geometric mean is a Riemannian gradient descent flow with fixed step size (Afsari *et al.*, 2013; Jeuris *et al.*, 2012) and the convergence rate of this algorithm deteriorates rapidly as the SNR decreases (high dispersion of points on the manifold). The same is true for the method based on approximate joint diagonalization in (Congedo *et al.*, 2015). Second order methods have complexity grown very fast with the size of the input matrices; thus, they are little useful in practical applications (Jeuris *et al.*, 2012). The algorithm proposed in Zhang (2014) has high complexity per iteration and slow convergence rates. For a review of available algorithms for estimating the geometric mean, see Congedo *et al.* (2017). Our algorithm does not need to make use of Riemannian geometry optimization in the manifold of SPD matrices, with consequent conceptual

and computational advantage. For instance, we will be able to derive a fast approximation based exclusively on triangular matrix algebra and on the Cholesky decomposition (details are in the paper by (Congedo *et al.*, 2017)).

2.3.4.1 A General Multiplicative Fixed-Point Algorithm

Hereafter it is convenient to lighten notation; let the weighted power mean of order p be denoted as P, which by (2.24) is equal to $G_{\mathcal{P}}(C; w; p)$ if $p \in (0, 1)$ or to $G_{\mathcal{P}}^{-1}(C^{-1}; w; -p)$ if $p \in (-1, 0)$. This method only needs to handle one expression for whatever value of $p \in (-1, 1) \backslash \{0\}$, such as

$$P^* = G_{\mathcal{P}}(C^*; w; |p|); \tag{2.25}$$

where $|p| = \text{abs}(p)$ and the dual operator * is defined as $* = \text{sgn}(p)$. Definition (2.25) is here introduced to define an algorithm with identical convergence behavior for all pairs of values $\pm p$ for $|p| \in (0, 1)$. Therefore, only the results for p positive are shown. As initialization, the closed form solution of the mean in the case when all matrices in set C all pair-wise commute is used, as given by (2.23). Let us now turn to the iterations. (2.25) can be written out from definition (2.22) and using (2.13) to obtain

$$P^* = P^{*/2} \left[\sum_k w_k \left(P^{-*/2} C_k^* P^{-*/2} \right)^{|p|} \right] P^{*/2}. \tag{2.26}$$

In Lim and Pálfia (2012), the authors showed that the map defined by $f(P^*) = G_{\mathcal{P}}(C^*; w; |p|)$; is a strict contraction for the Thompson metric (see Bhatia (2009)) with the least contraction coefficient less than or equal to $1 - |p|$, and as such, it has a unique SPD fixed point. Numerical experiments show that iterating expression (2.26) as it is (hereafter referred to as "naive fixed-point") results in a rather slow convergence rate. It becomes maximal for $|p| = 1/2$, but it becomes slower and slower as $|p|$ becomes closer to 0 or to 1. To hasten convergence we design a multiplicative algorithm as follows: post-multiplying both sides of (2.26) by $P^{-*/2}$ and taking the inverse at both sides, the following is obtained:

$$P^{-*/2} = H^{-1} P^{-*/2}, \tag{2.27}$$

where

$$H = \sum_k w_k \left(P^{-*/2} C_k^* P^{-*/2} \right)^{|p|}. \tag{2.28}$$

From (2.26), upon convergence, $H = I$. H here plays the role of the origin in the SPD manifold $\mathcal{M}$ for data linearly transformed by $P^{-*/2}$. In particular, the identity matrix

I is the point of symmetry in $\mathcal{M}$ corresponding to 0 in the Euclidean space due to the logarithmic map; as $P^{-1/2}$ is a whitening matrix for the arithmetic mean ($p = 1$), so $P^{-*/2}$ is a whitening matrix for the whole family of power means. We wish to proceed by multiplicative updates according to (2.29). Rather than converging to P^* itself, an algorithm converging to $P^{-*/2}$ is used, which is its inverse square root for $* = 1$, i.e., when $p \in (0, 1]$ and its square root for $* = -1$, i.e., when $p \in [-1, 0)$. The numerical stability of fixed-point iterates (2.29) is ensured by the fact that H converges toward I. Moreover, using our update rule, any update matrix with form $H^{-\phi}$ in (2.29) is equivalent to H^{-1} upon convergence. We have observed that replacing H^{-1} by $H^{-\phi}$ in the update rule (2.29) does not alter the convergence to the fixed point. Nonetheless, the value of exponent ϕ impacts the convergence rate. In practice, using an optimal value of ϕ leads to a significantly faster convergence as compared to the convergence achieved by setting $\phi = 1$. This holds true for power means in the whole interval $p \in (-1, 1)\backslash\{0\}$. Therefore, the following iterate equation is used,

$$P^{-*/2} = H^{-\phi} P^{-*/2}, \tag{2.29}$$

interestingly, optimal convergence speed is observed taking ϕ in an interval whose extremes vary proportionally to $|p|^{-1}$. An heuristic rule that has proven adequate in intensive experiments using both real and simulated data is

$$\phi = \frac{1}{2}\epsilon^{-1}/|p|, \quad \epsilon \in [1, 2], \tag{2.30}$$

where ϵ is a constant eccentricity parameter for hyperbolas (2.30) (Figure 2.3).

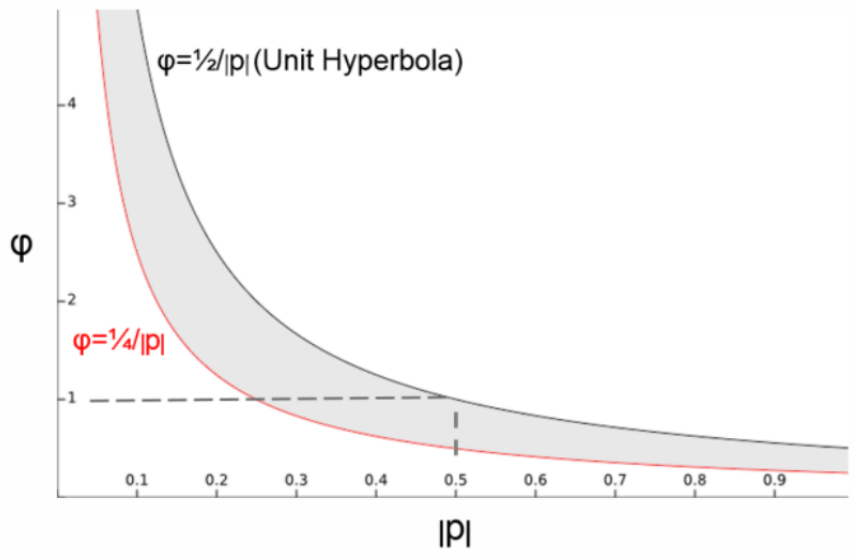


FIGURE 2.3: The ϕ function of $|p|$ (2.30) comprises a boomerang-shaped area enclosed by two hyperbolas: the upper limit is the unit hyperbola ($\epsilon = 1$) and the other hyperbola obtained for $\epsilon = 2$ is the lower limit. This area delimits an acceptable range of ϕ values for any given $|p|$.

The exponent $-\phi$ in (2.29) acts by retracting the jumps of the fixed point iterations. Since the fixed point is reached at $H = I$, and ϕ is always positive in (2.30), $H^{-\phi} = H\#_{-\phi}I = I\#_{1+\phi}H$ (see section 2.3.3.2) represents the movements over the geodesic from I to H (i.e., in the direction opposite to convergence), retracting H by a distance equal to ϕ times the distance between I and H (here ϕ is the arc-length parameter of equation (2.13)). The retraction is maximal for the unit hyperbola ($\epsilon = 1$) and minimal for $\epsilon = 2$. By increasing ϵ toward 2 we obtain faster convergence in general, up to a certain value, which according to our observations mainly depends on the signal-to-noise ratio. In this study we take ϵ as 4/3 and we keep it fixed in all analyses; this value has proven nearly optimal on the average of many combinations of SNR, input matrix sizes and dimensions we have tested. The MPM algorithm in algebraic pseudo-code is as follows:

Algorithm MPM (Multiplicative Power Means)

INPUT: $p \in [-1,1]\backslash\{0\}$, K positive weights $w = w_1, \ldots, w_K$ such that $\sum w_k = 1$ and $N \times N$ SPD matrices $C^* = \{C_1^*, \ldots, C_K^*\}$, with $* = \text{sgn}(p)$.

OUTPUT: P, the w-weighted power mean of order p.

Initialize X as the principal square root inverse of (2.23) if $p \in (0,1]$ or as its principal square root if $p \in [-1,0)$.

Set ζ equal to a small floating precision number (e.g., 10^{-10}).

Set $\phi = 0.375/|p|$.

REPEAT

$$H \leftarrow \sum_k w_k \left(X C_k^* X^T\right)^{|p|}$$
$$X \leftarrow H^{-\phi} X$$

UNTIL $\frac{1}{\sqrt{N}} \|H - I\|_F < \zeta$

RETURN
$$P = \begin{cases} X^{-1}X^{-T} & \text{if} \quad p \in (0,1], \\ X^T X & \text{if} \quad p \in [-1,0). \end{cases}$$

2.3.4.2 Geometric Mean Approximation by Power Means

As an approximation of the geometric mean of Section 2.3.3.2, the midpoint of the geodesic (2.13) is considered to join a pair of power means obtained by MPM at two small values $\pm p$ (in this research, $p = \pm 0.01$ is used). Current estimates of the geometric mean using the MPM algorithm were improved using this procedure.

2.3.5 Studies With Simulated Data

2.3.5.1 Simulated Data Model

In many engineering applications, the matrix condition number of the SPD matrices summarizing the data (observations, recordings,...) tends to be positively correlated with the number of sensors. Also, the dispersion in the manifold of the matrices is proportional to the noise level. The following generative model for input data matrices $C_1, \ldots, C_K$ of size $N \times N$ can able to reproduce these properties:

$$C_k = U D_k U^T + (V_k E_k V_k^T) + \alpha I, \tag{2.31}$$

where

- ***The signal*** part is given by $U D_k U^T$, where U is a matrix with elements drawn at random at each simulation from a uniform distribution in $[-1,1]$ and then

normalized to have columns with unit norm, and D_k are K diagonal matrices with diagonal elements $d_{k,n}$ randomly drawn at each simulation from a chi-squared random variable divided by its degree of freedom and multiplied by $1/2^n$. Thus, the expectation of each element is $1/2^n$, where $n \in \{1, \ldots, N\}$ is the index of the N diagonal elements; thus, forming elements of a well-known geometrical series absolutely converging to 1. The elements of the series represent the energy of N source processes, thus their sum is supposed to be finite (e.g. N brain dipole source processes with finite total energy).

- The ***uncorrelated noise part*** is given by αI, where I is the identity matrix and α here is taken as 10^{-6};

- The ***structured noise*** part is given by $V_k E_k V_k^T$, where the V_k matrices are generated as U above, the E_k matrices are generated as D_k previously and ν is a constant controlling the SNR of the generated points (2.31) through

$$\mathrm{SNR} = \frac{\mathrm{tr}\left(\sum_k U D_k U^T\right)}{\nu\left[\mathrm{tr}\left(\sum_k V_k E_k V_k^T + \alpha I\right)\right]}. \tag{2.32}$$

2.3.5.2 Simulation

The ensuing simulations studied relevant outcome parameters as a function of the SNR, which is inversely proportional to noise level as per (2.32), and a function of the size (N) and number (K) of input matrices. The gradient descent algorithm for estimating the geometric mean, (GDGM: Section 2.3.4, the naive fixed point algorithm for power means given in Lim and Pálfia (2012) (see (2.26) in Section 2.3.4) and the MPM algorithm here presented were compared, the latter for several values of p. In comparing the convergence rate of several algorithms, the stopping criterion should be determined to be identical for all of them. In addition, the relative error of matrix P with respect to a reference matrix P_{ref} is a dimensionless measure defined as follows (Higham, 1997):

$$\|P - P_{ref}\|_F^2 \,/\, \|P_{ref}\|_F^2 \,, \tag{2.33}$$

As a stopping criterion, considering two successive iterations $P_{(i-1)}$ and $P_{(i)}$, the following was used:

$$\frac{1}{N}\left\|P_{(i)}^{-1} P_{(i-1)} - I\right\|_F^2 \tag{2.34}$$

which magnitude does not depend on the size or on the norm of the matrices.
Simulated data was also used to study the estimation of the geometric mean obtained

by the gradient descent algorithm and by the procedure that uses the MPM algorithm, as per Section 2.3.4. We are interested in the relative error (2.33) of these estimations with respect to the "true" geometric mean: according to our data generating model (2.32), the true geometric mean is the geometric mean of the signal part given by matrices UD_kU^T, where $D_k, k = 1, \ldots, K$ are diagonal matrices. Because of the congruence invariance of the geometric mean, the true geometric mean is $G_{\mathcal{G}}(UD_1U^T, \ldots, UD_KU^T; w) = UG_{\mathcal{G}}(D_1, \ldots, D_K; w)U^T$ and has an algebraic solution, since the geometric mean of diagonal matrices is their Log-Euclidean mean (Arsigny *et al.*, 2007) i.e.

$$G_{\mathcal{G}}(D_1, \ldots, D_K; w) = \exp \sum_k w_k \log(D_k). \tag{2.35}$$

2.3.5.3 Results

Figure 2.4 shows the typical convergence behavior for the gradient descent algorithm for computing the geometric mean (GDGM), the naive algorithm with $p = 0.5$ and the MPM algorithm ($p = 0.5$ and $p = 0.001$), for $K = 100$ input SPD matrices of dimension $N = 20$, and SNR $= \{100, 10, 1, 0.1\}$. This example illustrates the typical observed trend: the MPM algorithm is consistently faster compared to both the naive and gradient descent algorithm. Moreover, the MPM algorithm also converges in situations when the gradient descent and the naive algorithm do not (see also Figure 2.5).

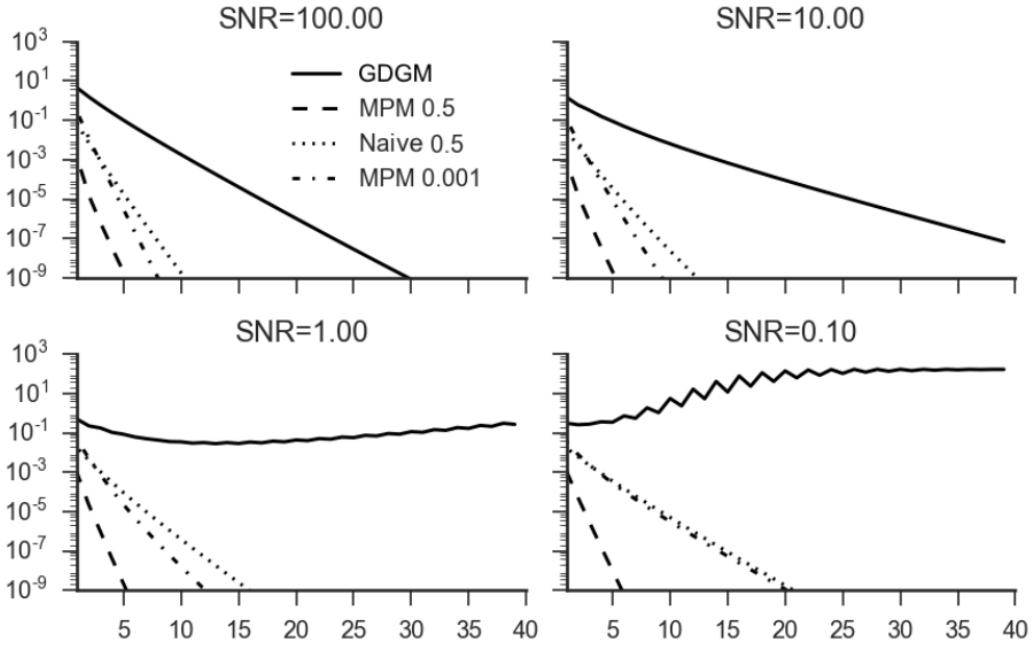


FIGURE 2.4: Typical convergence behavior (on abscissa, the number of iterations, and on the ordinate, the convergence as defined in (2.34)) on simulated data for the gradient descent algorithm for estimating the geometric mean (GDGM), naive fixed point power mean with $p = 0.5$ and the MDM algorithm with $p = \{0.5, 0.001\}$, for $N = 20$ (dimension of input matrices), $K = 100$ (number of input matrices) and SNR $= \{100, 10, 1, 0.1\}$ (2.32).

Figure 2.5 shows the analysis of the convergence behavior of the naive fixed point, the MPM fixed point and GDGM. The figure shows the main effects (bars) and their standard deviation (sd: lines) across 50 simulations of $N = \{10, 25, 50\}$, $K = \{10, 100, 500\}$ and SNR $= \{100, 1, 0.01\}$ on the number of iterations. Main effects means that for each level of N, K and SNR, the average and sd of the number of iterations are computed across all levels of the other two variables, as in a classical analysis of variance (ANOVA). The results show that the number of iterations required by the MPM algorithm is always smaller as compared to the naive algorithm and that the naive algorithm converges very slow or does not converge at all for $p = 0.01$ (the maximum number of iterations allowed was fixed to 50 for all algorithms).

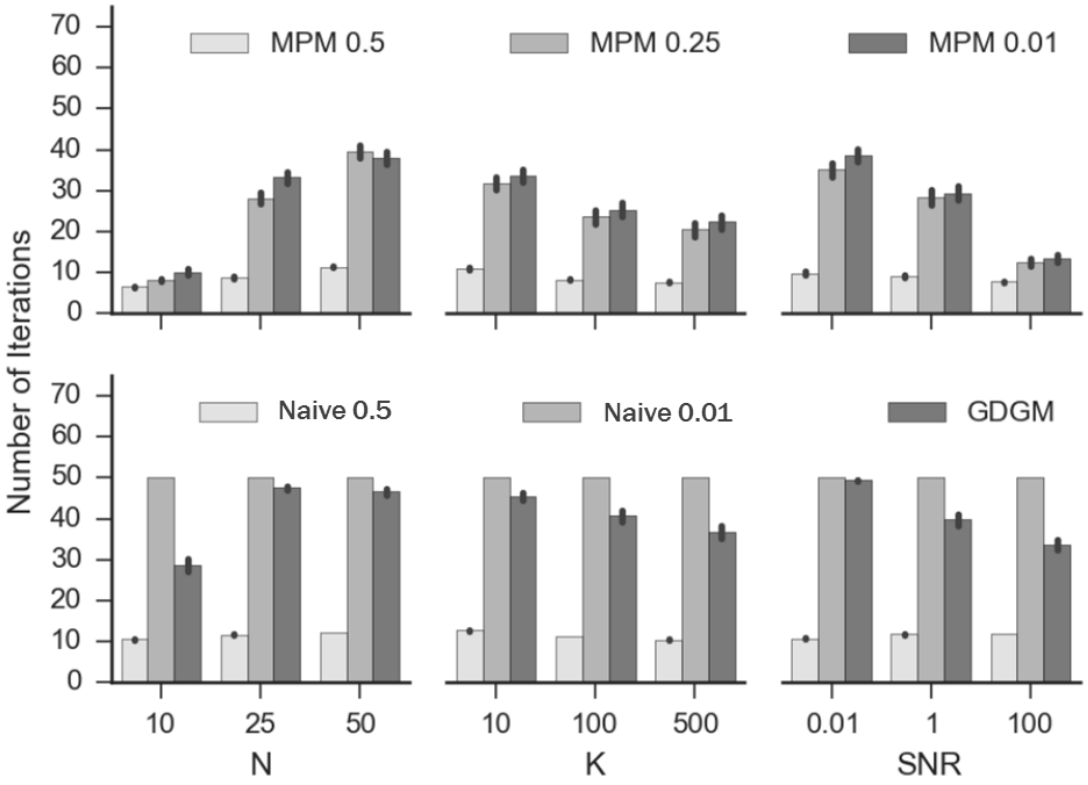


FIGURE 2.5: main effects average (bars) and sd (lines) number of iterations obtained across 50 repetitions for $N = \{10, 25, 50\}$, $K = \{10, 100, 500\}$ and SNR$= \{100, 1, 0.01\}$ for the MPM algorithm with $p = \{0.5, 0.25, 0.01\}$, the naive algorithm with $p = \{0.5, 0.01\}$ and the gradient descent algorithm for estimating the geometric mean (GDGM)

Figure 2.6 shows the relative error to the true geometric mean of the GDGM algorithm, MPM with $p = 0.1, 0.01$ and of the middle point of the geodesic joining the two MPM estimations obtained with $p = \pm 0.01$ (see Section 2.3.4), for several SNR in the range SNR$= \{10^{-3}, \ldots, 10^{3}\}$, $N = 20$, and $K = 5$ (left) or $K = 80$ (right). For all smaller SNR values (more noise than signal), all MPM-based estimations are closer to the true geometric mean as compared to the estimation offered by the gradient descent algorithm and that for all SNR values the midpoint of the geodesic joining the MPM estimations obtained with $p = \pm 0.01$ is as good as the best competitor, or better. Considering this and the convergence behavior of the MPM algorithm (Figure 2.5), we conclude that the procedure based on MPM described on section 2.3.4 is preferable for estimating the geometric mean.

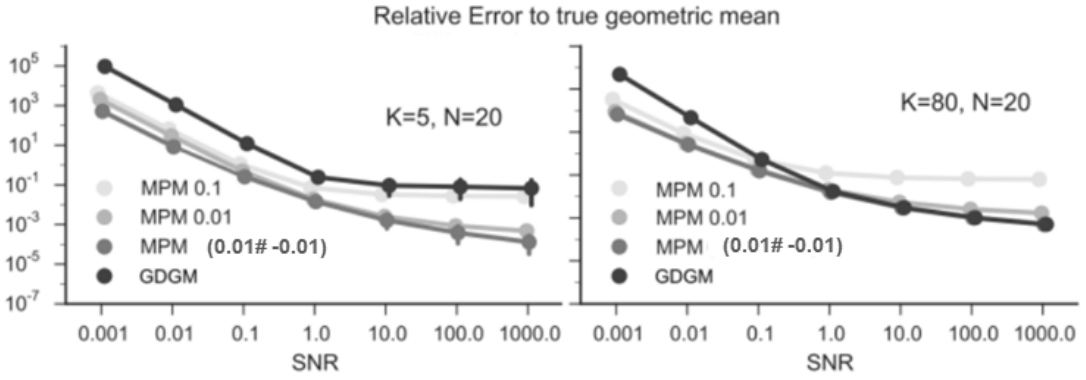


FIGURE 2.6: Relative error to the true geometric mean obtained with the GDGM algorithm, MPM with $p = 0.1$, MPM with $p = 0.01$ and as the midpoint of the geodesic joining the estimations obtained by MPM with $p = \pm 0.01$ (Section 2.3.4). Left: $N = 20$, $K = 5$. Right: $N = 20, K = 80$. In both plots, the horizontal axis is the SNR sampling the range $\{10^{-3}, \ldots, 10^{3}\}$.

2.3.6 Studies with Real Data

2.3.6.1 Procedures

We tested the classification performance obtained by several power means on a real electroencephalography (EEG) data set acquired at the GIPSA-lab in Grenoble on 38 pairs of subjects participating in a BCI experiment. The BCI used was the multi-subject Brain Invaders (Korczowski *et al.*, 2015), which the user-interface is similar to the joystick-controlled vintage video-game Space Invaders (Congedo *et al.*, 2011). The BCI shows for several levels of the game 36 aliens on the screen and flash them in random patterns of 6 aliens (Congedo *et al.*, 2011). The task of the participant is to destroy a TARGET alien only by concentrating on it (i.e. without moving at all). The on-line classifier analyzes the event-related potentials (ERPs) produced during 1s after each flash and decides after every sequence of 12 flashes what alien is to be destroyed. The level continues until the TARGET alien is destroyed or 8 attempts have failed, after which a new level begins. For this analysis, power means of special covariance matrices (see Section 2.2.2) for the TARGET and NON-TARGET ERPs were estimated on a training set, and the remaining trials were used for producing the area under the ROC curve (AUC). An AUC equal to 1 indicates perfect classification accuracy, while an AUC equal to 0.5 indicates random classification accuracy. The Riemannian classifier described in (Congedo, 2013) and Section 2.4.2.1 was employed, which only uses the means of SPD matrices and distance function (2.14) to reach a decision. In the experiment, across subjects the average (sd) numbers of TARGET and NON-TARGET trials available were 109.9 (26.2) and 549.48 (130.1), respectively. In order to keep the amount

of data constant across subjects, only the first 80 TARGET and 400 NON-TARGET trials were used. AUC is evaluated by using a Monte Carlo cross-validation (MCCV) procedure averaging 10 random samples comprising 25% of the data selected as the test set and the remaining used as training set. EEG data were acquired by 16 scalp electrodes. Power means were tested at values of $p \in \{\pm 1, \pm 0.8, \pm 0.6, \pm 0.4, \pm 0.2, \pm 0.1, 0\}$.

2.3.6.2 Results

The individual area under the ROC curve (AUC) for the BCI experiment on 38 subjects is shown in Figure 2.7. The AUC values are obtained based on the minimum distance to mean (MDM) classification rule which is looking for the correspond class of a trial in the test set which has the minimum riemannian distance with the power mean of TARGET or NON-TARGET trails in the training set. The MDM is discussed in Section 2.4.2.1. The AUC as a function of p is a smooth curve and the value of p offering the maximum AUC appears to gravitate around zero. This illustrates a reason why the geometric mean is found useful in practice. However, the geometric mean ($p = 0$) is optimal only for three out of the 38 subjects, and the optimal value of p is highly variable across individuals. This demonstrates that the use of power means instead of the sole geometric mean has potential to increase the accuracy. Finally, the Pearson correlation between the maximal value of AUC obtained and the corresponding value of p is 0.49. A statistical test for the null hypothesis that this correlation is equal to zero against the alternative hypothesis that is larger than zero, gives a probability of type I error equal to 0.002. Therefore, the null hypothesis is rejected and a higher AUC, that is, a higher SNR of the data correlates to the higher the optimal value of p. This result matches our intuition: when the noise is higher than the signal, a power mean with a negative p will suppress the noise more than the signal and vice versa.

2.3.7 Mean fields

The family of power means is continuous and monotonic. Figure 2.8 is a TraDe plot (log-trace vs. log-determinant) for a sampling of power means along continuum $p \in [-1, 1]$, illustrating the monotonicity of power means. We name a sampling of power means like those in 2.7 and 2.8 a Pythagorean Mean Field. Applications of mean fields include the possibility to evaluate the most appropriate choice of mean depending on its use and on the available data. Mean fields also allow robust extensions of current Riemannian classifiers, such as in (Arnaudon *et al.*, 2013; Barachant *et al.*, 2012, 2013; Congedo, 2013; Kalunga *et al.*, 2016; Li and Wong, 2013; Li *et al.*, 2012; Moakher, 2006; Zhang *et al.*, 2016). For instance, we may want to combine Riemannian classifiers

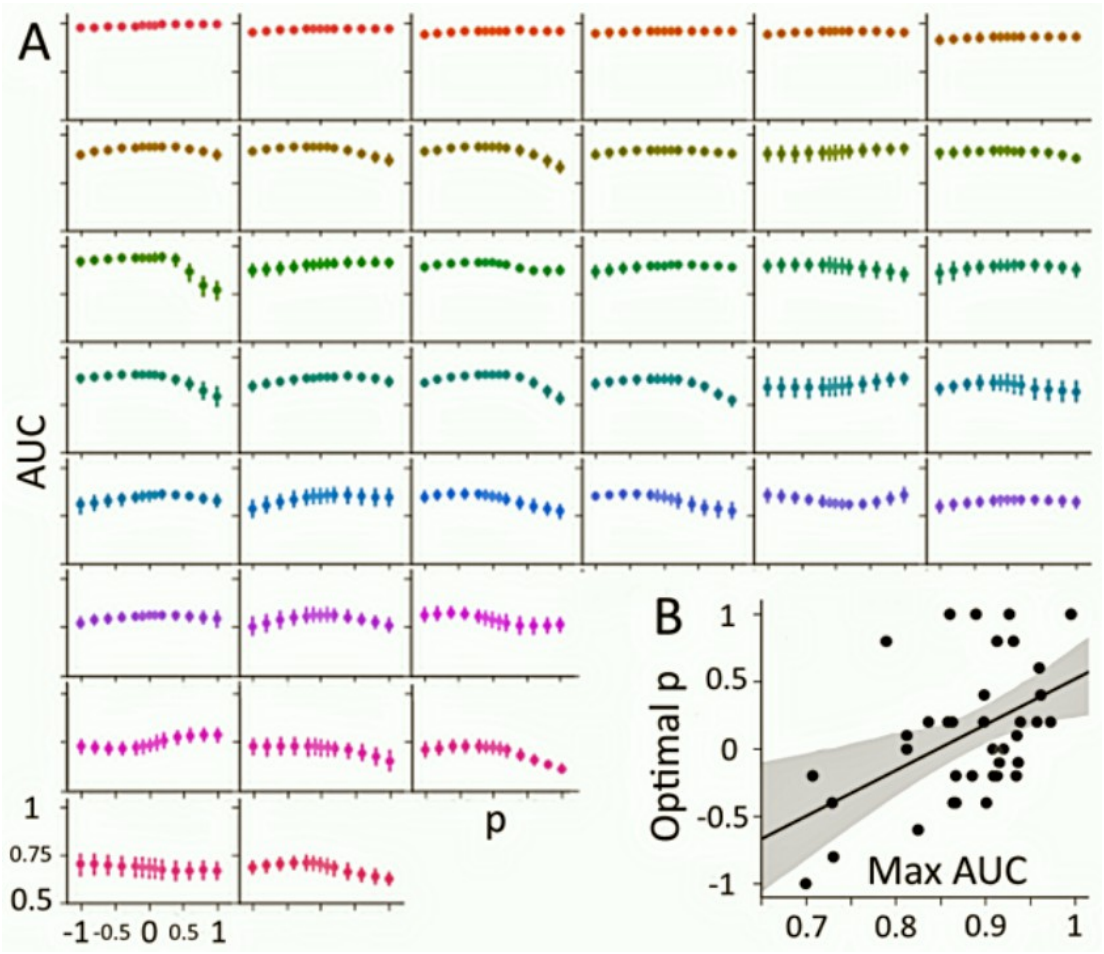


FIGURE 2.7: A: from left to right and from top to bottom, AUC (disks) ± one standard deviation (vertical bars) obtained for 38 healthy subjects sorted by decreasing value of maximal AUC obtained across a sampling of power means in the interval $p = \{-1, \ldots, 1\}$. B: scatter plot and regression line of the maximal AUC and the value of p allowing the maximal value. Each disk represents a subject.

applied to all the points of a mean field. The application of mean fields to real data will be the object of next sections.

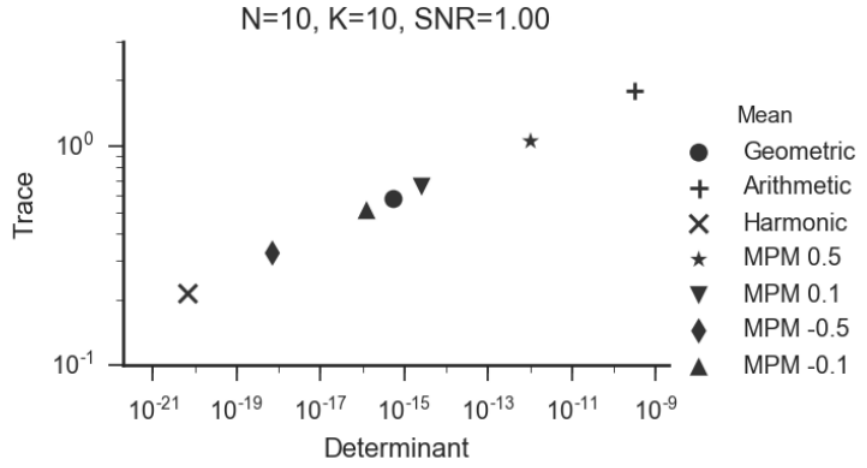


FIGURE 2.8: TraDe plot obtained with N=10, K=10 and SNR=1 for power means corresponding to $p = 1$ (arithmetic), $0.5, 0.1, 0$ (geometric), $-0.1, -0.5$ and -1 (harmonic). The relationship between the trace and the determinant of power means is log-log linear.

2.3.8 Conclusions

Power means are generalized means interpolating continuously in the interval $p \in [-1, 1]$, with $p = 1$ yielding the arithmetic mean, the limit of $p \to 0$ from both sides yielding the geometric mean and $p = -1$ yielding the harmonic mean. A new multiplicative algorithm of estimating power means of SPD matrices in the interval $p \in (-1, 1)\backslash\{0\}$ has been presented. Furthermore, a numerical analysis shows that its convergence rate is very fast and quasi-uniform for values of p close to $1/2$ and $-1/2$, while for values of p close to 0 or ± 1 it is still faster as compared to when the gradient descent with fixed step-size used to estimate the geometric mean. Furthermore, it converges also in low SNR situations, whereas the gradient descent algorithm fails. The approximation to the geometric mean proposed in Section 2.3.4 provides better estimates of the geometric mean with respect to the gradient descent algorithm. We can therefore prefer MPM also for estimating the geometric mean. In conjunction with the procedure for $p = 0$ of Section 2.3.4 and expression (2.17) and (2.18) for $p = 1$ and $p = -1$, respectively, the MPM algorithm can now estimate a number of means sampling along the continuum $p = [-1, 1]$.

2.4 Statistical Combinations of Power Means: Classification Study on Functional Data

2.4.1 Introduction

Manipulating functional data in machine learning studies has been highlighted in many practical studies, and the amount of interest in this field has been increasing. As a big data problem, the classification study of the functional data when the data appears as covariance matrices is proposed. Covariance matrices form a differentiable Riemannian manifold. Regarding this fact, some classification approaches are proposed, and they are assessed in terms of accuracy. As mentioned, there is a steadily growing interest in classification methods for functional data, they often exploit Riemannian geometry (see Barachant *et al.* (2010, 2012, 2013); Congedo *et al.* (2013); Korczowski *et al.* (2015)), therewith in this research the classification problem of functional data rising from EEG signal in BCI is considered. In brief, each observation to be classified is the brain activity (i.e. multiple electrodes) over a fixed period of time. Estimating the average of available sample covariance matrices is a crucial step in such classification problems. The Riemanian manifold of SPD matrices in coincidence with Riemannian geometry techniques are well adopted in BCI classification, and they provide a rich framework to manipulate in this context; see Section 2.3. Therefore, some classification approaches that use the mean field of covariance matrices on their manifold are proposed. As the univariate case, the best employed mean estimator to higher accuracy classification can be different from arithmetic or geometric means. In fact, a combination of power means is presented to provide the benefits of all power means regardless of the matrix data distribution, as discussed in the beginning introduction of the thesis, EEG part. For the classification of functional data the power means of covariance matrices, employing the MPM (provided in 2.3.4) and the MDM (minimum distance to mean; see 2.4.2.1) algorithms, are used. The behavior of different power means and their combinations are assessed in terms of classification accuracy using real data and the merits of proposed approaches are shown. Up to now, only the geometric mean has been used for such classification studies while our results showed that the optimal mean which produced the maximal accuracy could be different in the mean field of power means; however, no educated guesses regarding the optimum p may be possible.

2.4.2 Classification Methodologies

2.4.2.1 MDM (minimum distance to mean) Classification

As the classification methodology, for all classifications, a simple idea is used, namely MDM Congedo (2013). Based on it, once the training set (a bunch of trials which are SPD matrices) are obtained, the power means with several orders p for different classes are estimated. Then, the Riemanian distance between the correspond SPD matrix of the new case (a trial with unknown class) and each power means matrix of different classes are calculated. The predicted class for the new case, based on a fixed p, is the correspond class in which its power mean has the minimum Riemaninan distance from new the case.

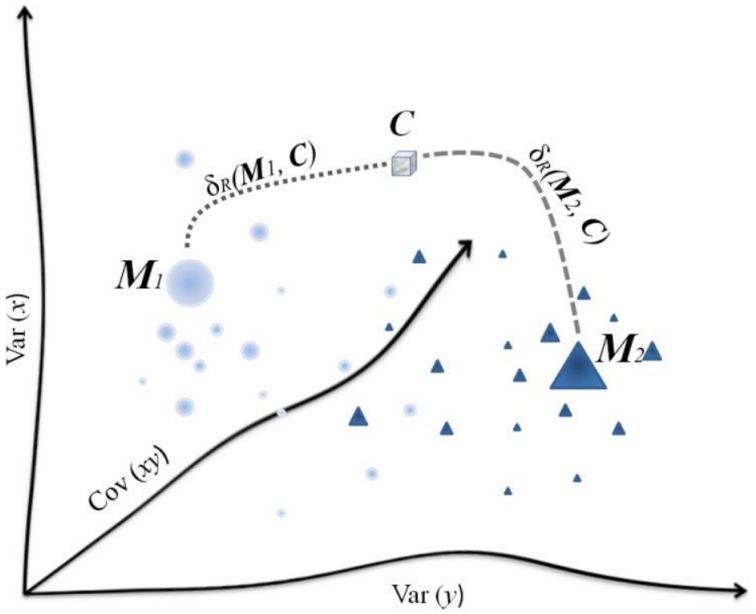


FIGURE 2.9: Schematic of MDM. C is a new observation (matrix) and M_1 and M_2 are the center of masses in two different groups (Congedo *et al.*, 2013).

In BCI, the trials need to be classified regardless of the kind of data: motorimagery (MI) trials, steady-state evoked potentials (SSEP) trials, or event-related potentials (ERP). Suppose there are several training trials with different K classes and for a new unlabeled trial C, which all are in the form of a covarinace matrices (Section 2.2.2) one of the K classes should be assigned. Considering this fact, the right metric is Riemanian on the SPD manifold we may want to compute the mean of the each classes in training set, $M_i; i = 1, \ldots, K$, and then look for the shortest Riemanian distance (Section 2.14) between C and M_i.

Algorithm 1: MDM
Input: set of trials C_{ij} (covariance matrices) of $j = 1, \ldots, K$ classes.
Input: C, the covariance matrix of new trial with an unknown class.
Output: $\hat{k}$, the predicted class of the new trial.
for $j = 1$ to K do
$G_{\mathcal{P}_j}^{(p)} = G_{\mathcal{P}}(C_{ij}; w; p)$
end **for**
$\hat{k} = \underset{j}{\operatorname{argmin}}\ \delta(G_{\mathcal{P}_j}^{(p)}, C)$

TABLE 2.1: Minimum distance to mean (MDM) algorithm for classification using power means of SPD matrices.

2.4.2.2 Application to Motorimagery data

The theory is now applied to our leading example. Figure 2.10 shows a general behavior of power means with $p \in \{\pm 1, \pm 0.75, \pm 0.50, \pm .25, 0\}$ in terms of classification accuracy of EEG signals on 9 subjects mentioned in the Section 2.1.1. As mentioned, power means have not been yet used in such classification problems. In addition, Figure 2.10 shows that the power mean which maximizes the accuracy slip between harmonic (i.e. $p = -1$) and arithmetic means (i.e. $p = +1$). However, we are not able to find any pattern or guess to find the optimal power mean, since the best p varies among subjects.

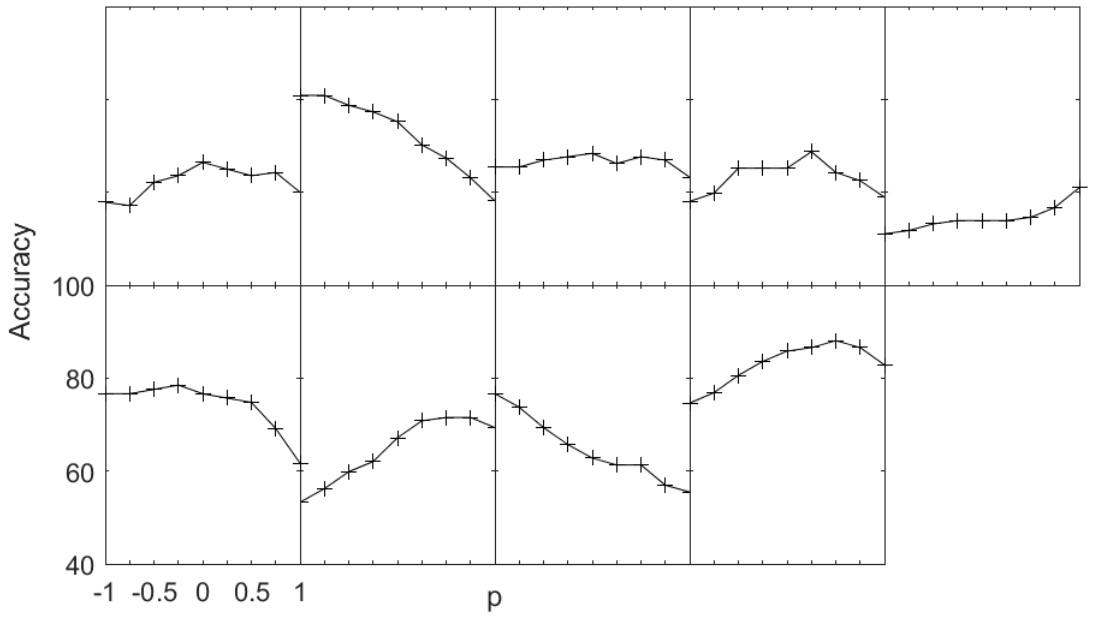


FIGURE 2.10: Classification accuracy on 9 subjects for the classes 3 vs 4 in Motorimagery task using MDM algorithm for the training and test sets in size of 288 trials. Power means with $p \in \{\pm 1, \pm 0.75, \pm 0.50, \pm 0.25, 0\}$ were estimated by MPM algorithm.

2.4.2.3 Combination of Power means

Although arithmetic mean seems to be the most usual and natural mean using MDM, but some applications on real data show it is the worth one, mostly! As mentioned, only geometric and arithmetic means has been used for EEG signals classifications, so far. Section 2.4.2.2 shows that neither arithmetic nor geometric means are optimal in every subject, while different p values have higher accuracy. However, it might not be possible to guess the optimal power mean facing a new subject. One could select the best p based on some cross-validation principle, but this approach is still far from being optimal, and results are often worse than simply using a prefixed p (e.g. geometric mean). Moving from these considerations, the idea of combining the classification of a set of power means has been arised. Furthermore, it is desirable that a combination is more affected by power means with better accuracy. Therefore, a combination of the classifications which is weighted depending on the accuracy of every power mean is presented.

Assuming two classes (having labels -1 and $+1$), the combined classification rule is the following:

$$ccr(C) = \operatorname{sgn}\left(\int_{-1}^{1} w(p)\ \operatorname{sgn}\left(\log\left(\frac{R(p,C)_1}{R(p,C)_2}\right)\right) dp\right), \tag{2.36}$$

where sgn is the sign function i.e. $\text{sgn}(x) = \begin{cases} 1 & x \geq 0 \\ -1 & x < 0, \end{cases}$ and $R(p, C)_j = \delta(C, G^{(p)}_{\mathcal{P}_j})$, while $G^{(p)}_{\mathcal{P}_j}$ is the power mean with order p related to the class j, and C is the correspond covariance matrix of a trial in new subject that needs to be classified. The empirical version of (2.36) on a set of orders of power means P can be obtained as

$$\hat{ccr}(C) = \text{sgn}\left(\sum_{p\in P} w(p)\ \text{sgn}\left(\log\left(\frac{R(p,C)_1}{R(p,C)_2}\right)\right)\right), \tag{2.37}$$

where the weights $w(p)$ are exponential transformed of some (scaled) accuracy, namely acc_p, of a pre-classification on a initial training set, i.e. $w(p) = \exp(acc_p); p \in P$. The pre-classification for obtaining weights can be done on the same training and test sets. The possible $\widehat{ccr}$ values of -1 or 1 show the predicted classes, respectively. In general, one should note that, depending on the available problem, exponential or sng functions might been replaced by some other desirable functions.

2.4.3 Application

This section presents the accuracy performance of classifications on two types of EEG data, namely, P300 and Motorimagenary data which has been presented in Section 2.1.1 and discussed throughout the paper.

2.4.3.1 Application to P300 data

The classification performance obtained by several power means on a real EEG data set were examined. The data set acquired at the GIPSA-lab in Grenoble, France, on 19 pairs of subjects participating in a BCI experiment. The BCI used was the multi-subject Brain Invaders Korczowski *et al.* (2015), which the user-interface is similar to the joystick-controlled vintage video-game Space Invaders Congedo *et al.* (2011). The BCI shows several times 36 aliens on the screen and flashes them in random pattern of 6 aliens Congedo *et al.* (2011). The task of the participant is to destroy a TARGET alien only concentrating on it. The on-line classifier analyzes the ERP produced during 1s after each flash. In the experiment, across subjects, the average (sd) number of TARGET and NON-TARGET trials available were 109.9 (26.2) and 549.48 (130.1). EEG data were acquired by 32 scalp electrodes, but only a subset of 13 electrodes that were found optimal on the average of all subjects were used here. Power means were tested with $P = \{\pm 1, \pm 0.75, \pm 0.5, \pm 0.25, 0\}$.

To do the classification with several power means, for each subject, a random training group of size n among all available TARGET and NON-TARGET trials was chosen and

the power means with different $p \in P$ according to two classes were computed. Then, the rest of trials were used as the test group to classify the TARGET and NON-TARGET trials by MDM. The random training set can be used also to obtain the weights $w(p)$s. To obtain the accuracy classification, this procedure was repeated M times and the average accuracy was computed. $M = 50$ was chosen to have stable results upon a certain precision. For combination, once the values of $w(p)$s were obtained, for a trail in the test group, $R(p, C)_1$ and $R(p, C)_2$ were computed and the predicted class of the new trial was obtained by using (2.37).

Algorithm 2: Classification accuracy by combined classification

Input: set of trials C_{ij} (i-th trial and j-th class) of S subjects with two classes.
Output: $cacc_s$, the combined accuracy in s-th subject
for $s = 1$ to S do
  for $m = 1$ to M do
    choose a random training group (T) of size n
    estimate $G_{\mathcal{P}_j}^{(p)}$ for classes $j = 1, 2$ and $p \in P$ on T
    compute the weights in (2.37) by MDM on T
    compute $c\hat{c}r$ in (2.37) for the rest of trials
  end **for**
$cacc_s$ =average accuracy in M loops
end **for**

TABLE 2.2: Algorithm to do classification by combination approach with M number of cross-validation using MPM and MDM.

Figure 2.11 shows a general behavior of power means in terms of classification accuracy. As mentioned previously, power means have not been used in such classification problems, so far. Figure 2.11 shows that the optimal power mean which maximizes the average accuracy is different from geometric mean, and in many cases, the arithmetic mean is the worse one (+ sign in the bounded area). However, no pattern or educated guess of the optimal power mean could be obtained.

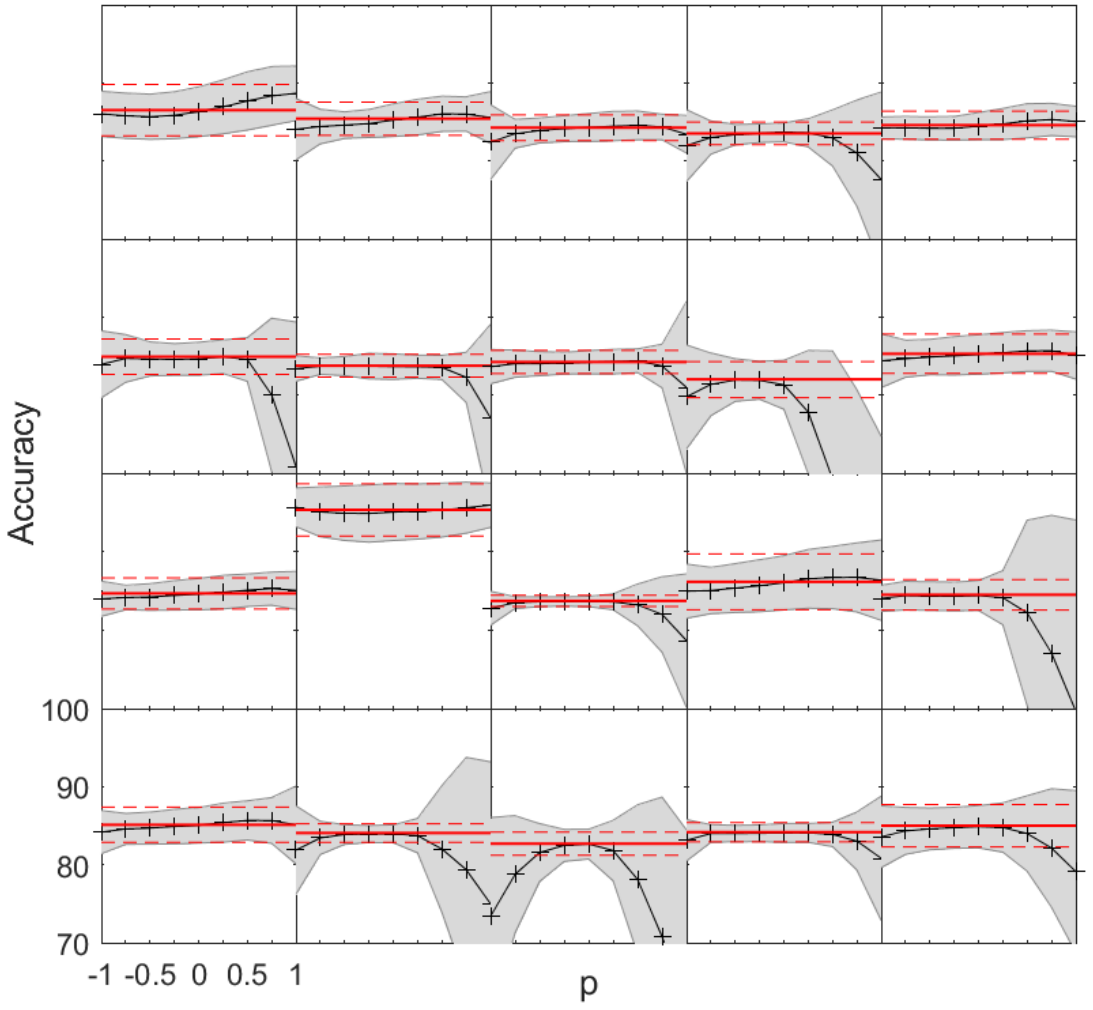


FIGURE 2.11: Accuracy of classification on 19 subjects in P300 data with $n = 25$ using power means by MPM and MDM algorithms. The solid line shows the average accuracy over $M = 50$ replication of combination approach bounded by 1 standard deviation by dashed line, and + shows the average accuracy classification over $M = 50$ repeats using several power means with $P = \{\pm 1, \pm 0.75, \pm 0.5, \pm 0.25, 0\}$ bounded by the area of 1 standard deviation. The last plot on the right bottom side shows the average accuracy over all subjects.

	subjects									
	1	2	3	4	5	6	7	8	9	10
Geometric mean	85.96	84.85	84.09	84.01	85.30	85.19	83.86	84.23	81.11	85.56
Best p	85.63	84.43	83.10	83.36	84.75	84.59	83.07	83.97	81.08	84.65
Combination	86.16	85.03	84.15	83.97	85.29	85.40	83.92	84.26	81.91	85.62
	subjects									
	11	12	13	14	15	16	17	18	19	Ave.
Geometric mean	85.62	94.84	83.64	85.27	84.59	85.37	83.56	82.79	83.95	84.94
Best p	85.38	95.51	82.08	84.64	83.97	85.10	82.30	79.84	83.58	84.27
Combination	85.81	95.14	83.61	85.35	84.65	85.49	83.56	82.54	83.93	85.04

TABLE 2.3: Accuracy of classification with $n = 25$ on 19 subjects using geometric mean, best p and the combination approach with $M = 50$.

In the Figure 2.11, however, we see that the optimal power mean can be different, by combination approach (solid lines) we can catch an accuracy almost close to the accuracy of optimal power mean. So in practice, by using combination approach, while we do not know which power mean is the optimal one, we can have an enough trusted accuracy with respect to the optimal power mean. In Table 2.3, first on the random initial training and test sets of size n, a pre-classification is performed to find the best p. Then, using that best p, which was mostly different from the geometric mean, a classification is performed on the rest of trials, and the average accuracy is obtained by repeating this procedure M times to obtain the accuracy by cross validation. The combination method has the higher accuracy in most of the subjects and also in average over all subjects. Also, it is verified by paired t-test.

2.4.3.2 Motorimagery data

We use the Motorimagery data from BCI Competition 2008 Graz data set A; see Section 2.1.1. There were 22 Ag/AgCl electrodes (with inter-electrode distances of 3.5 cm) used to record the EEG. All signals were recorded monopolarly with the left mastoid serving as reference, and the right mastoid serving as the ground. The signals were sampled with 250 Hz and band-pass filtered between 0.5 Hz and 100 Hz. The sensitivity of the amplifier was set to 100 μV. An additional 50 Hz notch filter was enabled to suppress line noise. To perform the classification with several power means, for each subject in classes 3 and 4, a random training group of size n among all available trials was chosen, then, the power means according to the two classes and $w(p)$ with different $p \in P = \{\pm 1, \pm 0.75, \pm 0.50, \pm 0.25, 0\}$ were computed. Then, the rest trials

were used as the test group to classify by MDM. To obtain the accuracy classification by cross validation, this procedure was repeated M times and the average accuracy was computed. $M = 50$ was chosen to have stable results upon a certain precision. For combination, once the value of $w(p)$s were obtained, for a trail in the test group, $R(p, C)_1$ and $R(p, C)_2$ were computed and we obtained the predicted class of the new trial by using (2.37); see Algorithm 2. Because of the similarity of classification results of all pairs of classes, to save more pages the results of the classes 3 and 4 were considered here and the classification results between the other classes pairs in this data set are provided in the Appendix.

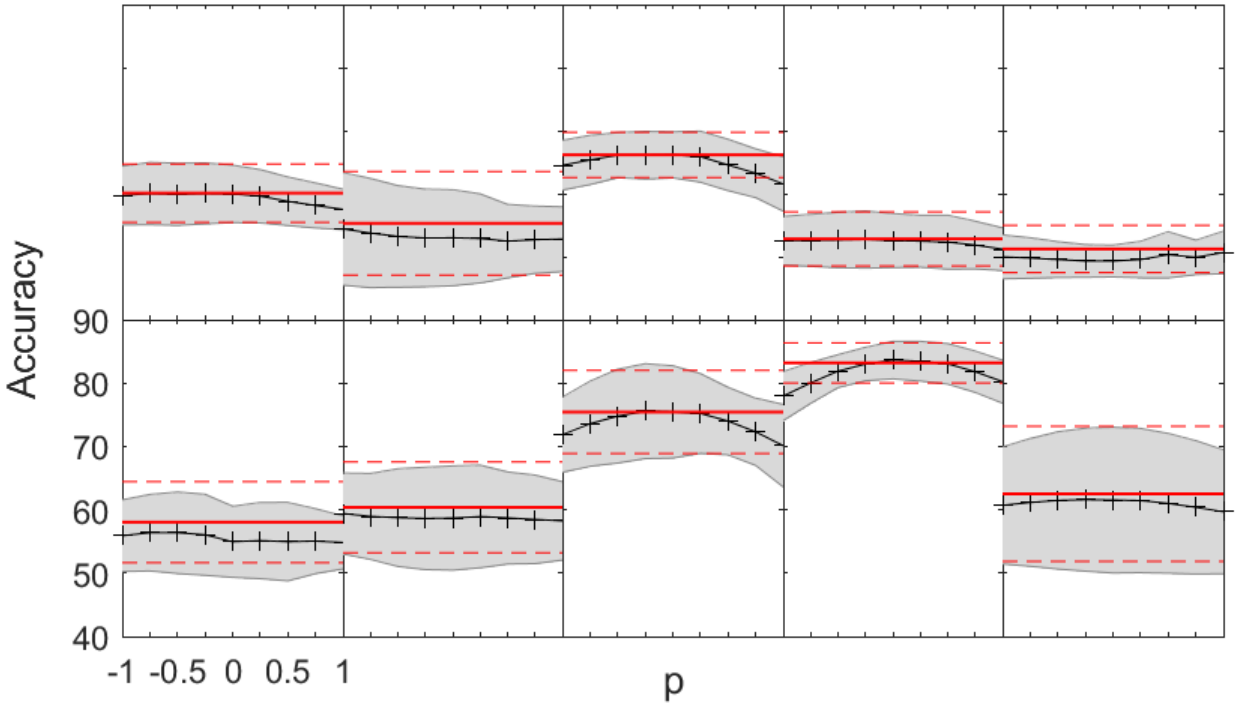


FIGURE 2.12: Average accuracy of classification for class 3 vs 4 on 9 subjects in Motorimagery data for $n = 50$ (576 trials) using power means by MPM and MDM algorithms. The solid line shows the average accuracy over $M = 50$ replication of combination approach bounded by 1 standard deviation by dashed line. The + shows the average accuracy classification over $M = 50$ repeats using several power means with $P = \{\pm 1, \pm 0.75, \pm 0.5, \pm 0.25, 0\}$ bounded by the area of 1 standard deviation. The last plot on the right bottom side shows the average accuracy over all subjects.

	subjects									
	1	2	3	4	5	6	7	8	9	Ave.
Geometric mean	59.97	53.01	66.16	52.56	49.31	54.94	58.69	75.45	83.63	61.53
Best p	60.33	56.71	63.73	54.77	51.40	58.42	60.76	71.33	78.62	61.79
Combination	60.08	55.30	66.15	52.82	51.22	58.05	60.39	75.43	83.19	62.51

TABLE 2.4: Accuracy of classification with $n = 50$ on 9 subjects class 3 vs 4 Motorimagery data using geometric mean, best p and combination approach with $M = 50$.

Figure .2 shows using the combination approach that an accuracy close to the optimal power mean can be obtained and that an accuracy higher than the accuracy obtained using the optimal power mean can be obtained in some subjects. Moreover, the last plot on the right bottom in Figure .2 shows that in average over all subjects, the combined accuracy is higher than all power means and that the optimal power mean is different from the geometric mean. Also, the combination with the best p and geometric mean were compared in Table 2.4. The combination approach has higher accuracy, however, the paired t-test showed combination approach and best p are not significantly different.

2.4.4 Conclusion

This work proposes the classification study of functional data when the working data appeared as sample covariance matrices. This can be seen in EEG signals in BCI which is indeed useful when the subject wants to control the machine by brain commands such as video-games or for people with physical disabilities. In this thesis, the features of Riemanninan manifold of SPD matrices and Riemanninan geometry techniques were employed to do classification, and an efficient and fast algorithm MPM was proposed in 2.3.4 and Congedo *et al.* (2017) to estimate power means. Up to now, only geometric mean has been used for such classification studies however, there is a convergence problem in some cases when computing the geometric mean using the previous existing algorithms. Our results showed that the optimal mean which caused the maximum accuracy could be different in the mean field of power means; however, it may not be possible to have any guess, and interestingly this topic has not been covered so far! Finally, a combination approach of power means was proposed. Using the combination approach, accuracy is close enough, or higher than the accuracy of the optimal power mean in some cases. Moreover, on average within all subjects , the accuracy of the combination approach was higher than the best p and geometric mean. Thus, in practice, the user can apply the combination approach while not knowing which power mean supplies the maximum accuracy.

Appendix

Other results of the classification study related to the motorimagery real data in Section 2.4.3.2.

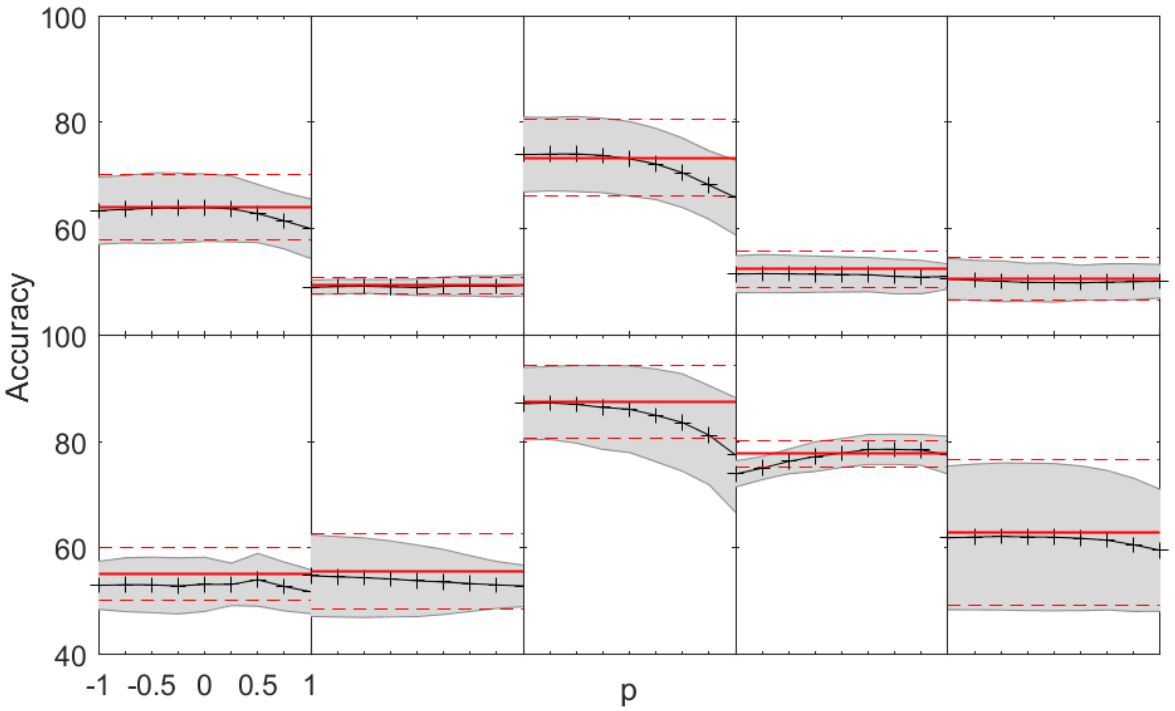


FIGURE .1: Average accuracy of classification for class 1 vs 2 on 9 subjects in Motorimagery data for $n = 50$ (576 trials) using power means by MPM and MDM algorithms. The solid line shows the average accuracy over $M = 50$ replication of combination approach bounded by 1 standard deviation by dashed line, and + shows the average accuracy classification over $M = 50$ repeats using several power means with $P = \{\pm 1, \pm 0.75, \pm 0.5, \pm 0.25, 0\}$ bounded by the area of 1 standard deviation. The last plot on the right bottom side shows the average accuracy over all subjects.

	subjects									
	1	2	3	4	5	6	7	8	9	Ave.
Geometric mean	63.97	48.99	73.14	51.42	49.90	53.17	53.84	86.09	77.85	62.04
Best p	63.97	50.12	72.29	53.40	51.99	55.66	56.56	86.46	78.69	63.24
Combination	64.10	49.31	73.36	52.45	50.65	55.15	55.59	87.46	77.76	62.87

TABLE .1: Accuracy of classification with $n = 50$ on 9 subjects class 1 vs 2 motorimagery data using geometric mean, best p and combination approach with $M = 50$.

	Best p	Combination
Geometric mean	Rejected	Rejected
Best p	*	Accepted

TABLE .2: Paired t-test for the accuracy of classification with $n = 50$ on 9 subjects class 1 vs 2 motoimagery data using geometric mean, best p and combination approach with $M = 50$. Each cell shows the decision about null hypothesis which is mean equality of two groups.

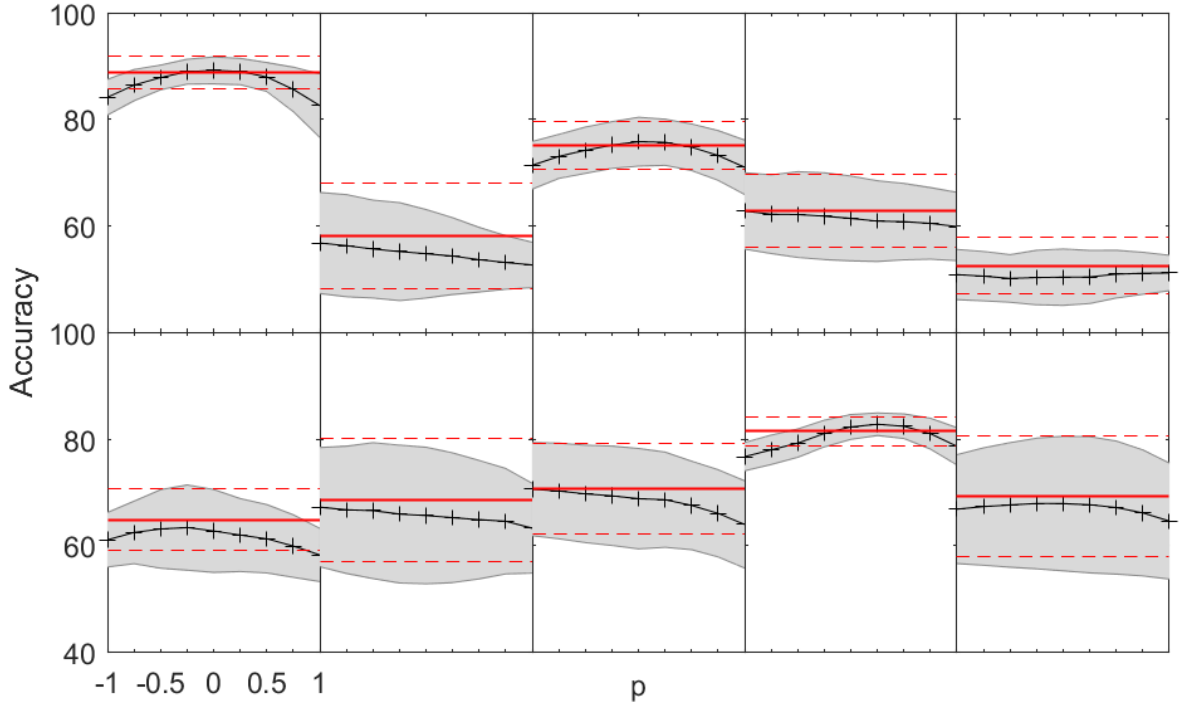


FIGURE .2: Average accuracy of classification for class 1 vs 3 on 9 subjects in motorimagery data for $n = 50$ (576 trials) using power means by MPM and MDM algorithms. The solid line shows the average accuracy over $M = 50$ replication of combination approach bounded by 1 standard deviation by dashed line, and + shows the average accuracy classification over $M = 50$ repeats using several power means with $P = \{\pm 1, \pm 0.75, \pm 0.5, \pm 0.25, 0\}$ bounded by the area of 1 standard deviation. The last plot on the right bottom side shows the average accuracy over all subjects.

	subjects									
	1	2	3	4	5	6	7	8	9	Ave.
Geometric mean	89.22	54.82	75.85	61.44	50.42	62.78	65.67	68.84	82.30	67.93
Best p	88.26	63.43	72.72	65.03	56.71	67.17	70.14	71.58	78.72	70.42
Combination	88.77	58.14	75.15	62.88	52.59	64.86	68.54	70.79	81.48	69.25

TABLE .3: Accuracy of classification with $n = 50$ on 9 subjects class 1 vs 3 motorimagery data using geometric mean, best p and combination approach with $M = 50$.

	Best p	Combination
Geometric mean	Accepted	Rejected
Best p	*	Accepted

TABLE .4: Paired t-test for the accuracy of classification with $n = 50$ on 9 subjects class 1 vs 3 motorimagery data using geometric mean, best p and combination approach with $M = 50$. Each cell shows the decision about null hypothesis which is mean equality of two groups.

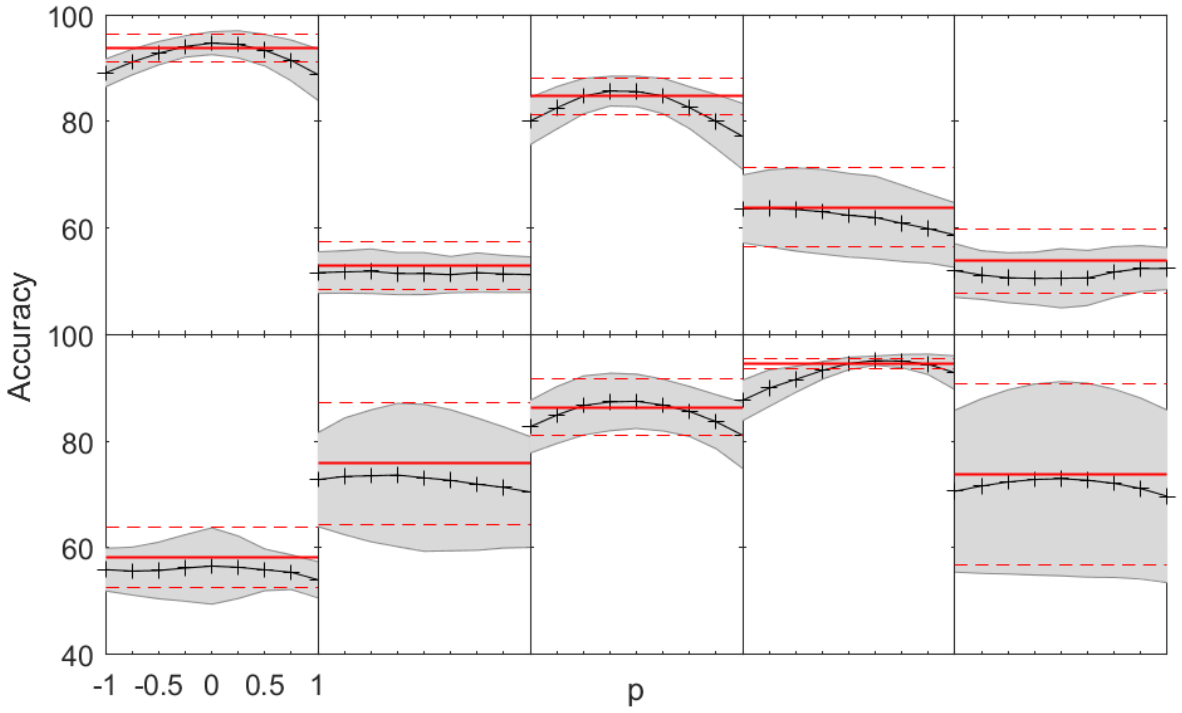


FIGURE .3: Average accuracy of classification for class 1 vs 4 on 9 subjects in motorimagery data for $n = 50$ (576 trials) using power means by MPM and MDM algorithms. The solid line shows the average accuracy over $M = 50$ replication of combination approach bounded by 1 standard deviation by dashed line, and + shows the average accuracy classification over $M = 50$ repeats using several power means with $P = \{\pm 1, \pm 0.75, \pm 0.5, \pm 0.25, 0\}$ bounded by the area of 1 standard deviation. The last plot on the right bottom side shows the average accuracy over all subjects.

	subjects									
	1	2	3	4	5	6	7	8	9	Ave.
Geometric mean	94.72	51.46	85.65	62.41	50.58	56.57	73.13	87.50	94.59	72.96
Best p	92.83	54.40	80.93	67.29	58.62	61.09	75.78	83.53	89.53	73.78
Combination	93.89	52.86	84.74	63.91	53.81	58.15	75.80	86.39	94.55	73.79

TABLE .5: Accuracy of classification with $n = 50$ on 9 subjects class 1 vs 4 motorimagery data using geometric mean, best p and combination approach with $M = 50$.

	Best p	Combination
Geometric mean	Accepted	Accepted
Best p	*	Accepted

TABLE .6: Paired t-test for the accuracy of classification with $n = 50$ on 9 subjects class 1 vs 4 motorimagery data using geometric mean, best p and combination approach with $M = 50$. Each cell shows the decision about null hypothesis which is mean equality of two groups.

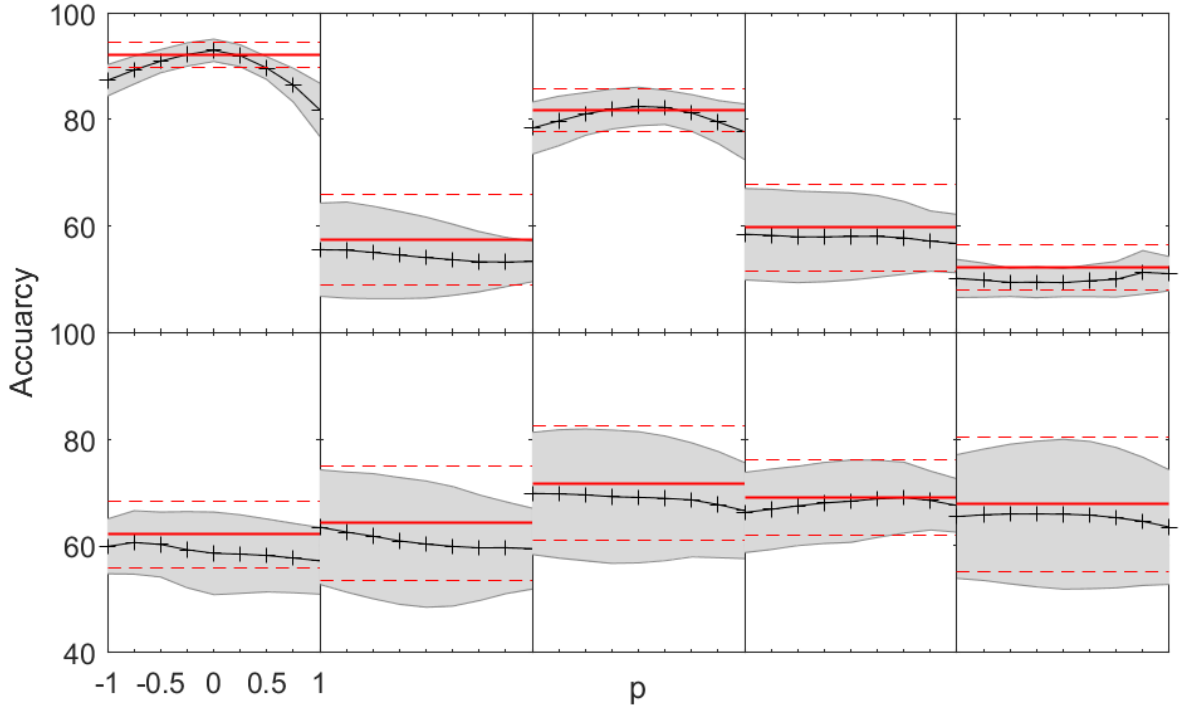


FIGURE .4: Average accuracy of classification for class 2 vs 3 on 9 subjects in motorimagery data for $n = 50$ (576 trials) using power means by MPM and MDM algorithms. The solid line shows the average accuracy over $M = 50$ replication of combination approach bounded by 1 standard deviation by dashed line, and + shows the average accuracy classification over $M = 50$ repeats using several power means with $P = \{\pm 1, \pm 0.75, \pm 0.5, \pm 0.25, 0\}$ bounded by the area of 1 standard deviation. The last plot on the right bottom side shows the average accuracy over all subjects.

	subjects									
	1	2	3	4	5	6	7	8	9	Ave.
Geometric mean	92.99	54.09	82.45	58.05	49.39	58.58	60.31	69.10	68.37	65.93
Best p	91.58	61.66	79.68	61.71	52.26	63.99	65.53	71.90	69.09	68.60
Combination	92.18	57.34	81.69	59.70	52.25	62.19	64.26	71.74	69.10	67.83

TABLE .7: Accuracy of classification with $n = 50$ on 9 subjects class 2 vs 3 motorimagery data using geometric mean, best p and combination approach with $M = 50$.

	Best p	Combination
Geometric mean	Rejected	Rejected
Best p	*	Accepted

TABLE .8: Paired t-test for the accuracy of classification with $n = 50$ on 9 subjects class 2 vs 3 motorimagery data using geometric mean, best p and combination approach with $M = 50$. Each cell shows the decision about null hypothesis which is mean equality of two groups.

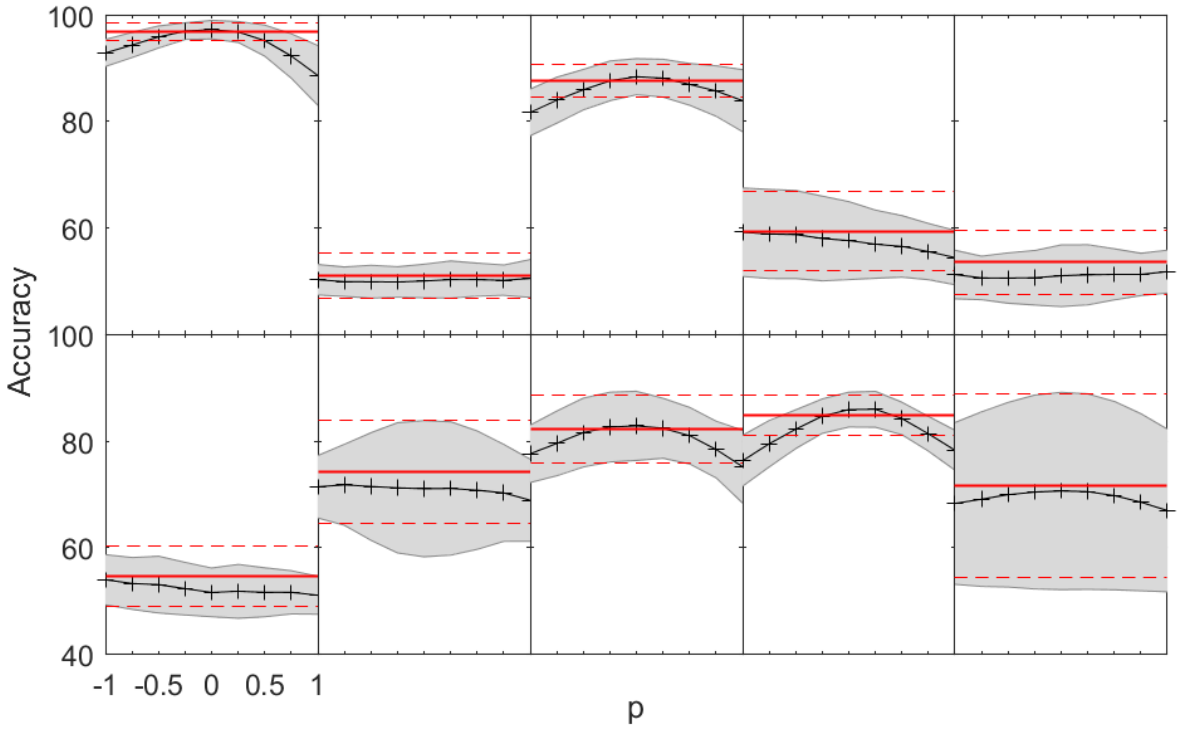


FIGURE .5: Average accuracy of classification for class 2 vs 4 on 9 subjects in motorimagery data for $n = 50$ (576 trials) using power means by MPM and MDM algorithms. The solid line shows the average accuracy over $M = 50$ replication of combination approach bounded by 1 standard deviation by dashed line, and + shows the average accuracy classification over $M = 50$ repeats using several power means with $P = \{\pm 1, \pm 0.75, \pm 0.5, \pm 0.25, 0\}$ bounded by the area of 1 standard deviation. The last plot on the right bottom side shows the average accuracy over all subjects.

	subjects									
	1	2	3	4	5	6	7	8	9	Ave.
Geometric mean	97.24	50.07	88.42	57.66	51.06	51.63	71.08	82.88	85.93	70.66
Best p	95.88	57.60	79.15	65.33	61.12	61.82	75.94	80.99	82.80	73.40
Combination	96.85	51.09	87.66	59.42	53.56	54.65	74.27	82.38	84.86	71.64

TABLE .9: Accuracy of classification with $n = 50$ on 9 subjects class 2 vs 4 motorimagery data using geometric mean, best p and combination approach with $M = 50$.

	Best p	Combination
Geometric mean	Accepted	Accepted
Best p	*	Accepted

TABLE .10: Paired t-test for the accuracy of classification with $n = 50$ on 9 subjects class 2 vs 4 motorimagery data using geometric mean, best p and combination approach with $M = 50$. Each cell shows the decision about null hypothesis which is mean equality of two groups.

Bibliography

Afsari, B., Tron, R. and Vidal, R. (2013) On the convergence of gradient descent for finding the riemannian center of mass. *SIAM Journal on Control and Optimization* **51**(3), 2230–2260.

Ando, T., Li, C.-K. and Mathias, R. (2004) Geometric means. *Linear algebra and its applications* **385**, 305–334.

Arnaudon, M., Barbaresco, F. and Yang, L. (2013) Riemannian medians and means with applications to radar signal processing. *IEEE Journal of Selected Topics in Signal Processing* **7**(4), 595–604.

Arsigny, V., Fillard, P., Pennec, X. and Ayache, N. (2007) Geometric means in a novel vector space structure on symmetric positive-definite matrices. *SIAM journal on matrix analysis and applications* **29**(1), 328–347.

Barachant, A., Bonnet, S., Congedo, M. and Jutten, C. (2010) Common spatial pattern revisited by riemannian geometry. In *Multimedia Signal Processing (MMSP), 2010 IEEE International Workshop on*, pp. 472–476.

Barachant, A., Bonnet, S., Congedo, M. and Jutten, C. (2012) Multiclass brain–computer interface classification by riemannian geometry. *IEEE Transactions on Biomedical Engineering* **59**(4), 920–928.

Barachant, A., Bonnet, S., Congedo, M. and Jutten, C. (2013) Classification of covariance matrices using a riemannian-based kernel for bci applications. *Neurocomputing* **112**, 172–178.

Bhatia, R. (2009) *Positive definite matrices*. Princeton university press.

Bhatia, R. and Holbrook, J. (2006) Riemannian geometry and matrix geometric means. *Linear algebra and its applications* **413**(2-3), 594–618.

Blankertz, B., Tomioka, R., Lemm, S., Kawanabe, M. and Muller, K.-R. (2008) Optimizing spatial filters for robust eeg single-trial analysis. *IEEE Signal Processing Magazine* **25**(1), 41–56.

Borg, I. and Groenen, P. J. (2005) *Modern multidimensional scaling: Theory and applications*. Springer Science & Business Media.

Brants, M., Baeck, A., Wagemans, J. and de Beeck, H. P. O. (2011) Multiple scales of organization for object selectivity in ventral visual cortex. *Neuroimage* **56**(3), 1372–1381.

Chebbi, Z. and Moakher, M. (2012) Means of hermitian positive-definite matrices based on the log-determinant α-divergence function. *Linear Algebra and its Applications* **436**(7), 1872–1889.

Congedo, M. (2013) *EEG source analysis*. Ph.D. thesis, Citeseer.

Congedo, M., Afsari, B., Barachant, A. and Moakher, M. (2015) Approximate joint diagonalization and geometric mean of symmetric positive definite matrices. *PloS one* **10**(4), e0121423.

Congedo, M., Barachant, A. and Andreev, A. (2013) A new generation of brain-computer interface based on riemannian geometry. *arXiv preprint arXiv:1310.8115* .

Congedo, M., Barachant, A. and Koopaei, E. K. (2017) Fixed point algorithms for estimating power means of positive definite matrices. *IEEE Transactions on Signal Processing* **65**(9), 2211–2220.

Congedo, M., Goyat, M., Tarrin, N., Ionescu, G., Varnet, L., Rivet, B., Phlypo, R., Jrad, N., Acquadro, M. and Jutten, C. (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.

Cox, D. D. and Savoy, R. L. (2003) Functional magnetic resonance imaging (fmri)brain reading: detecting and classifying distributed patterns of fmri activity in human visual cortex. *Neuroimage* **19**(2), 261–270.

Crosilla, F. and Beinat, A. (2002) Use of generalised procrustes analysis for the photogrammetric block adjustment by independent models. *ISPRS Journal of Photogrammetry and Remote Sensing* **56**(3), 195–209.

Devrim, M. (2003) Generalized procrustes analysis and its applications in photogrammetry. *prepared for Praktikum in Photogrammetrie, Fernerkundung und GIS, ETH Zuerich* .

Dwyer, P. S. and MacPhail, M. (1948) Symbolic matrix derivatives. *The annals of mathematical statistics* pp. 517–534.

Eckart, C. and Young, G. (1936) The approximation of one matrix by another of lower rank. *Psychometrika* **1**(3), 211–218.

Faraki, M., Harandi, M. T. and Porikli, F. (2015) More about vlad: A leap from euclidean to riemannian manifolds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4951–4960.

Fillard, P., Arsigny, V., Ayache, N. and Pennec, X. (2005) A riemannian framework for the processing of tensor-valued images. In *DSSCV*, pp. 112–123.

Fletcher, P. T. (2013) Geodesic regression and the theory of least squares on riemannian manifolds. *International journal of computer vision* **105**(2), 171–185.

Georgiou, T. T. (2007) Distances and riemannian metrics for spectral density functions. *IEEE Transactions on Signal Processing* **55**(8), 3995–4003.

Goodall, C. (1991) Procrustes methods in the statistical analysis of shape. *Journal of the Royal Statistical Society. Series B (Methodological)* pp. 285–339.

Gorgolewski, K. J., Storkey, A. J., Bastin, M. E., Whittle, I. and Pernet, C. (2013) Single subject fmri test–retest reliability metrics and confounding factors. *Neuroimage* **69**, 231–243.

Gower, J. C. (1975) Generalized procrustes analysis. *Psychometrika* **40**(1), 33–51.

Hastie, T., Tibshirani, R. and Friedman, J. (2008) *The elements of statistical learning*. Volume 2. Springer series in statistics Springer, USA.

Haxby, J. V., Guntupalli, J. S., Connolly, A. C., Halchenko, Y. O., Conroy, B. R., Gobbini, M. I., Hanke, M. and Ramadge, P. J. (2011) A common, high-dimensional model of the representational space in human ventral temporal cortex. *Neuron* **72**(2), 404–416.

Higham, N. J. (1997) Stable iterations for the matrix square root. *Numerical Algorithms* **15**(2), 227–242.

Huettel, S. A., Song, A. W. and McCarthy, G. (2004) *Functional magnetic resonance imaging*. Volume 1. Sinauer Associates Sunderland.

Hung, C. P., Kreiman, G., Poggio, T. and DiCarlo, J. J. (2005) Fast readout of object identity from macaque inferior temporal cortex. *Science* **310**(5749), 863–866.

Jeuris, B., Vandebril, R. and Vandereycken, B. (2012) A survey and comparison of contemporary algorithms for computing the matrix geometric mean. *Electronic Transactions on Numerical Analysis* **39**(EPFL-ARTICLE-197637), 379–402.

Jiang, X., Ning, L. and Georgiou, T. T. (2012) Distances and riemannian metrics for multivariate spectral densities. *IEEE Transactions on Automatic Control* **57**(7), 1723–1735.

Johnson, R. M. (1963) On a theorem stated by eckart and young. *Psychometrika* **28**(3), 259–263.

Kalunga, E. K., Chevallier, S., Barthélemy, Q., Djouani, K., Monacelli, E. and Hamam, Y. (2016) Online ssvep-based bci using riemannian geometry. *Neurocomputing* **191**, 55–68.

Kiani, R., Esteky, H., Mirpour, K. and Tanaka, K. (2007) Object category structure in response patterns of neuronal population in monkey inferior temporal cortex. *Journal of neurophysiology* **97**(6), 4296–4309.

Korczowski, L., Congedo, M. and Jutten, C. (2015) Single-trial classification of multi-user p300-based brain-computer interface using riemannian geometry. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 1769–1772.

Kristof, W. and Wingersky, B. (1971) A generalization of the orthogonal procrustes rotation procedure to more than two matrices. In *Proceedings of the Annual Convention of the American Psychological Association*.

Langleben, D. D. and Moriarty, J. C. (2013) Using brain imaging for lie detection: Where science, law, and policy collide. *Psychology, Public Policy, and Law* **19**(2), 222.

Lawson, J. and Lim, Y. (2013) Weighted means and karcher equations of positive operators. *Proceedings of the National Academy of Sciences* **110**(39), 15626–15632.

Lawson, J. and Lim, Y. (2014) Karcher means and karcher equations of positive definite operators. *Transactions of the American Mathematical Society, Series B* **1**(1), 1–22.

Li, Y. and Wong, K. M. (2013) Riemannian distances for signal classification by power spectral density. *IEEE Journal of Selected Topics in Signal Processing* **7**(4), 655–669.

Li, Y., Wong, K. M. and de Bruin, H. (2012) Electroencephalogram signals classification for sleep-state decision–a riemannian geometry approach. *IET signal processing* **6**(4), 288–299.

Lim, Y. and Pálfia, M. (2012) Matrix power means and the karcher mean. *Journal of Functional Analysis* **262**(4), 1498–1514.

Logothetis, N. K., Pauls, J., Augath, M., Trinath, T. and Oeltermann, A. (2001) Neurophysiological investigation of the basis of the fmri signal. *Nature* **412**(6843), 150–157.

Moakher, M. (2005) A differential geometric approach to the arithmetic and geometric means of operators in some symmetric spaces. *SIAM. J. Matrix Anal. Appl* **26**(3), 735–747.

Moakher, M. (2006) On the averaging of symmetric positive-definite tensors. *Journal of Elasticity* **82**(3), 273–296.

Moakher, M. and Zéraï, M. (2011) The riemannian geometry of the space of positive-definite matrices and its application to the regularization of positive-definite matrix-valued data. *Journal of Mathematical Imaging and Vision* **40**(2), 171–187.

Nakamura, N. (2009) Geometric means of positive operators. *Kyungpook mathematical journal* **49**(1), 167–181.

Pálfia, M. (2016) Operator means of probability measures and generalized karcher equations. *Advances in Mathematics* **289**, 951–1007.

Pfurtscheller, G. and Da Silva, F. L. (1999) Event-related eeg/meg synchronization and desynchronization: basic principles. *Clinical neurophysiology* **110**(11), 1842–1857.

Schönemann, P., Bock, R. and Tucker, L. (1965) Some notes on a theorem by eckart and young. *Research Memorandum* (25).

Schönemann, P. H. (1966) A generalized solution of the orthogonal procrustes problem. *Psychometrika* **31**(1), 1–10.

Skovgaard, L. T. (1984) A riemannian geometry of the multivariate normal model. *Scandinavian Journal of Statistics* **11**(4), 211–223.

Spiridon, M. and Kanwisher, N. (2002) How distributed is visual category information in human occipito-temporal cortex? an fmri study. *Neuron* **35**(6), 1157–1165.

Sra, S. (2016) Positive definite matrices and the s-divergence. *Proceedings of the American Mathematical Society* **144**(7), 2787–2797.

Ten Berge, J. M. (1977) Orthogonal procrustes rotation for two or more matrices. *Psychometrika* **42**(2), 267–276.

Tsao, D. Y., Freiwald, W. A., Tootell, R. B. and Livingstone, M. S. (2006) A cortical region consisting entirely of face-selective cells. *Science* **311**(5761), 670–674.

Zhang, T. (2014) A majorization-minimization algorithm for the karcher mean of positive definite matrices. *arXiv, 1312.4654* .

Zhang, X., Wang, Y., Gou, M., Sznaier, M. and Camps, O. (2016) Efficient temporal sequence comparison and classification using gram matrix embeddings on a riemannian manifold. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4498–4507.

Zou, H. and Hastie, T. (2005) Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **67**(2), 301–320.

Ehsan Kharati Kooapei

CURRICULUM VITAE

Date of Birth: March 25, 1989
Place of Birth: Shiraz, Iran
Nationality: Iranian

Contact Information

University of Padova
Department of Statistics
via Cesare Battisti, 241-243
35121 Padova. Italy.

Tel. +39 3270566709
e-mail: kharati@stat.unipd.it

Current Position

Since November 2014; (expected completion: February 2018)
PhD Student in Statistical Sciences, University of Padova.
Thesis title: Classification Approaches in Neuro Science: A Geometrical Point of View
Supervisor: Prof. Livio Finos
Co-supervisor: Prof. Bruno Scarpa.

Research interests

- Applied Statistics:
  - Big Data Analysis ,Data Mining and Machine learning
    - Neuro science
    - Brain image analysis
    - fMRI data, EEG signals
    - Brain-Computer interface (BCI)
  - GLM, GLMM
  - Nonparametric Statistics
  - Experiments of Design
  - Sampling
- Pure Statistics:
  - Statistical Inference: Hypothesis testing, Maximum likelihoods, Confidence intervals
  - Probability Theory

Education

September 2012 – September 2014
Master of Science, degree in Mathematical Statistics.
University of Shiraz, Faculty of School of Science, Department of Statistics.
Title of dissertation: "Statistical Inference on the Coefficient of Variation of Several Normal Populations: based on Parametric Bootstrap Approach. "
Supervisor: Prof. Soltan Mohammad Sadooghi-Alvandi
Final mark: Excellent (19.30/20)

September 2008 – September 2012
Bachelor of Science, degree in Statistics.

University of Shiraz, Faculty of School of Science, Department of Statistics.
Title of dissertation: Statistical project
Supervisor: Prof. Soltan Mohammad Sadooghi-Alvandi
Final mark: 19.5/20

Visiting periods

April 2015 – June 2015
Name of Institution: GIPSA-lab, Grenoble Images Speech Signal and Control, a joint research unit of CNRS and University of Grenoble,
Grenoble, France.
Supervisor: Prof. Marco Congedo.

Awards and Scholarship

November 2014
PhD scholarship for three years from University of Padova.

Computer skills

- R
- Latex
- MATLAB
- SPSS
- Minitab
- C standard
- MS Office (MS Word, MS Math type, MS PowerPoint)

Language skills

Persian (Farsi): native; English: fluent; Italian: basic.

Publications

Articles in journals

Kharati Koopaei, Ehsan, Sadooghi Alvandi, Soltan Mohammad, (2014). Testing equality of coefficients of variation of several normal populations: with parametric bootstrap method. *Journal of Statistical Sciences* **8 (1)**, 37–56.

Eftekhar, Sana, Kharati Koopaei, Ehsan Sadooghi Alvandi, Soltan Mohammad, (2015). Confidence intervals for the ratio and difference of two C_pmk idices based on parametric bootstrap and asymptotic approaches. *Journal of Statistical Sciences* **9 (2)**, 169–188.

Kharrati-koopaei, Mahmood, Kharati Koopaei, Ehsan, (2016). A note on the multiple comparisons of exponential location parameters with several controls under heteroscedasticity. *Hacettepe University Bulletin of Natural Sciences and Engineering Series B: Mathematics and Statistics* **46 (127)**, 1– 1.

Congedo, Marco, Barachant, Alexandre, Kharati Koopaei, Ehsan, (2017). Fixed point algorithms

for estimating power means of positive definite matrices. *IEEE Transactions on Signal Processing* **65 (9)**, 2211 – 2220.

Working papers
Kharati Kooapei, Ehsan, Finos, Livio, Scarpa,Bruno (2017). Classification and prediction on fMRI data.

Kharati Kooapei, Ehsan, Finos, Livio (2017). Statistical combinations of power means: classification study on functional data.

Conference presentations

Kharati Koopaei,Ehsan (2014). Statistical inference on the coefficient of variation of several normal populations: Based on Parametric Bootstrap approach. *The 12 Iraian Statistical Conference* , Razi University, Kermanshah, Iran.

Kharati Koopaei,Ehsan, Eftekhar, Sana, Sadooghi Alvandi, Soltan Mohammad (2015). Aymptotic parametric and bootstrap confidence intervals for the ratio and difference of two indices C-pmk. *The 10th Seminar on Probblity and Stochastic Process* , Yazd University, Yazd, Iran.

Teaching experience

April 2013 – June 2013
Course name: Bio-Statistics
Degree: Master of Biology
Teaching task: lab, 14 hours
Institution: Department of Statistics, College of Science, Shiraz University
Instructor: Prof. Alireza Nematollahi

September 2013 – June 2014
Course name: Mathematical Statistics (1,2)
Degree: Bachelor of Statistics
Teaching task: exercises, 3 semesters each semester 2 hours per week
Institution: Department of Statistics, College of Science, Shiraz University
Instructor: Prof. Rasool Borhani Haghighi

October 2012 – March 2013
Course name: Multivariate Analysis
Degree: Bachelor of Statistics
Teaching task: lab, 8 hours
Institution: Department of Statistics, College of Science, Shiraz University
Instructor: Prof. Alireza Nematollahi

References

Prof. Mohammad Sadooghi-Alvandi
Institution: Department of Statistics, Shiraz University
Address: Shiraz, Iran
Phone: -
e-mail: smsa51@hotmail.com

Prof. Bruno Scarpa
Institution: Department of Statistical Sciences, University of Padova
Address: Padova, 35121, Italy
Phone: -
e-mail: bruno.scarpa@unipd.it