Chapter 5

Software: EEG tool

5.1 Introduction

We developed a user friendly software named "EEG tool" allowing to use all the methods presented in the previous chapters. This software is a graphical user interface (GUI) with point-and-click features developed using GUIDE, the MATLAB Graphical User Interface Development Environment.

5.2 How to run EEG tool

To use the EEG tool, you must first save in a single directory the files from the CD provided with this text and follow instructions in file "readme.txt". This is a standalone C version of the Matlab GUI. Consequently, you don't need to install Maltab. To run the application, you simply launch the file: `eegtool.exe`. The main window will then appear (Figure 5.1). The main window allow the user to access 3 different menus. These menus will be detailed in the following sections.

Figure 5.1: Main window of the EEG tool.

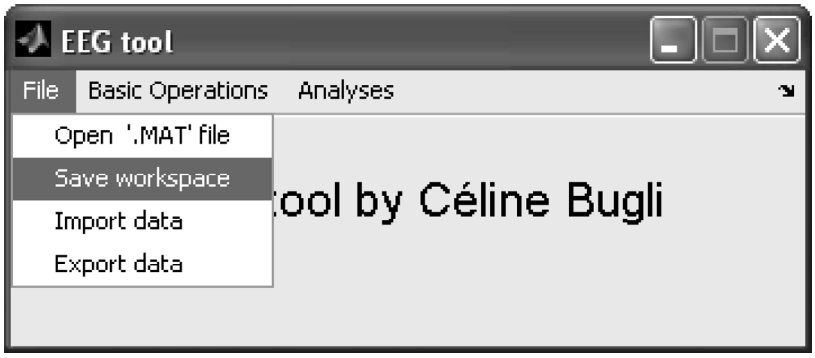

Figure 5.2: Main window of the EEG tool.

5.3 File menu

This menu enables to manage data by importing and exporting data files, opening Matlab '.MAT' datasets and saving the current workspace (Figure 5.2).

The Open '.MAT' action allows to open a Matlab dataset (Figure 5.3). The button "Open file" opens a browse window to select the file (Figure 5.4). Moreover, a previsualization of the available variables allows to extract only the needed variables. By clicking on the button "Export selected variables to workspace", you can either select all variables (button "Select all variables") or only some of them.

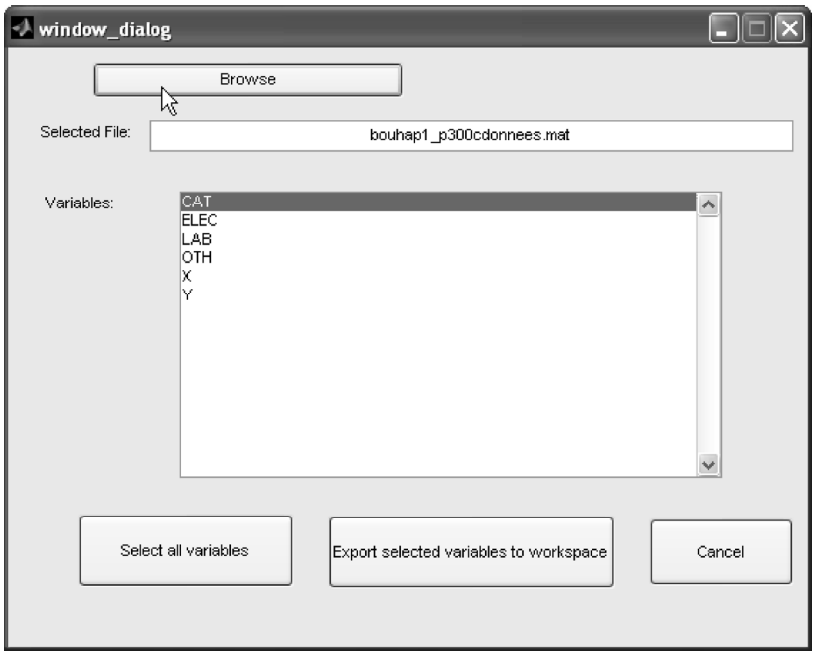

Figure 5.3: 'Open .MAT dialog window.

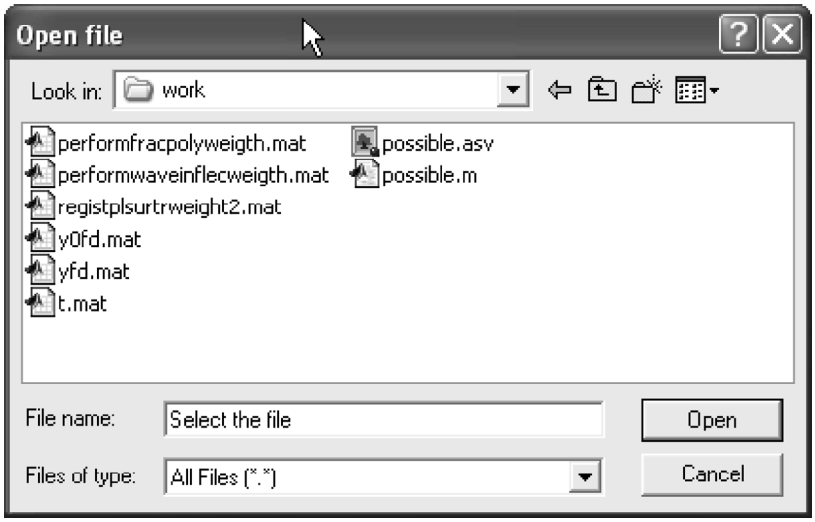

Figure 5.4: Browse window.

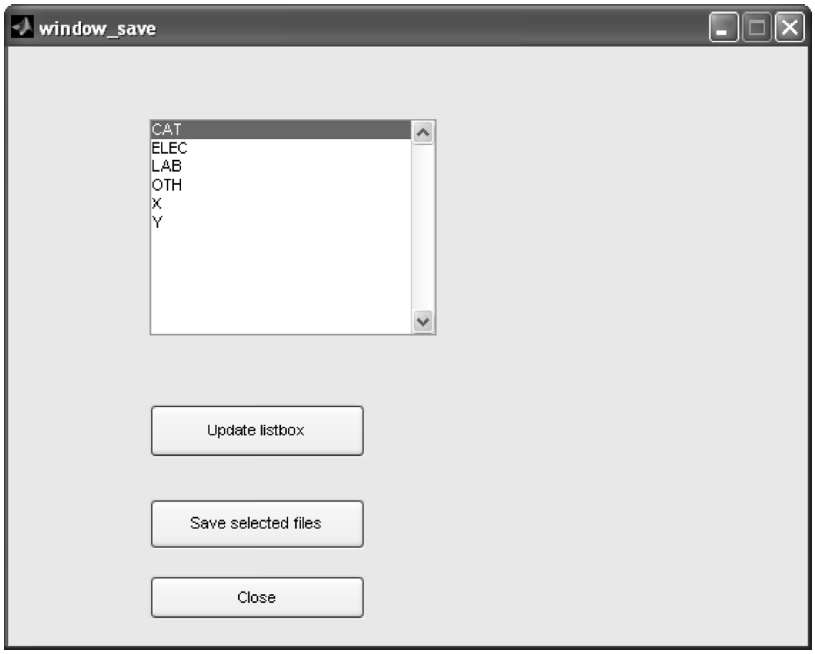

Figure 5.5: 'Save workspace' dialog window.

The 'Save workspace' action enables to select variables from the workspace to save into a '.MAT' file (Figure 5.5). The 'Update listbox' button refresh the list of available variables in the workspace. The 'Save selected variables' button opens a browse window to save the file in the appropriate folder.

The 'Import data' action allow to import a 'CNT' EEG file and create a Matlab dataset '.MAT' file (Figure 5.6) saved in the same directory. The name of this dataset is the same as the name of the '.CNT' file plus the extension 'donnees.mat'.

The 'Export data' action creates an ASCII text '.txt' file from a Matlab dataset '.MAT' file containing a matrix (Figure 5.7). The limitation of this item is that we can only export one matrix and not several variables.

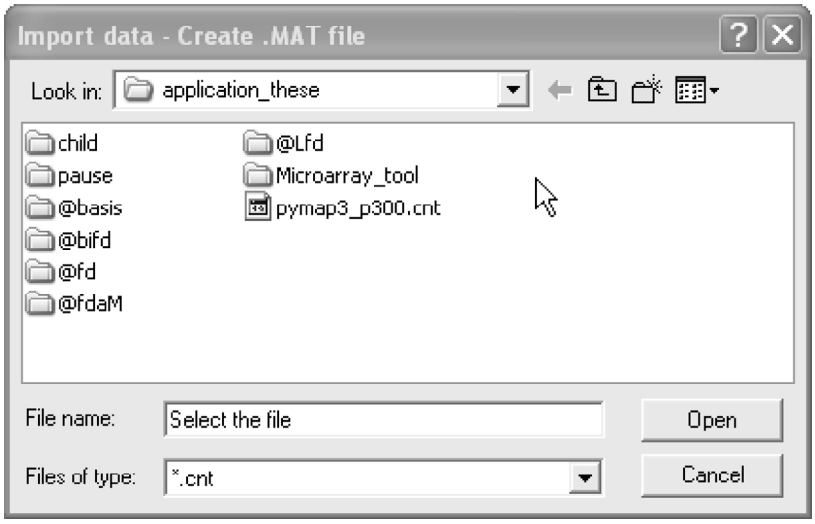

Figure 5.6: 'Import data' dialog window.

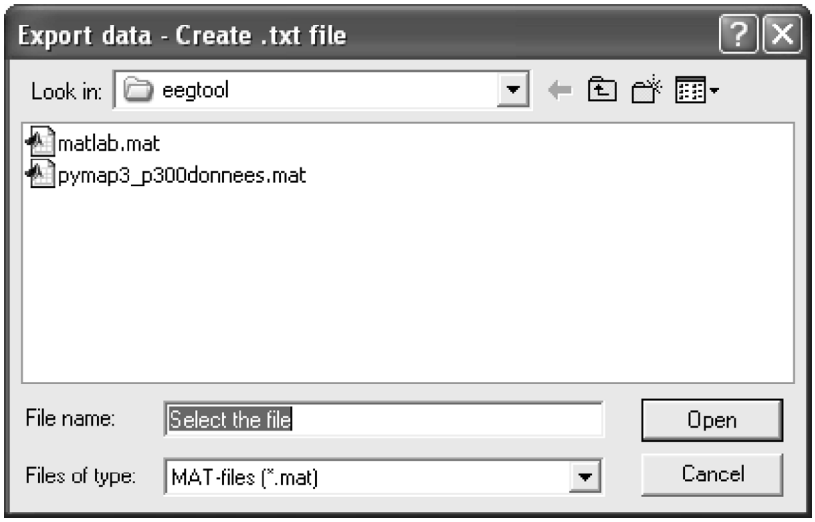

Figure 5.7: 'Export data' dialog window.

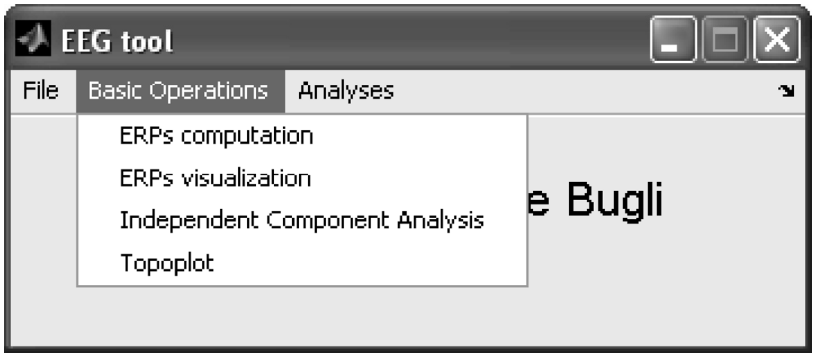

Figure 5.8: 'Basic Operations' menu.

5.4 Basic Operations menu

This menu enables basic computations on EEG data like ERP's computation and visualization, computation and visualisation of ICA components with a topographic map named topoplot as explained in Chapter 4 (Figure 5.8).

The 'ERPs computation' item averages EEG signals to compute the ERP curves as explained in Chapter 2 (Figure 5.9). The 'Update list' button refresh the list of available variables. You can choose the matrix of data by clicking on the variable. The 'Select data' button confirm the selection of the matrix of EEG data. The 'Select stimuli file' button opens a browse window to select a '.EV2' file which contains definition of the auditory stimuli. This file is normally provided with the '.CNT' data files. You can change parameters in the input windows 'Matrix of results name' and 'Interval length'.

The 'ERPs visualization' item generates a plot of the ERP curves (Figure 5.10). The 'Rows to plot' input box enables to choose the rows of the ERP matrix to drawn. Click 'All' in the input box if you want all the ERP curves. Type the number of the rows to plot separated by a comma if you want to select only some ERP curves. Each ERP curve will be plotted into a separated figure window (Figure 5.11).

The 'Independent Component Analysis' item computes independent components

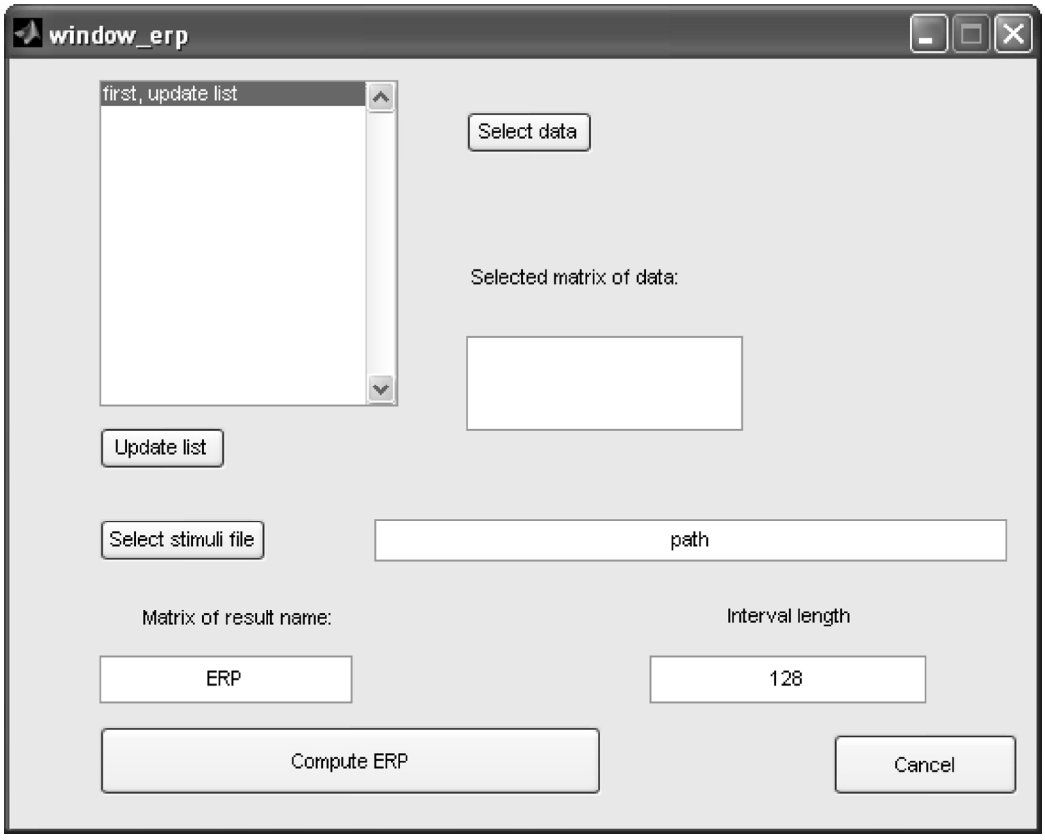

Figure 5.9: 'ERPs computation' dialog window.

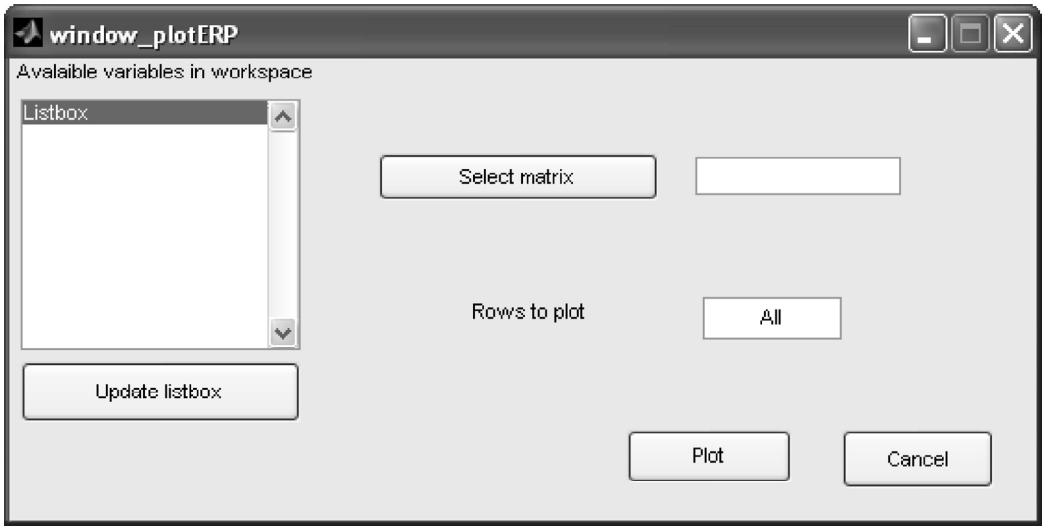

Figure 5.10: 'ERPs visualization' dialog window.

using the JADE algorithm as used in Chapter 4 (Figure 5.12). This algorithm is explained in Appendix A. You can choose the parameters by filling The 'Number of independent components' input box. You can change the name of the mixing and unmixing matrix in the corresponding input boxes. You have to check or uncheck the check boxes to select if you want to export the mixing and unmixing matrix to the workspace. Note that you only need the unmixing matrix to compute the ICA components in the workspace but you need the mixing matrix to use the topoplot item.

The 'Topoplot' action plots the contribution of ICA components with a topographic map named topoplot as explained in Chapter 4 (Figure 5.13). Figure 5.14 shows an example of topoplot.

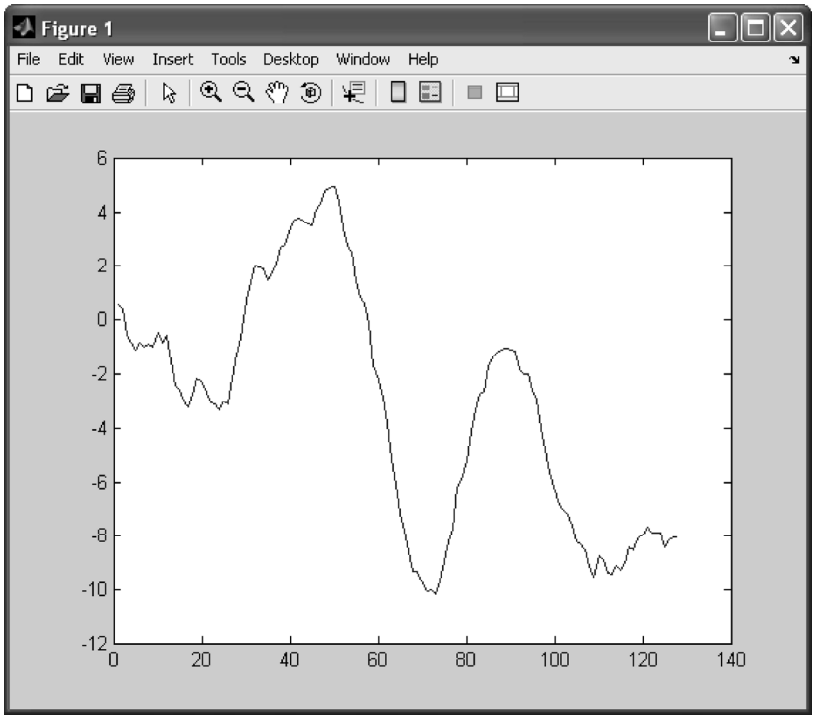

Figure 5.11: An example of ERP.

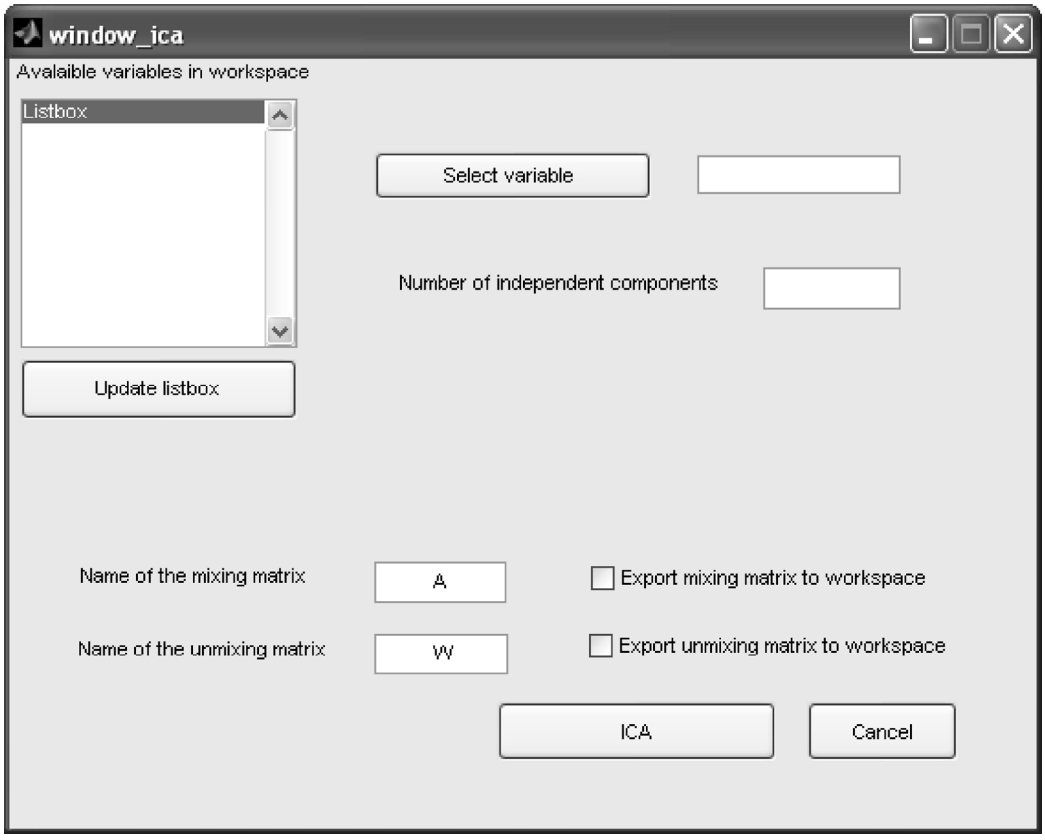

Figure 5.12: 'Independent Component Analysis' dialog window.

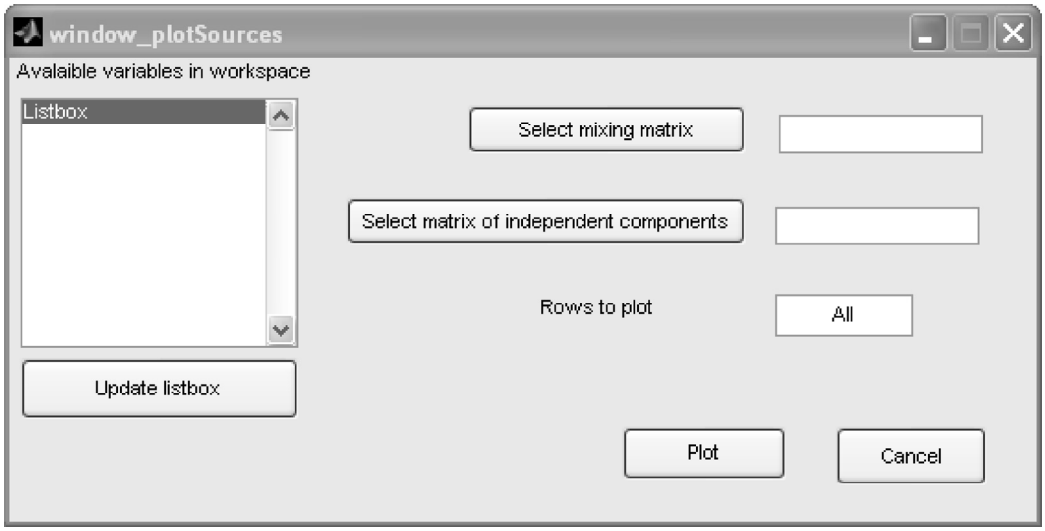

Figure 5.13: 'Topoplot' dialog window.

5.5 Analyses menu

This menu can be used to run the analyses presented in this thesis: the cleaning of EEG data presented in Chapter 4, the functional ANOVA presented in Chapter 5 and the registration by fractional polynomials presented in Chapter 3 (Figure 5.15).

The 'Cleaning' item enables to clean EEG data using the 2-steps methodology explained in Chapter 3 (Figure 5.16). Check boxes allow to use artifactual electrodes or not and to use the 2-steps method or one of these steps. Note that if you check the check box '2nd step' but not the check box 'Use artifactual electrodes (1st step)', only the second step of cleaning will be used. You can define the number of the artifactual electrodes, the thresholds for the correlation and for the kurtosis in input boxes. You have to fill in a input box to define the name of the matrix with cleaned data.

The 'Functional ANOVA' item computes the functional ANOVA modelling with P-splines as explained in Chapter 5 (Figure 5.17). You can define if the time scale is the same for each curve ('Same time scale for each curve' check box) or use a time

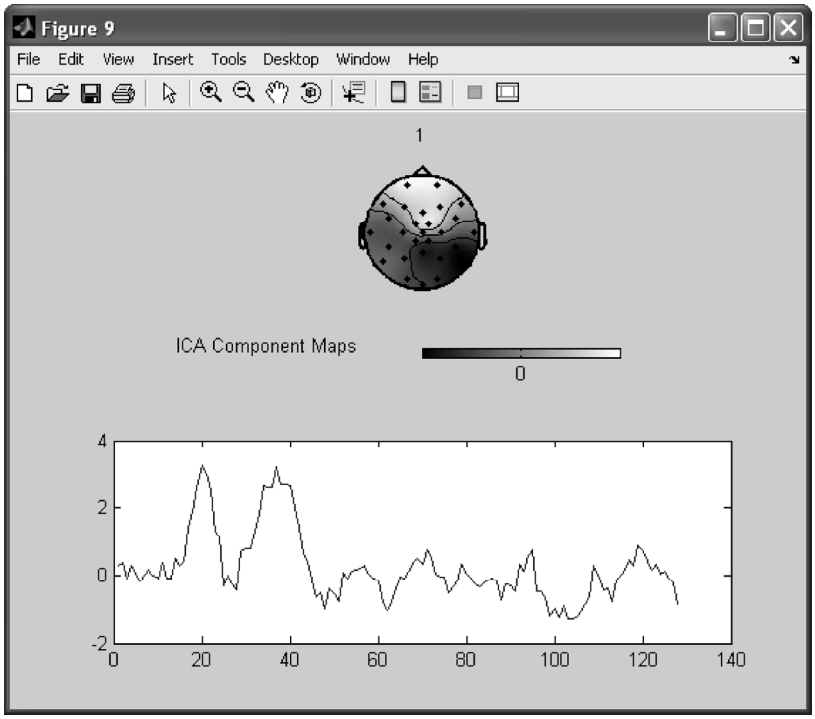

Figure 5.14: Example of topoplot.

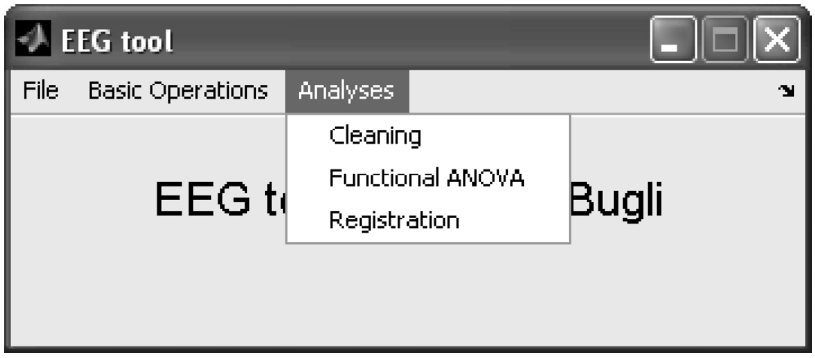

Figure 5.15: Analyses Menu.

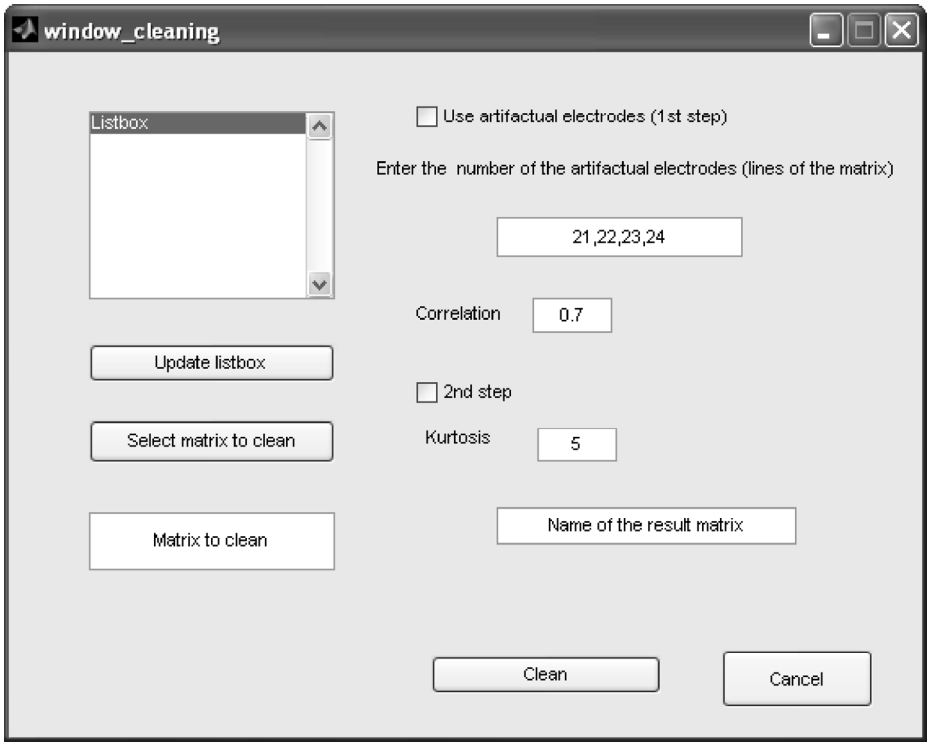

Figure 5.16: 'Cleaning' dialog window.

matrix defining the time scale for each curve separately ('Time matrix' input box). When the time scale is the same for each curve, you can change the values of the input boxes 'Start point', 'End point' and 'number of points' to define another time scale. You can create the design matrix of your model by selecting fixed and/or random variables in the workspace and finally clicking the 'Create design matrix' button. A check box allow to plot the effect curves with the confidence bands (an example is presented in Figure 5.19). You can change the parameters of the P-spline basis in the input boxes 'Number of knots', 'Degree', Penalty order'. When you push the 'Compute model' button, a dialog box opens to choose the smoothing parameter for each effect curve. You can directly enter the value in the input box, or use the slider (Figure 5.18).

The 'Registration' item aligns curves using the fractional polynomial registration presented in Chapter 3 (Figure 5.20). You can align the curves on their mean (check box 'Target function is the mean of the curves to register') or on a selected target from the workspace.

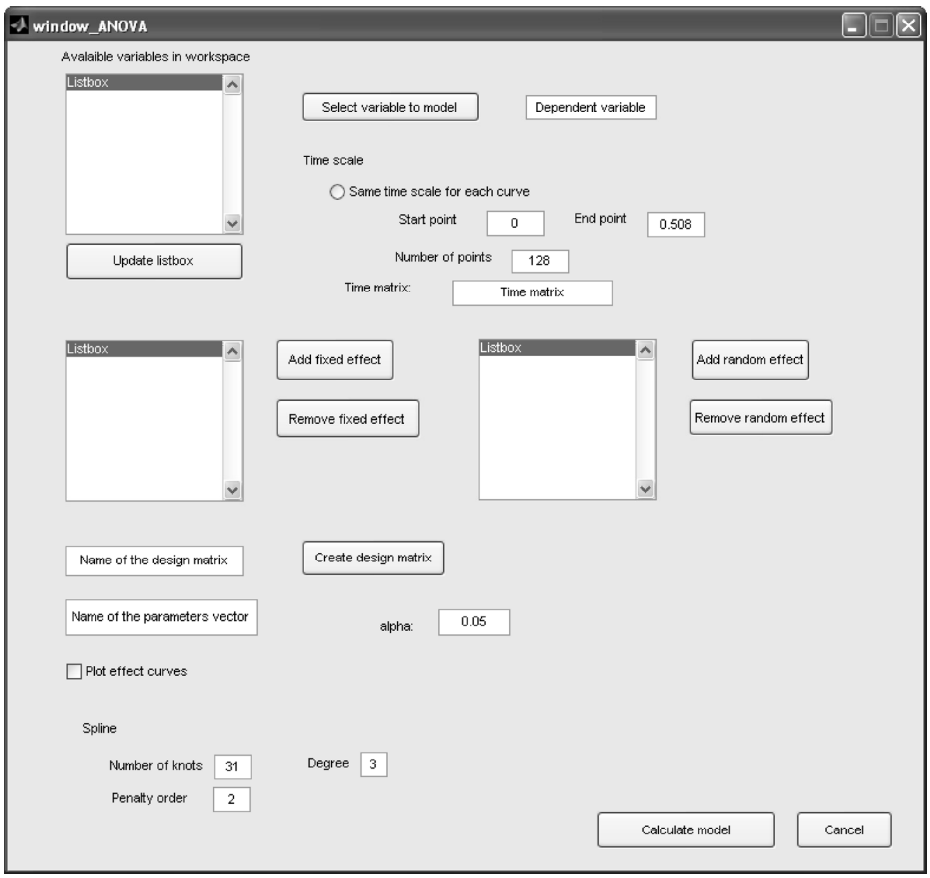

Figure 5.17: 'Functional ANOVA' dialog window.

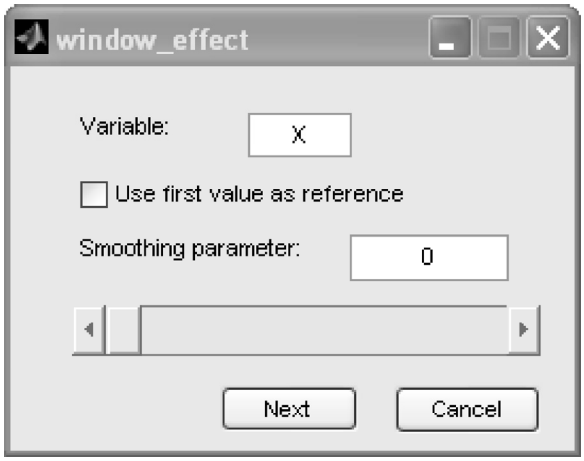

Figure 5.18: 'Smoothing parameter' dialog window.

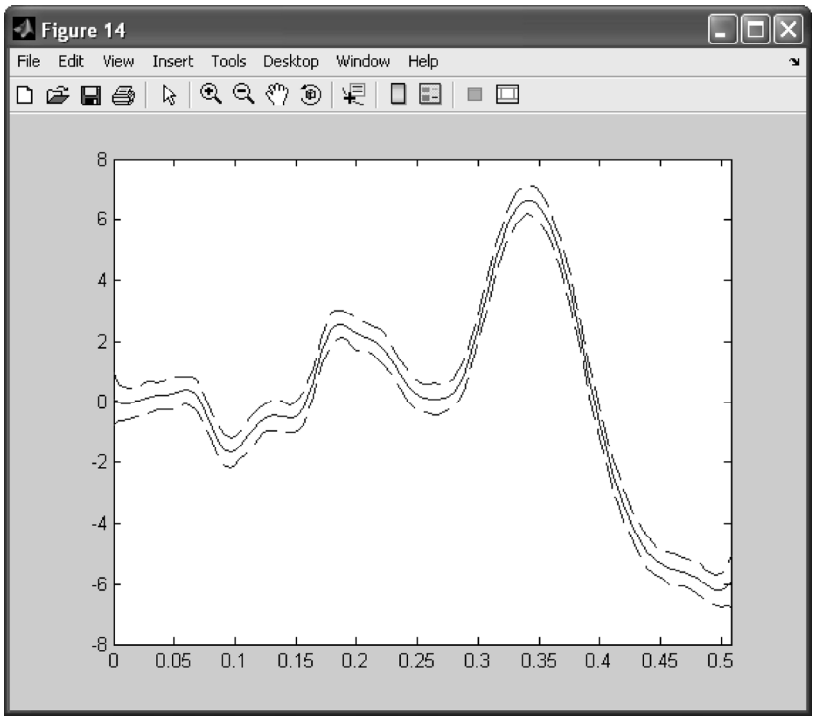

Figure 5.19: Example of effect curve with confidence bands.

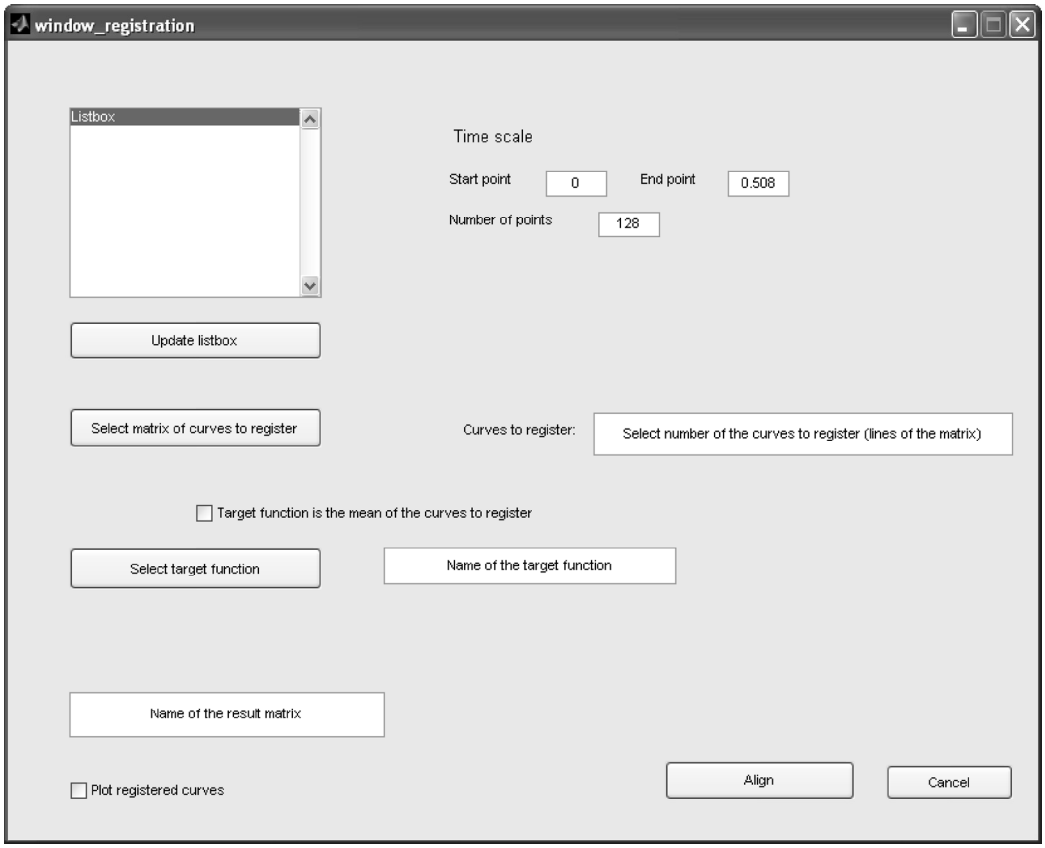

Figure 5.20: 'Registration' dialog window.

Conclusion and future research

In this thesis, we have proposed several techniques specifically dedicated to the analysis of electroencephalograms. These techniques, presented in the EEG analysis framework, are also relevant whenever functional data are obtained. The application of these techniques to other applications could be an interesting future work[1]. Moreover, it could also be interesting for other application to integrate the different tools. For example, functional ANOVA of independent component could improve results compared with functional ANOVA of signals from electrodes.

The first topic presented was the use of ICA for EEG analysis. Our main contributions in this topic are to make the cleaning process of EEG completely automatic, a proposition of an efficient method for dimensionality reduction and an important insight in the understanding of treatment effect on the brain thanks to the identification and analysis of 2 interesting ICA components. Moreover, we proposed a method allowing to compare independent sources computed from several datasets. In the future, it could be interesting to test other ICA algorithm than the JADE algorithm. Methods of ICA dealing with autocorrelation and non stationary variance exist. However, the different criteria make different assumption on the data, and, hence, the choice of the criterion is problem specific. When the data have no time structure, the basic ICA method can be used. If the data have clear time-dependencies, these

[1] We already applied functional ANOVA in bioengineering for the analysis of microarray data. See joint work with L. Eyers, Functional ANOVA with P-splines: an application to the analysis of melting curves of nucleic acids (submitted).

usually imply nonzero (positive) autocorrelation functions, and the methods based on autocorrelations can be used. However, such methods work only if the autocorrelations are different for each independent component (because the algorithm need to find distinct eigenvalues of the covariance matrix, see e.g. Tong et al., 1991). When some of the autocorrelations are identical, one could try to use the method based on non stationarity (see e.g. Matsuaka et al., 1995), since these methods utilize the time structure but do not require the time structure of each independent component to be different. On the other hand, the basic ICA method often works well even if the target independent components have time-dependencies. The basic methods do not use the time structure, but this does not mean that they would be disturbed by such a structure (Hyvärinen, Karhunen and Oja, 2001, chapter 18). Consequently, we decided to use the basic ICA method in this thesis. However, comparison with methods using autocorrelation could be very interesting for future work.

The second approach presented was a new parametric method of curve alignment using fractional polynomials. This method of registration has the main advantage to be very simple to compute and to be very flexible for the range of applications of EEG analysis. Although the fractional polynomial technique is very flexible, it could be interesting to find another parametric model with neurophysiological meaningful parameters to align placebo and treatment ERP curves. This task would imply the identification of relevant biologic parameter. Another interesting extension would be the computation of confidence bands for the warping function which would allow to identify significant changes between placebo and treatment curves. Bootstrap or Bayesian methods could be used. Regarding the huge quantity of data, it would be time consuming for our application.

The last topic we contributed was the implementation of a functional ANOVA method using P-splines (Bugli and Lambert, 2006). Functional ANOVA with P-splines is a powerful tool to model curves. Indeed, the model can be easily and clearly stated. In the field of EEG analysis, it allows to bring information about the modification of shape of the peak and not only about some specific characteristics like

amplitude or latency of one particular peak that are difficult to identify from noisy EEG signals. Functional ANOVA using splines is already used to model curves with individual behaviour. For example, Durban et al. (2005) suggested individual curves modelled as penalized splines with random intercept. The originality of our work is that the random effect is a curve instead of a random intercept. Future work could be the improvement of selection of the smoothing parameters. A Bayesian setup is a possible solution to take into account efficiently the large uncertainty of smoothing parameters in the computation of the confidence intervals (see e.g. Lambert and Eilers, 2005).

All procedures presented in this thesis have been implemented in Matlab. We provide a friendly user interface to use all our contributions.

Consequently, we have a tool that can give, at the early stage of clinical testing, indications of treatment effect usually observed in the long run. The derived biomarkers can help to gain time during the development of a new treatment with impact on the cognitive functions while, at the time, giving a better understanding of the mechanisms involved by cerebral activity.

Bibliography

[1] Baudena P, Halgren E, Heit G, Clarke JM, Intracerebral potentials to rare target and distractor auditory and visual stimuli. III. Frontal cortex. *Electroencephalography and Clinical Neurophysiology* 1995; **94**: 251–264.

[2] Bigot J. Landmark-based registration of 1D curves and functional analysis of variance with wavelets. *Technical Report TR0333, PAI (Interuniversity Attraction Pole network)* 2003.

[3] Bigot J. A scale-space approach with wavelets to singularity estimation, to appear in *ESAIM: PS* 2005.

[4] Brown H, Prescott R. *Applied Mixed Models in Medicine.* Wiley: Chichester, 2003.

[5] Brumback B, Rice J. Smoothing spline models for the analysis of nested and crossed samples of curves, *Journal of the American Statistical Association* 1998; **93**: 961–976.

[6] Bugli C, Lambert P, Boulanger B, Ledent E, Pereira A,Nardone P. Statistical analysis of electroencephalograms: independent component analysis of event-related potentials, *Discussion paper number* DP403 (2004), Institut de Statistique, Université catholique de Louvain, B-1348 Louvain-la-Neuve, Belgium.

[7] Bugli C, Lambert P, Bigot J. Curve registration using fractional polynomials with application to electroencephalograms analysis, *Discussion paper number* DP0513

(2005), Institut de Statistique, Université catholique de Louvain, B-1348 Louvain-la-Neuve, Belgium.

[8] Bugli C, Lambert P. Functional ANOVA with random functional effects: an application to event-related potentials modelling for electroencephalograms analysis, to appear in Statistics in Medicine (2006)

[9] Cardoso JF, Souloumiac A. Blind Beamforming for non Gaussian Signal., *IEE-Proceedings-F* 1993; **140**(6): 362–370.

[10] Comon P. Independent Component Analysis, a New Concept?, *Signal Processing* 1994; **36**: 287–314.

[11] Currie I, Durban M, Eilers P. Generalized linear array models with applications to multidimensional smoothing *Journal of the Royal Statistical Society, Series B* 2006; **68**: 259–280.

[12] de Boor C. *A practical guide to splines*, Springer, New York, 2001.

[13] Delorme A, Makeig S, Sejnowski T. Automatic Artifact Rejection for EEG Data Using High-order Statistics and Independent Component Analysis. *Proceedings of the 3rd International Independent Component Analysis and Blind Source Decomposition Conference*, December 2001, San Diego (USA).

[14] Dierckx P. *Curve and surface fitting with splines*, Clarendon, Oxford, 1995.

[15] Durban M, Harezlak J, Wand MP, Carroll RJ. Simple fitting of subject-specific curves for longitudinal data, *Statistics in Medicine* 2005; **24**: 1153–1167.

[16] Eilers P, Marx B. Flexible smoothing with B-splines and penalties (with discussion). *Statistical Science* 1996; **11**: 89–121.

[17] Eilers P. Curve and spline analysis of variance, *Statistical modeling: Proceedings of the 14th international workshop on statistical modeling* 1999.

[18] Eilers P. Parametric time warping, *Analytical Chemistry* 2004;**76**: 404–411.

[19] Eilers P, Currie I, Durban M. Fast and compact smoothing on large multidimensional grids Computational *Statistics and Data Analysis* 2006; **50**, 61–76.

[20] Eubank RL. *Spline smoothing and nonparametric regression*, Marcel Dekker, New York, 1988.

[21] Fan J, Gijbels I. *Local polynomial modelling and its applications*, Chapman and Hall, London, 1996.

[22] Friedman J. Multivariate adaptive regression splines, *Annals of Statistics* 1991; **19**(1): 1–141.

[23] Golub G, Health M, Wabha G. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics* 1979; **21**:215–223.

[24] Green P, Silverman BW. *Nonparametric Regression and Generalized Linear Models*, Chapman and Hall, London, 1994.

[25] Härdle W. *Smoothing Techniques With Implementation in S*, Springer-Verlag, New York, 1990.

[26] Hastie T, Tibshirani R. *Generalized Additive Models for Medical Research*, Chapman and Hall, London, 1991.

[27] Herault J, Jutten J. Blind Separation of Sources, Part I: An Adaptative Algorithm Based on Neuromimetic architecture, *Signal Processing* 1991; **24**: 1–10.

[28] Herrmann CS, Knight RT. Mechanisms of human attention : event-related potentials and oscillations. *Neuroscience and Biobehavioral Reviews* 2001; **25**: 465–476.

[29] Huber P. Projection pursuit. *The Annals of Statistics* 1985; **13(2)**: 435–475.

[30] Hyvärinen A, Karhunen J, Oja E. *Independent Component Analysis*, Wiley, New York, 2001.

[31] Jasper HH. The 10-20 system of the international federation. *Electroencephalography and Clinical Neurophysiology* 1958; **10**: 371–375.

[32] Jung TP, Makeig S, Humphries C, Lee TW, McKeown MJ, Iragui V, Sejnowski TJ. Removing Electroencephalographic Artifacts by Blind Source Separation. *Psychology* 2000; **37**(2): 163–178.

[33] Kneip A, Gasser T. Statistical tools to analyze data representing a sample of curves. *The annals of Statistics* 1992; **20**(3): 1266–1305.

[34] Lagarias JC, Reeds J., Wright M, Wright P. Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions, *SIAM Journal of Optimisation* 1998; **9(1)**: 112–147.

[35] Lambert P, Eilers P.H.C. Bayesian proportional hazards model with time varying regression coefficients: a penalized Poisson regression approach, *Statistics in Medicine* 2005; **24**: 3977–3989.

[36] Makeig S, Bell AJ, Jung TP, Sejnowski TJ. Independent Component Analysis of Electroencephalographic Data, *Advances in Neural Information Processing Systems* 1996; 145–151.

[37] Matsuoka K, Ohya M, Kawamoto M. A neural net for blind separation of nonstationary signals, *Neural Networks* 1995; **8(3)**:411–419.

[38] O'Sullivan F. A statistical perspective on ill-posed inverse problems. *Statistical Science* 1986; **1**: 505–527.

[39] O'Sullivan F. Fast computation of fully automated log-density and log-hazard estimators. *SIAM Journal on scientific and statistical computing* 1988; **9**: 363–379.

[40] Polich J, Kok A. Cognitive and Biological Determinants of P300 : an Integrative Review. *Biological Psychology* 1995; **41**: 103–146.

[41] Quian Quiroga R. Quantitative analysis of EEG signals: Time-frequency methods and Chaos theory,*Ph.D. Thesis; Lbeck* 1998.

[42] Ramsay J. Estimating smooth monotone functions. *Journal of the Royal Statistical Society, series B* 1998; **60**: 365–375.

[43] Ramsay J, Li X. Curve registration. *Journal of the Royal Statistical Society, series B* 1998; **60**: 351–363.

[44] Ramsay J, Silverman B. *Functionl data analysis.* Springer-Verlag, New York, 1997.

[45] Rice J, Wu C. Nonparametric mixed effects models for unequally sampled noisy curves, *Biometrics* 2001; **57**: 253–259.

[46] Royston P, Altman D. Approximating statistical functions by using fractional polynomial regression. *The statistician* 1997; **46**(3): 411–422.

[47] Ruppert D, Wand MP, Carroll R, *Semiparametric regression*, Cambridge University Press, New York, 2003.

[48] Schumaker L. *Spline Functions: Basic Theory*, Wiley, New-York, 1981.

[49] Searle S.*Linear Models*, Wiley, New-York, 1971.

[50] Silverman B. *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, London, 1986.

[51] Silverman B. Incorporating parametric effects into functional principal component analysis. *Journal of the Royal Statistical Society, series B* 1995; **57**: 673–689.

[52] Stone C,Hansen MH, Kooperberg C, Truong Y. Polynomial splines and their tensor products in extended linear modeling, *Annals of Statistics* 1997; **25**: 1371–1470.

[53] Tarter ME, Lock MD. *Model-Free Curve Estimation*, Chapman and Hall, New-York, 1993.

[54] Tong L, Liu R-W, Soon VC, Huang Y-F. Indeterminacy and identifiability of blind identification, *IEEE Trans. on Circuits and Systems* 1991; **38**: 499–509.

[55] Tuddenham R, Snyder M. Physical growth of California boys and girls from birth to age 18. *Calif. Publ. Child Develop.* 1954; **1**: 183–364.

[56] Van Loan C. Computational Frameworks for the Fast Fourier Transform. *Philadelphia : SIAM* 1992.

[57] Wahba G. *Spline Models for Observation Data*, SIAM, 1990.

[58] Wand MP, Jones MC. *Kernel Smoothing*, Chapman and Hall, New-York, 1995.

[59] Wand MP. Smoothing and mixed models, *Computational Statistics* 2003; **18**:223–249.

Appendix

Appendix A: Jade algorithm

The JADE algorithm was presented in Cardoso and Souloumiac (1993). This algorithm uses an approximation of the mutual information based on the joint diagonalization of cross-cumulants.

We first introduce the concepts of cumulants and cumulant matrix. Next, we explain the use of the cumulant matrix in the JADE algorithm.

Assume that $\mathcal{X}$ is a random variable with probability density function $f(x)$. Let $\phi(t)$ be the characteristic function of $\mathcal{X}$:

$$\phi(t) = \int_{-\infty}^{+\infty} e^{jtx} f(x) dx$$

The cumulants κ_k are defined as the coefficients of the expansion of $\ln(\phi(t))$:

$$\ln(\phi(t)) = \sum_{k=0}^{\infty} \kappa_k \frac{(jt)^k}{k!}.$$

In terms of moments, the cumulant of order k are defined by:

$$\kappa_k = \mu_k - \sum_{i=0}^{k-1} \mu_i \kappa_{k-1}$$

where μ_i is ith moment of $\mathcal{X}$. For example,

$$\kappa_1 = \mu_1 = E[\mathcal{X}],$$

is the mean of the random variable $\mathcal{X}$,

$$\kappa_2 = \mu_2 - \mu_1\kappa_1 = E[\mathcal{X}^2] - E[\mathcal{X}]E[\mathcal{X}] = \sigma^2,$$

is the variance of the random variable $\mathcal{X}$ and

$$\begin{aligned}\kappa_4 = \quad & E[\mathcal{X}^4] - 3\left(E[\mathcal{X}^2]\right)^2 - 4E[\mathcal{X}^3]E[\mathcal{X}] \\ & +12E[\mathcal{X}^2]\left(E[\mathcal{X}]\right)^2 - 6\left(E[\mathcal{X}]\right)^4\end{aligned}$$

is the kurtosis of $\mathcal{X}$. Suppose that we want to compute the cumulantof of the fourth order for 4 random variables $\mathcal{X}_i, \mathcal{X}_j, \mathcal{X}_k, \mathcal{X}_l$. A more practical notation of κ_4 is then $Cum[\mathcal{X}_i, \mathcal{X}_j, \mathcal{X}_k, \mathcal{X}_l]$.

More practically, if we define $\epsilon[.]$ as

$$\epsilon[\mathcal{X}] = \mathcal{X} - E[\mathcal{X}]$$

for a random variable $\mathcal{X}$, we can write the cumulants of the fourth order as:

$$\begin{aligned}Cum[\mathcal{X}_i, \mathcal{X}_j, \mathcal{X}_k, \mathcal{X}_l] \quad & = E[\epsilon(X_i)\epsilon(\mathcal{X}_j)\epsilon(\mathcal{X}_k)\epsilon(\mathcal{X}_l)] \\ & -E[\epsilon(\mathcal{X}_i)\epsilon(\mathcal{X}_j)]E[\epsilon(\mathcal{X}_k)\epsilon(\mathcal{X}_l)] \\ & -E[\epsilon(\mathcal{X}_i)\epsilon(\mathcal{X}_k)]E[\epsilon(\mathcal{X}_j)\epsilon(\mathcal{X}_l)] \\ & -E[\epsilon(\mathcal{X}_i)\epsilon(\mathcal{X}_l)]E[\epsilon(\mathcal{X}_j)\epsilon(\mathcal{X}_k)].\end{aligned}$$

for the random variables $\mathcal{X}_i, \mathcal{X}_j, \mathcal{X}_k, \mathcal{X}_l$.
Now, we shall introduce the notion of cumulant matrix. The cumulant matrix associated to a random $n \times 1$ vector $\mathcal{X}$ and any arbitrary $n \times n$ matrix M is defined component-wise by:

$$[Q^{\mathcal{X}}(M)]_{ij} = \sum_{k,l=1}^{n} Cum[\mathcal{X}_i, \mathcal{X}_j, \mathcal{X}_k, \mathcal{X}_l] M_{kl}.$$

We shall now use these notions to explain the JADE algorithm.

In the ICA problem, we have

$$\boldsymbol{x}(t) = A\boldsymbol{s}(t)$$

for $t \in \mathcal{T}$. Thus,

$$x_{ij} = \sum_k a_{ik} s_{kj}$$

and so

$$\boldsymbol{x}_i^T = \sum_k a_{ik} \boldsymbol{s}_k^T,$$

where $\boldsymbol{x_i}^T$ is the ith line of the matrix $\mathcal{X}$ and $\boldsymbol{s_k}^T$ is the kth line of the matrix S. One can see that

$$\begin{aligned} Cum[\boldsymbol{x_i}^T, \boldsymbol{x_j}^T, \boldsymbol{x_k}^T, \boldsymbol{x_l}^T] &= Cum[\textstyle\sum_p a_{ip} \boldsymbol{s_p}^T, \sum_q a_{jq} \boldsymbol{s_q}^T, \sum_r a_{kr} \boldsymbol{s_r}^T, \sum_s a_{ls} \boldsymbol{s_s}^T] \\ &= \textstyle\sum_{pqrs} a_{ip} a_{jq} a_{kr} a_{ls} Cum[\boldsymbol{s_p}^T, \boldsymbol{s_q}^T, \boldsymbol{s_r}^T, \boldsymbol{s_s}^T] \end{aligned}$$

using the additive properties of cumulants (see Cardoso and Souloumiac, 1993). Moreover, by definition, different sources $\boldsymbol{s_p}^T, \boldsymbol{s_q}^T, \boldsymbol{s_r}^T, \boldsymbol{s_s}^T$ are statistically independent. Therefore,

$$Cum[\boldsymbol{s_p}^T, \boldsymbol{s_q}^T, \boldsymbol{s_r}^T, \boldsymbol{s_s}^T] = \delta(p, q, r, s) \kappa_4(\boldsymbol{s_p}^T)$$

where δ is the Kroenecker symbol and $\kappa_4(\boldsymbol{s_p}^T)$ is the kurtosis (defined above) of sources $\boldsymbol{s_p}^T, \boldsymbol{s_q}^T, \boldsymbol{s_r}^T, \boldsymbol{s_s}^T$ for $p = q = r = s$. Hence,

$$Cum[\boldsymbol{x_i}^T, \boldsymbol{x_j}^T, \boldsymbol{x_k}^T, \boldsymbol{x_l}^T] = \sum_{m=1}^{n} a_{im} a_{jm} a_{km} a_{lm} \kappa_4(\boldsymbol{s_m}^T).$$

The goal of the JADE algorithm is, therefore, to find signals $\boldsymbol{s_u}^T$ which verify the above relation. We shall now show that, to obtain this result, we must use eigen value decomposition as preprocessing step. We transform the observed random vector $\boldsymbol{x}(t)$ linearly so that we obtain a new random vector $\boldsymbol{z}(t)$ which is white, i.e. such that its components are uncorrelated with their variances equal to unity. Let V be a whitening and

$$\boldsymbol{z}(t) = V\boldsymbol{x}(t)$$

the sphered vector. The whitening transformation is always possible. Because whitening is essentially decorrelation followed by scaling, the technique of PCA can be used. The method consists in using the eigenvalue decomposition (EVD) of the covariance matrix: $E[\boldsymbol{x}(t)\boldsymbol{x}^T(t)] = EDE^T$, where E is the orthogonal matrix of eigenvectors of $E[\boldsymbol{x}(t)\boldsymbol{x}^T(t)]$ and D is the diagonal matrix of eigenvalues. Note that $E[\boldsymbol{x}(t)\boldsymbol{x}^T(t)]$ can be estimated in a standard way from the available sample. Whitening can now be done through $\boldsymbol{z}(t) = ED^{-1/2}E^T\boldsymbol{x}(t)$.
Next, we define the matrix

$$U = VA$$

such that

$$\boldsymbol{z}(t) = V\boldsymbol{x}(t) = VA\boldsymbol{s}(t) = U\boldsymbol{s}(t).$$

Since $\boldsymbol{z}(t)$ is white, the matrix $U = VA$ must be orthonormal:

$$U^TU = I.$$

If we suppose that S is a matrix of independent sources, we know from equation (5.1) that the cumulant matrix $Q^z(M)$ associated to a matrix M must be:

$$\begin{aligned}[Q^z(M)]_{ij} &= \sum_{k,l=1}^{n} Cum[\boldsymbol{z_i}^T, \boldsymbol{z_j}^T, \boldsymbol{z_k}^T, \boldsymbol{z_l}^T] M_{kl} \\ &= \sum_{k,l=1}^{n} \sum_{m=1}^{n} u_{im} u_{im} u_{im} u_{im} \kappa_4(\boldsymbol{s_m}^T) M_{kl}.\end{aligned}$$

Therefore, the ICA problem reduces to finding matrices U (orthonormal) and Δ such as:

$$[Q^z(M)] = U \triangle U^T$$

where $\triangle = diag(\kappa_4(\boldsymbol{s_1}^T)\boldsymbol{u_1}^T M \boldsymbol{u_1}, \ldots, \kappa_4(\boldsymbol{s_n})\boldsymbol{u_n}^T M \boldsymbol{u_n})$ for any matrix M and $\boldsymbol{u_i}$ is the ith column of U.

If $\hat{U}$ is an estimate of U, one can estimate A by : $\hat{A} = V^{-1}\hat{U}$. $\triangle$ is diagonal if and only if the sum of the square of the non diagonal elements of $\triangle$ equals zero. So, we can compute $\hat{U}$, the estimate of U as the orthonormal matrix which minimizes :

$$\sum_{i \neq j} [\triangle]_{ij}^2 = \sum_{i \neq j} [\hat{U}^T Q^z(M) \hat{U}]_{ij}^2$$

for any arbitrary $p \times p$ matrix M. In practice, we minimize the above relation for a set of $p \times p$ matrices: $\mathcal{M} = \{M_1, M_2, \ldots\}$. This kind of minimization can be performed using a Jacobi optimisation. The Jacobi method is an iterative technique of optimisation over a set of orthonormal matrices. The orthonormal transform is obtained as a sequence of plane rotations. Each plane rotation is a rotation applied to a pair of coordinates. The $(i,j)th$ plane rotation by an angle ϕ_{ij} changes the coordinates i and j of $\boldsymbol{z}(t)$ according to :

$$\begin{pmatrix} \boldsymbol{z_i}^T \\ \boldsymbol{z_j}^T \end{pmatrix} \leftarrow \begin{pmatrix} \cos\phi_{ij} & \sin\phi_{ij} \\ -\sin\phi_{ij} & \cos\phi_{ij} \end{pmatrix} \begin{pmatrix} \boldsymbol{z_i}^T \\ \boldsymbol{z_j}^T \end{pmatrix},$$

while leaving the other coordinates unchanged. The **Jacobi method** can be summarized as :

1. *Preprocessing step.* Compute a whitening matrix V and set $\boldsymbol{z}(t) = V\boldsymbol{x}(t)$.

2. *One sweep.* For $1 \leq i < j \leq p$, do :

   - Compute the angle ϕ_{ij}, minimizing $\sum_{i \neq j} [\hat{U}^T Q^z \hat{U}]_{ij}^2$ when the pair $(\boldsymbol{z_i}^T, \boldsymbol{z_j}^T)$ is rotated.

   - If $\phi_{ij} < \phi_{min}$, do rotate the pair $(\boldsymbol{z_i}^T, \boldsymbol{z_j}^T)$. ϕ_{min} is a small angle, which controls the accuracy of the optimisation. Typically, we take $\phi_{min} = 10^{-2}/\#\mathcal{T}$.

3. If no pair has been rotated in previous sweep, end. Otherwise go to step 2 for another sweep.

Finally, the **JADE algorithm** can be summarized as:

1. *Preprocessing step.* Estimate a whitening matrix V and set $\boldsymbol{z}(t) = V\boldsymbol{x}(t)$.

2. *Form Statistics.* Estimate a set $\{\hat{Q}_i^z\}$ of cumulants matrices defined as

$$\{Q^z(M_i) | M_i \in \mathcal{M}\}, \text{ where}$$

   $\mathcal{M}$ is any basis for the p^2-dimensional linear space of $p \times p$ matrices. A canonical basis for this space is $\left\{e_k e_l^T | 1 \leq k < l \leq n\right\}$ where e_k is a column vector with a 1 in kth position and 0's elsewhere. So, we have

$$[Q^z(e_k e_l^T)]_{ij} = Cum[\boldsymbol{z_i}^T, \boldsymbol{z_j}^T, \boldsymbol{z_k}^T, \boldsymbol{z_l}^T].$$

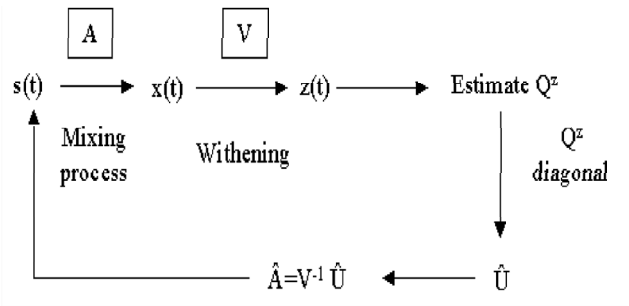

Figure 5.21: JADE algorithm.

3. *Optimise an orthogonal contrast.* Find the rotation matrix $\hat{U}$ such that the cumulant matrices are as diagonal as possible, that is solve

$$\hat{U} = \arg\min \sum_{i \neq j} [\hat{U}^T Q^z \hat{U}]_{ij}^2.$$

(We use the Jacobi method.)

4. *Separate.* Estimate A as

$$\hat{A} = V^{-1}\hat{U}$$

and estimate the sources as

$$\hat{S} = \hat{A}^{-1}x = \hat{U}^T z.$$

The JADE methodology is summarized in figure 5.21.

Appendix B: Variance of the fixed and random effects

$$\text{var}(\widetilde{X}\hat{\boldsymbol{b}}) = \widetilde{X}\text{var}(\hat{\boldsymbol{b}})\widetilde{X}^T$$

Substituting the inverse Fisher information matrix to var($\hat{\boldsymbol{b}}$), one gets:

$$\text{var}(\widetilde{X}\hat{\boldsymbol{b}}) \approx \widetilde{X}(\widetilde{X}^T V^{-1}\widetilde{X} + \left(\Lambda_f \otimes D^T D\right) + \kappa I)^{-1}\widetilde{X}^T.$$

Using similar arguments,

$$\begin{aligned}\text{var}(\widetilde{Z}\hat{\boldsymbol{u}}) &= \widetilde{Z}\text{var}(\hat{\boldsymbol{u}})\widetilde{Z}^T \\ &\approx \widetilde{Z}(\widetilde{Z}^T\widetilde{R}^{-1}\widetilde{Z} + \left(\Lambda_r \otimes D^T D\right) + \kappa I + \widetilde{G}^{-1})^{-1}\widetilde{Z}^T.\end{aligned}$$

Finally,

$$\begin{bmatrix} \text{var}(X\hat{\boldsymbol{b}}) \\ \text{var}(\widetilde{Z}\hat{\boldsymbol{u}}) \end{bmatrix} \approx C(C^T R^{-1} C + B_G + \left(\Lambda \otimes D^T D\right) + \kappa I)^{-1} C^T.$$

where $C = [\widetilde{X}, \widetilde{Z}]$, $\Lambda = diag(\lambda_0, \lambda_1, \ldots, \lambda_p, \underbrace{\lambda^{(r)}, \ldots, \lambda^{(r)}}_{\delta \text{ elements}})$ and

$$B_G = \begin{bmatrix} 0_{(S(P+1))} & 0 \\ 0 & \widetilde{G}^{-1} \end{bmatrix}$$

Appendix C: Computation of the correction factors in F-test

Hastie and Tibshirani (1991) showed using bootstrap that one can estimate trace(HH^T) by 0.75trace(H) + 0.5 for a symmetric matrix H. Thus, we have:

$$\begin{aligned}
\delta_1 = &\quad N - 2\ \mathrm{trace}(H_2) + \mathrm{trace}(H_2 H_2^T) \\
\delta_2 = &\quad N - 4\ \mathrm{trace}(H_2) + 6\ \mathrm{trace}(H_2 H_2^T) \\
&\quad -4\ \mathrm{trace}(H_2 H_2^T H_2) + \mathrm{trace}(H_2 H_2^T H_2 H_2^T) \\
\nu_1 = &\quad N - 2\ \mathrm{trace}(H_1) + \mathrm{trace}(H_1 H_1^T) - \delta_1 \\
\nu_2 = &\quad 0.75\ \nu_1 + 0.5,
\end{aligned}$$

where

$$\begin{aligned}
\mathrm{trace}(H_i H_i^T) = &\quad 0.75\ \mathrm{trace}(H_i) + 0.5 \\
\mathrm{trace}(H_i H_i^T H_i H_i^T) = &\quad 0.75\ \mathrm{trace}(H_i H_i^T) + 0.5
\end{aligned}$$

Using bootstrap, we obtained the following additional approximation:

$$\begin{aligned}
\mathrm{trace}(H_i H_i^T H_i) \approx &\quad -0.0043\ \mathrm{trace}(H_i H_i^T)^2 \\
&\quad +0.9157\ \mathrm{trace}(H_i H_i^T) - 0.0002
\end{aligned}$$

thereby showing that only trace(H_i) must be calculated to obtain the correcting factors.

Appendix D: Construction of the design matrices for the ERP example

We shall now show how to construct the design matrices $\widetilde{X}$ and $\widetilde{Z}$. Consider, for example, only 2 subjects ($\delta = 2$ instead of $\delta = 15$), 2 treatments (placebo, as reference, and Lorazepam) and 3 levels of concentration of the Lorazepam (instead of 10).
We get for Equation (4.42) in matrix notation:

$$\mu = \left(\begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix} \otimes \Phi \right) \operatorname{vec}(B^T) + \left(\begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \otimes \Phi \right) \operatorname{vec}(U^T)$$

$$
\begin{aligned}
&= \begin{bmatrix}
\Phi & 0 & \Phi & 0 & 0 \\
\Phi & 0 & 0 & \Phi & 0 \\
\Phi & 0 & 0 & 0 & \Phi \\
\Phi & \Phi & \Phi & 0 & 0 \\
\Phi & \Phi & 0 & \Phi & 0 \\
\Phi & \Phi & 0 & 0 & \Phi \\
\Phi & 0 & \Phi & 0 & 0 \\
\Phi & 0 & 0 & \Phi & 0 \\
\Phi & 0 & 0 & 0 & \Phi \\
\Phi & \Phi & \Phi & 0 & 0 \\
\Phi & \Phi & 0 & \Phi & 0 \\
\Phi & \Phi & 0 & 0 & \Phi
\end{bmatrix}
\begin{bmatrix}
\boldsymbol{\beta}^{(mean)} \\
\boldsymbol{\beta}^{(Lorazepam)} \\
\boldsymbol{\beta}^{(level=1)} \\
\boldsymbol{\beta}^{(level=2)} \\
\boldsymbol{\beta}^{(level=3)}
\end{bmatrix}
+
\begin{bmatrix}
\Phi & 0 \\
\Phi & 0 \\
\Phi & 0 \\
\Phi & 0 \\
\Phi & 0 \\
\Phi & 0 \\
0 & \Phi \\
0 & \Phi \\
0 & \Phi \\
0 & \Phi \\
0 & \Phi \\
0 & \Phi
\end{bmatrix}
\begin{bmatrix}
\boldsymbol{u}^{(subject=1)} \\
\boldsymbol{u}^{(subject=2)}
\end{bmatrix} \\
&= \widetilde{X}\boldsymbol{b} + \widetilde{Z}\boldsymbol{u},
\end{aligned}
$$

where each block Φ denotes the matrix of B-spline bases with ($N = 128$ rows and $S = 24$ columns):

$$
\Phi = \begin{bmatrix}
\phi_1(t_1) & \phi_2(t_1) & \dots & \phi_{34}(t_1) \\
\phi_1(t_2) & \phi_2(t_2) & \dots & \phi_{34}(t_2) \\
 & \vdots & & \\
\phi_1(t_{128}) & \phi_2(t_{128}) & \dots & \phi_{34}(t_{128})
\end{bmatrix},
$$

and

$$
B = \begin{bmatrix}
\boldsymbol{\beta}^{(mean)^T} \\
\boldsymbol{\beta}^{(Lorazepam)^T} \\
\boldsymbol{\beta}^{(level=1)^T} \\
\boldsymbol{\beta}^{(level=2)^T} \\
\boldsymbol{\beta}^{(level=3)^T}
\end{bmatrix} = \begin{bmatrix}
\beta_1^{(mean)} & \dots & \beta_{34}^{(mean)} \\
\beta_1^{(Lorazepam)} & \dots & \beta_{34}^{(Lorazepam)} \\
\beta_1^{(level=1)} & \dots & \beta_{34}^{(level=1)} \\
\beta_1^{(level=2)} & \dots & \beta_{34}^{(level=2)} \\
\beta_1^{(level=3)} & \dots & \beta_{34}^{(level=3)}
\end{bmatrix},
$$

$$U = \begin{bmatrix} \boldsymbol{u}^{(subject=1)^T} \\ \boldsymbol{u}^{(subject=2)^T} \end{bmatrix} = \begin{bmatrix} u_1^{(subject=1)} & \dots & u_{34}^{(subject=1)} \\ u_1^{(subject=2)} & \dots & u_{34}^{(subject=2)} \end{bmatrix}.$$

Consequently, in the matrix $\widetilde{X}$, each column is in fact a block of columns because each element Φ is a matrix. The first block of columns describe the mean curve. The second block of columns correspond to Lorazepam effect. The blocks of columns 3, 4 and 5 correspond to the 3 levels of concentration of Lorazepam. The blocks of rows 1, 2, 3, 7, 8 and 9 correspond to curves under placebo and the others blocks of rows to curves under Lorazepam. The blocks of rows 1, 4, 7 and 10 correspond to curves under the first level of concentration of Lorazepam, the blocks of rows 2, 5, 8 and 11 correspond to curves under the second level of concentration of Lorazepam and the others blocks of rows to curves under the third level of concentration of Lorazepam. Matrix $\widetilde{Z}$ allows specific subject pattern.