Sparse matrix factorization for Brain Computer Interfaces

Alberto Llera Arenas
Donders Institute/Biophysics Department
Radboud University Nijmegen
The Netherlands.
a.llera@donders.ru.nl

Vicenç Gómez
v.gomez@science.ru.nl

Hilbert J. Kappen
B.Kappen@science.ru.nl

Abstract

We present a novel sparse dimensionality reduction approach to reconstruct biological signals for brain computer interfaces (BCI). The proposed technique may be used in the design of an adaptive Brain Computer Interface which uses interaction error potentials.

1. Introduction

Interaction error potentials (IEP) are potentials detected in the recorded EEG of a subject controlling a device, just after the device performs an error. The error is the difference between the result of the action that the subject expected, based on his/her action, and the actual outcome.

Since the 1990's there has been many studies related to the presence of error potentials. They can be classified as follows: the response error potential [3] found in speeded reaction tasks; the feedback error potential [8] which appears in reinforcement learning tasks; the observation error potential [12] and finally, the IEP, which can be detected in a Brain Computer Interface (BCI) context [4].

The precise detection of an IEP after the BCI makes a classification error can help us to construct a more robust BCI, by either correcting the BCI output directly, or more interestingly, by adapting the BCI classifier so that it is less likely to make a similar mistake in the future. This idea is illustrated in Figure 1.

From EEG studies it is well known [3, 4, 8, 12] that the error (as we introduced above) is usually followed by what is called event-related negativity (ERN) which is found in the α-band in fronto-central channels. More recently, an MEG study [7] on the detection of error fields in MEG has shown an increase in the frontal μ-power and a decrease in the posterior α and central β-power.

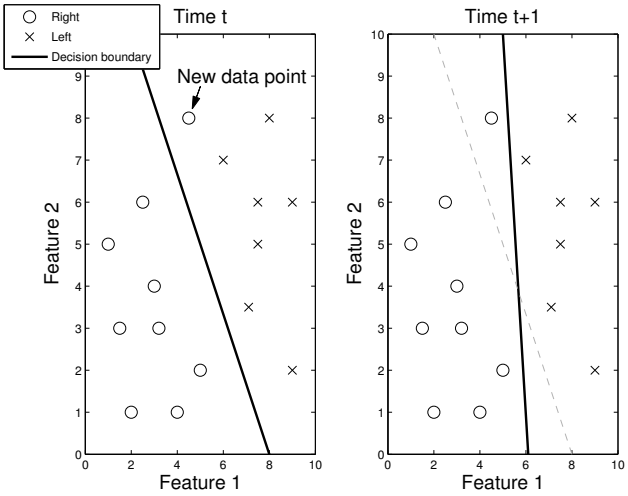

Figure 1. Illustration of an adaptive BCI for a binary task. Each point is labeled with the movement (class) that was intended by the subject (left or right) and denotes a brain state encoded using two features. Bold line indicates the decision boundary of the BCI classifier. **(Left)** A new point is misclassified. The IEP recognized by the BCI provides a mechanism to detect the misclassification. **(Right)** The decision boundary is changed and the BCI is adapted during performance.

The application of the IEP to BCI [4] requires its reliable detection. The IEP may in principle be localized in various channels, various frequency bands, and may be subject dependent.

In this paper, we propose a novel dimensionality reduction approach which can be used to analyse the IEP. We propose a sparse version of singular value decomposition (SVD) that describes the high dimensional signal as a sum of a small number of sparse templates that change through time. The sparsity means that the number of channels that are used in each template is small and it will greatly im-

978-1-4244-4799-2/09/$25.00

prove the interpretability of our findings. Our approach is then related to works which rely on signal decomposition using different spatio-temporal features [6, 10] and opens new doors on how to classify the interaction error fields (IEF, which are the MEG equivalent of the IEP) since we do not need to focus in just a few electrodes, but we may use all the electrodes to increase the quality of the classification.

Our approach is presented without any preselected frequency band for detection of IEF, since the aim of this work is only to present a new method and is not specially focused on solving the IEF classification. However, it can be applied to any specific frequency band that previous knowledge might indicate is the most relevant for a particular problem.

2. Experimental setup

We describe now the experimental framework we used in our data acquisition. The main goal of this experimental design is to gain insight into how error signals are encoded in the brain. Up to now, we gathered measurements from two subjects. Each subject performed 6 sessions composed of 84 trials with a minute between two sessions. We plan to acquire data from 25 more subjects.

All the data used during this work was collected using an MEG system with 275 channels from which 273 were in use. EOG and ECG were also recorded and trials with ocular or muscular artifacts were removed from the data using an automatic routine.

The experiment is designed as follows:

1. First, two squares and a fixation cross appear in the screen.

2. After 300 ms, the fixation cross becomes an arrow (pointing to left or right). The subject is instructed to direct the attention to the direction pointed by the arrow points *while keeping the sight in the center of the screen*.

3. After 2000 ms, the arrow disappears and is replaced with a text indicating the decision of the device (*right* or *left*). This lasts for 1000 ms, and it is the period of main interest.

4. Finally, the text disappears and the two squares remain in the screen for 1000 ms before the new trial starts.

Note that subjects are instructed to control the device using directed (or *covert*) attention, a well known paradigm for BCI control, based on the lateralization of the power on the α band in the posterior channels [11]. However, we could also have used another paradigm such as, for instance, motor imagery, without any change in our protocol.

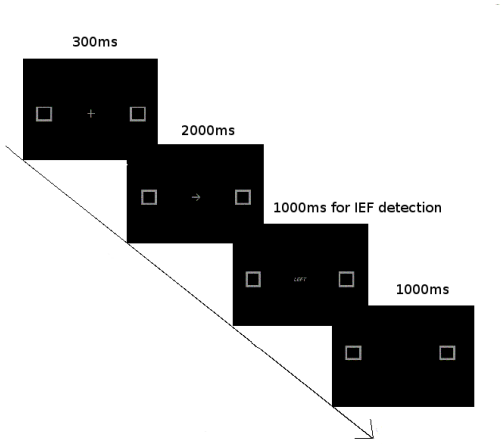

Figure 2. Experimental protocol.

In this preliminary setup, to focus in the goal of error detection, the device returns automatically a random 20% of error responses. We labeled as *error* trials those with the wrong feedback (when the text does not correspond to the direction pointed by the arrow) and *correct* trials otherwise.

The length of the trials was reduced to 1800 ms. For that we selected the full period for IEF detection (1000 ms) plus 800 ms of the arrow. The recording sampling rate was 1200 Hz which gave us a total of 2160 time points per trial. This means that our data matrix for a single trial has size $n \times t$, where $n = 273$ and $t = 2160$.

3. Theoretical framework

In this section we present our method to obtain a reconstruction of the data using a reduced and sparse set of features. First, we describe how we perform dimensionality reduction and then we focus on sparsity.

3.1. Matrix Factorization and Dimensionality Reduction

Lets assume that we have a data matrix $Y \in \mathcal{M}_{n\times t}$ where n and t indicate number of channels and time-steps respectively. When facing the problem of matrix factorization in a general setting, our goal is to find two matrices F and G that minimize

$$||FG - Y||_2^2. \tag{1}$$

where $||FG - Y||_2$ is the Frobenius norm of the matrix $FG - Y$. This can be seen as constructing a basis matrix F for which the coefficients for the data are in matrix G.

A common first step when classifying data is to reduce the effect of the noise and use the most informative features. This is usually done using dimensionality reduction techniques. In our case, we retain the most informative k basis vectors and discard the rest.

In this general setting, we see that for any given $k \in \mathbb{N}$ we can find matrices $F \in \mathcal{M}_{n\times k}$ and $G \in \mathcal{M}_{k\times t}$ that minimize expression (1). We are interested in the case where $k \ll n$. Here appears the model selection problem, or how to select the parameter k.

For $k = n$, the singular value decomposition (SVD) can be used to factorize Y and obtain three matrices: $U \in \mathcal{M}_{n\times n}$, a diagonal matrix $S \in \mathcal{M}_{n\times t}$ and $V' \in \mathcal{M}_{t\times t}$ such that

$$Y = USV^*. \tag{2}$$

where * denotes conjugate transpose of a matrix, and the singular values of Y are sorted by their absolute value in descending order along the diagonal of S. If we define $\mathbf{F} = U$ and $\mathbf{G} = SV^*$, such a factorization corresponds to the minimization of (1) for the case of $k = n$.

For $k \ll n$, we define the matrix F considering only the first k columns of $\mathbf{F}$ and equivalently, G considering the first k rows of $\mathbf{G}$. Hence, FG is an approximation of Y, which becomes more accurate as k increases.

In this work, we use the *Akaike information criterion* (AIC) [1] to select the value of k. In our particular case, under the assumption that errors are normally distributed, the AIC selects the k which minimizes

$$||FG - Y||_2^2 + k(n + t). \tag{3}$$

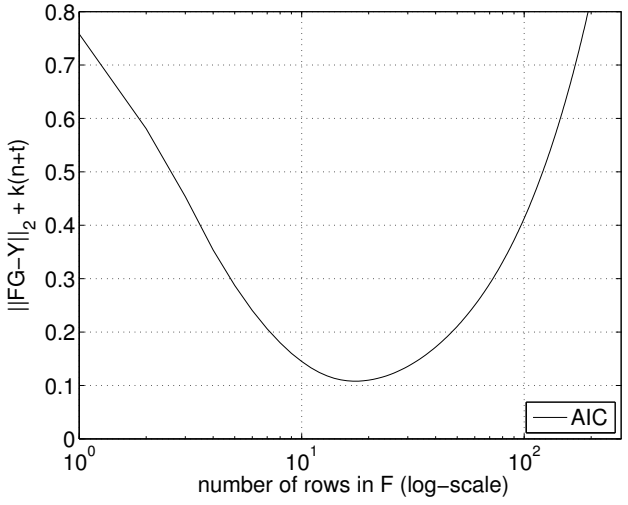

Figure 3. The Akaike information criterion (AIC) for model selection is used to select the number of features k in our approach.

Figure 3 shows the AIC for different trials corresponding to the experiment described in section 2. From now on we can assume that k is fixed.

3.2. Sparsity

Up to now we have described how to represent the matrix Y using a *reduced* basis F of k vectors. Each of the k vectors can be considered as a feature composed of a mixture of different channels. To reconstruct the original signal over time these features are weighted by the corresponding coefficients in G.

In this section we explain how to make the basis F sparse. Enforcing sparsity in F will result in features composed of a reduced number of channels thus providing a more compact and structured representation of the data and consequently, increasing the interpretability of the recovered signal.

A natural method to obtain a sparse F is an extension to matrices of the ℓ_1-norm regularized least squares method [2]. Given the data Y, and assuming an initial G fixed, we are interested in the F which minimizes

$$||FG - Y||_2^2 + \lambda||F||_1, \tag{4}$$

where the $||F||_1$ is the sum of the absolute values of the elements in the matrix F.

Instead of minimizing Equation (4) directly, we make use of an extension of the algorithm described in [5]. Given a matrix A and a vector y, [5] describes an interior-point method for solving x which minimizes:

$$||Ax - y||_2^2 + \lambda||x||_1. \tag{5}$$

Note first that the minimization of (4) is equivalent to the minimization of

$$||G^{\mathrm{T}}F^{\mathrm{T}} - Y^{\mathrm{T}}||_2^2 + \lambda||F||_1. \tag{6}$$

Thus [5] gives a solution to our problem for $n = 1$.

Now denote the s-th column of Y^{T} by Y_s^{T}. Using [5] we can also find a solution to

$$||G^{\mathrm{T}}x - Y_s^{\mathrm{T}}||_2^2 + \lambda||x||_1, \tag{7}$$

where x is exactly the s-th column of F^{T}. Repeating this procedure for every $s \in \{1 \dots n\}$ we can find F, a solution of (6) and consequently of (4). In other words, we have expressed the global minimization (4) as n independent minimizations of the form (7), one for each channel.

Parameter λ plays the role of a trade-off between sparsity and quality of the reconstruction. On one hand, for a small λ, the quality of the reconstructed signals will be high. However, F will less sparse. On the other hand, a large λ will result in a very sparse F, but in poor approximations of the original signals.

After having defined a procedure to find a reduced and sparse basis F, we can find a new G which minimizes Equation (1). Since (1) is a differentiable quadratic form in G, the solution can be found analytically and we can write the optimal G in closed form:

$$G = (F^{\mathrm{T}}F)^{-1}(F^{\mathrm{T}}Y). \tag{8}$$

Note that the inverse $(F^{\mathrm{T}}F)^{-1}$ is only defined when $\text{rank}(F) = k$, and this is not generally guaranteed. In particular, the more sparse F is, the more likely is that $\text{rank}(F) < k$. This means that there exists a maximal λ which limits the level of sparsity that can be achieved by our method. In practice, this limitation does not restrict our method, as we will show in the next sections.

4. Algorithms for Sparse matrix factorization

After introducing the theoretical building blocks of our approach, we present two possible algorithms. Both algorithms take as input the BCI data Y, the regularization parameter λ and the desired sparsity of the solution (number of zero entries in F).

Algorithm 1 applies SVD to the original signal Y and then uses AIC (see Section 3.1) to select k. This results in a matrix G with k rows which is used in the ℓ_1-norm minimization (step 4 of Algorithm 1) to find the sparse basis $F^\star$. After the minimization, some of the entries in $F^\star$ are very small in absolute value. We set the required entries to zero as long as the matrix $F^\star$ has full rank (in practice, we always found full rank matrices even using 50% of sparsity).

Algorithm 1

Require: x (number of zeros in F), λ and matrix Y
1: $\mathbf{G} \leftarrow SVD(Y)$.
2: $k \leftarrow AIC$.
3: $G \leftarrow$ select k rows of $\mathbf{G}$.
4: $F^\star \leftarrow \text{argmin}_{F'} ||F'G - Y||_2^2 + \lambda ||F'||_1$.
5: **repeat**
6: $\quad (i,j) \leftarrow$ find smallest non-zero absolute value $F^\star$
7: $\quad F^\star(i,j) := 0$
8: **until** $F^\star$ has x zeros or $\text{rank}(F) < k$.
9: $G^\star \leftarrow \text{argmin}_G ||FG - Y||_2^2$
10: **return** $F^\star, G^\star$

Figure 4 shows the behavior of the algorithm for three different values of λ as a function of the number of zeros. As can be seen, the larger the λ, the more sparse can F be made without increasing significantly the error. Note, however, that for small λ, the initial errors (those corresponding to non-sparse solutions) are smaller than for large λ.

The interplay between λ and the level of sparsity suggests a modification of the algorithm in which the matrix F resulting from SVD, instead of $F^\star$, is used as a final basis. The latter is used only to select which entries of F must be zero. Algorithm 2 describes this alternative approach.

Figure 5 shows a comparison of both methods for a fixed $\lambda = 300$ as a function of the number of zero entries. As can be seen, the alternative algorithm performs better than the previous one as long as the solution is not very sparse.

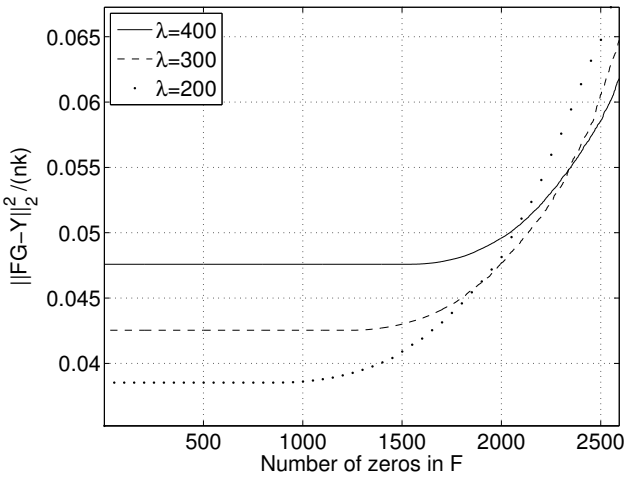

Figure 4. Performance of Algorithm 1 from one random trial. k is fixed to 19 using AIC and $\lambda = \{200, 300, 400\}$.

Algorithm 2

Require: x (number of zeros in F), λ and matrix Y
1: $\mathbf{F}, \mathbf{G} \leftarrow SVD(Y)$.
2: $k \leftarrow AIC$.
3: $F, G \leftarrow$ select k cols. and rows from $\mathbf{F}, \mathbf{G}$ respectively.
4: $F^\star \leftarrow \text{argmin}_{F'} ||F'G - Y||_2^2 + \lambda ||F'||_1$
5: **repeat**
6: $\quad (i,j) \leftarrow$ find smallest non-zero absolute value $F^\star$
7: $\quad F(i,j) := 0$
8: **until** F has x zeros or $rank(F) < k$.
9: $G^\star \leftarrow \text{argmin}_G ||FG - Y||_2^2$
10: **return** $F, G^\star$

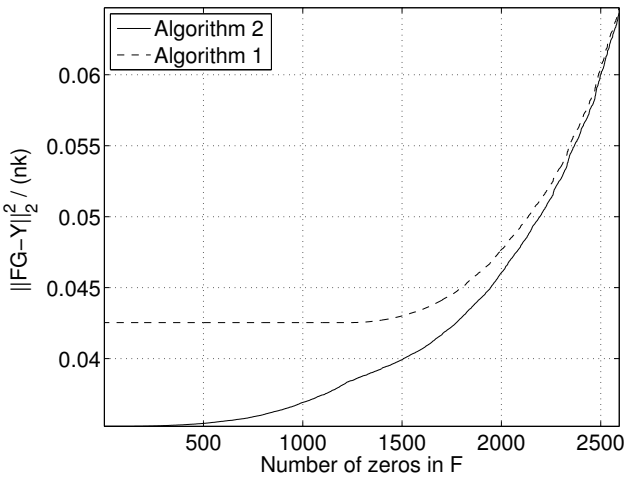

Figure 5. Performance of Algorithm 2 (solid) versus Algorithm 1 (dashed) from one random trial. $k = 19$ and $\lambda = 300$.

4.1. Choosing the regularization parameter λ

Given a level of sparsity, is there a λ for which the error is minimal? If this is the case, we could choose automatically

the λ provided the number of zero entries in the matrix F.

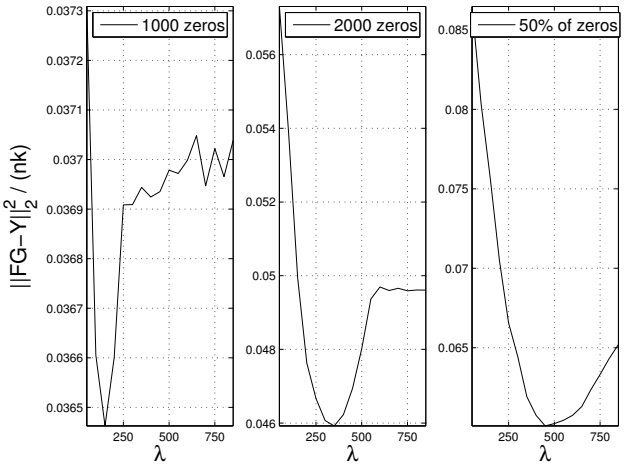

Figure 6. Performance of Algorithm 2 as a function of λ for different levels of sparsity. Results are equivalent if we consider all the trials.

Figure 6 shows the performance of Algorithm 2 as a function of λ for different levels of sparsity: 1000, 2000 and 2594 (50% of the entries). It shows that there exists a optimal value of λ for any level of sparsity. This optimal value could be easily found, for instance, using line search.

For both algorithms we found that the optimal λ, as well as the error, are larger as we increase the level of sparsity.

4.2. Why not make sparse the SVD directly?

Another way to look at the problem would be to simply make zeros the positions of smallest absolute values of F, and then updating G using (8). In Figure 7 we show that this is not a good strategy. As we can see, the error of Algorithm 2 with $\lambda = 300$ is *always* smaller than this alternative approach, regardless of the level of sparsity, showing the advantage of using the ℓ_1-norm minimization.

This can also be viewed from the perspective that the regularization term used in step 4 of Algorithm 2 has by definition the property to produce parameter shrinkage in the least relevant directions of the data.

5. Results: Sparse reconstruction of signals

In this section we illustrate with an example the quality of the reconstruction made by our method. We show results for the MEG signal acquired according the experimental procedure described in Section 2.

Step 2 of Algorithm 2 gives $k = 19$. Since the MEG system has 273 active channels, this result in a matrix $F \in \mathcal{M}_{273\times 19}$, so the matrix F has a total of 5187 elements. For this example we will require Algorithm 2 to make 2000 zeros in F. For this level of sparsity, we selected $\lambda = 300$.

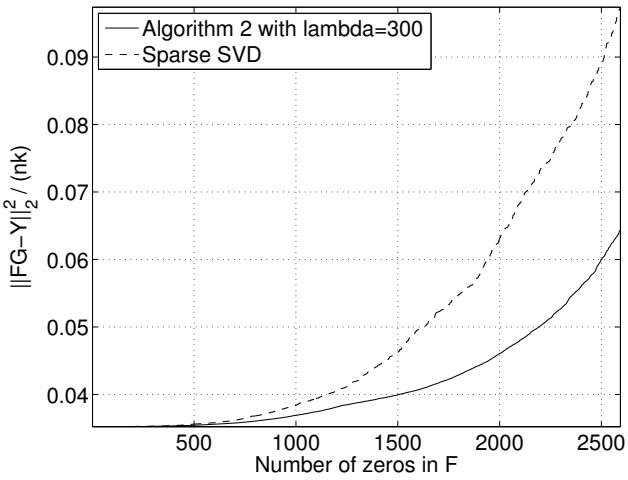

Figure 7. Performance of Algorithm 2 (solid) versus sparsifying the initial SVD (dashed). $k = 19$ and $\lambda = 300$.

As expected, we observe that columns of F associated with the most relevant features (leftmost columns) are less sparse than the rightmost columns, However, it is not the case that a column becomes totally zero, which would indicate that $\text{rank}(F) < k$.

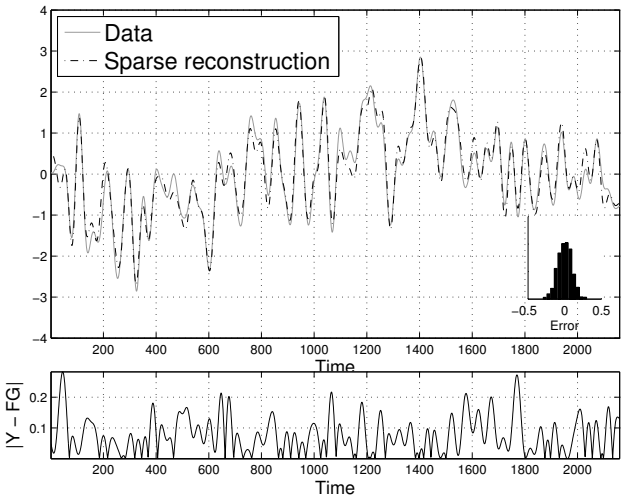

Figure 8. Example of signal reconstruction for one channel and one trial selected randomly. (**Top**) Original signal (grey solid) and the approximation (black dashed-dotted) over time. The approximation was calculated using Algorithm 2 with $k = 19$, $\lambda = 300$ and 2000 zeros in F. The inset shows an histogram of the residuals, which look normally distributed. (**Bottom**) Residuals as a function of time.

Figure 8 illustrates the reconstruction obtained from a random channel (random row of Y) in one trial using Algorithm 2. For this particular channel there are 7 zeros out of the 19 elements in the respective row of F. The sparsity of the whole matrix F is 38%, whereas the selected channel

appears as irrelevant in 37% of the features. As can be seen from the figure, the reconstruction is very accurate.

5.1. Discussion and ongoing research

We have developed a method to decompose a space/time signal into a small set of features and shown its applicability in MEG signal reconstruction. The method not only leads to a more understandable signal but, more importantly, is also appropriate to be used in a BCI setup, such as the one presented in Section 1, where the reconstructed signal is used in the classification of IEP. This is our current direction of research.

We devise some possibilities to improve/extend the proposed method. First, since the role of the regularizer in our algorithms is just that to select which positions in F should be zero, we might get similar results by using the *Tikhonov* regularization, also known as ℓ_2-regularized least squares [9]. This approach would be much more efficient in computational terms since the regularization becomes a a quadratic differentiable form which therefore has an analytic solution. We have promising preliminary results in this direction.

Another extension is to perform the analysis into the frequency domain, more often used in BCI. Notice that the method can be easily adapted to this case: first, the source data Y would be transformed using a selected band of frequencies (low frequencies are more convenient in our paradigm) and then our sparse factorization would be applied to the transformed data. The resulting basis would constitute a set of spectral features which change over time, the analogous counterpart to our original features.

References

[1] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

[2] S. Boyd and L. Vandenberghe. Effects of error in choice reaction tasks on the ERP under focused and divided attention. In *Convex Optimization*, pages 308–310. Cambridge University Press, 2001.

[3] M. Falkenstein, J.Hohnsbein, J. Hoormann, and L. Blanke. Effects of error in choice reaction tasks on the ERP under focused and divided attention. In C. Brunia, A. Gaillard, and A. Kok, editors, *Psychophysiological Brain Research*, pages 192–195. Tilburg Univerity Press, Tilburg, 1990.

[4] P. Ferrez. *Error-related EEG potentials in brain-computer interfaces*. Phd thesis, Thèse Ecole polytechnique fédérale de Lausanne EPFL, no 3928, 2007.

[5] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky. An Interior-Point Method for Large-Scale ℓ_1-Regularized Least Squares. *IEEE Journal of Selected Topics in Signal Processing*, 1(4):606–617, 2007.

[6] Z. J. Koles, J. C. Lind, and A. C. K. Soong. Spatio-temporal decomposition of the EEG: a general approach to the isolation and localization of sources. *Electroencephalography and Clinical Neurophysiology*, 95(4):219 – 230, 1995.

[7] A. Mazaheri, I. L. Nieuwenhuis, H. van Dijk, and O. Jensen. Prestimulus alpha and mu activity predicts failure to inhibit motor responses. 30(6):1791–1800, 2009.

[8] W. H. R. Miltner, C. H. Braun, and M. G. H. Coles. Event-related brain potentials following incorrect feedback in a time-estimation task: Evidence for a "generic" neural system for error detection. *J. Cognitive Neuroscience*, 9(6):788–798, 1997.

[9] A. Neumaier. Solving ill-conditioned and singular linear systems: A tutorial on regularization. *SIAM Review*, 40(3):636–666, 1998.

[10] P. A. Valdés-Sosa, M. Vega-Hernández, J. M. Sánchez-Bornot, E. Martínez-Montes, and M. A. Bobes. EEG source imaging with spatio-temporal tomographic nonnegative independent component analysis. *Human Brain Mapping*, 30(6):1898 – 1910, 2009.

[11] M. van Gerven and O. Jensen. Attention modulations of posterior alpha as a control signal for two-dimensional brain-computer interfaces. *Journal of Neuroscience Methods*, 179(1):78 – 84, 2009.

[12] H. T. van Schie, R. B. Mars, M. G. H. Coles, and H. Bekkering. Modulation of activity in medial frontal and motor cortices during error observation. *Nature Neuroscience*, 7(5):549–554, 2004.