CCIT Report #531

May 2005

Learning Spatial and Temporal Filters for Single-Trial EEG Classification

Dmitry Model and Michael Zibulevsky

May 7, 2005

Abstract

There is a wide variety of electroencephalography (EEG) analysis methods. Most of them are based on averaging over multiple trials in order to increase signal-to-noise ratio. The method introduced in this article is a *single trial* method. Our approach is based on the assumption that the "real brain signal" of each task is smooth, and is contained in several sensor channels. We propose two stage preprocessing. At first, we use *spatial* filtering, by taking weighted linear combination of sensors. At the second step, we perform *time-domain* filtering. Both stages are performed *blindly*, by maximizing the between class discrimination and minimizing the total variation of result average or , alternatively, suppressing the signal at the windows, where it is known to be absent. No other information on signals of interest is assumed to be available.

1 Introduction

People have speculated that EEG might be used as alternative communication channel, which allows the brain to act bypassing peripheral nerves and muscles, since electroencephalography was first described by Hans Berger in 1929 [1]. First simple communication systems, that were driven by electrical activity recorded from the head, appeared about three decades ago [2]. In the past years, it has been shown that it is possible to recognize distinct mental precesses from online EEG (see, for example [3, 4, 5, 6]). By associating certain EEG patterns to simple commands, it is possible to control a

computer, creating an alternative communication channel, which is usually called *Brain-Computer Interface* (BCI) [2, 7].

One of the most complicated problem of the BCI is classifying very noisy EEG signals, obtained by registering the brain activity of the subject. The first approach suggests dealing with this problem by requiring extensive training, in order to teach the subject to acquire self-control over a certain EEG components, such as sensorimotor μ-rhythm [7] or slow cortical potentials [8]. This ability to create certain EEG patterns *at will* is translated by BCI system to cursor movement [7, 9] or selection of letters or words on computer monitor [10, 8].

The second approach suggests developing *subject-specific* classifiers to recognize different cognitive processes from EEG signals [4, 5, 11]. In this case, the typical BCI procedure consists of two stages. First, the person trains the system by concentrating on predefined mental tasks. Usually two different tasks are used in the training. BCI registers several EEG samples of each task. Then, the training data is being processed in order to build a classifier. In the second stage, the subject concentrates on one of the tasks again, and the system *automatically* classifies the EEG signals. The key for successful classification is a good preprocessing of raw data. The objective of this paper is to develop preprocessing methods, which will improve the classification accuracy.

Some preprocessing methods explore the fact that the signal of interest is contained in several sensor channels, since the skull and scalp cause a spatial smearing of the cortical signals. Those methods suggests spatial filtering, in order to increase signal-to-interference ratio and maximize the between-class discrimination [12, 13]. However, these methods make no explicit use of the time courses of EEG signals.

The proposed method is based on the assumption that the signal of interest of each task is temporary *smooth* (i.e. has limited total variation) and/or is expected to be small in certain time windows, where the task is not performed. We propose two stage preprocessing algorithm. At first, we perform *spatial* filtering, by taking weighted linear combination of sensors. At the second step, we perform *time-domain* filtering. Both stages are performed *blindly*. Filter coefficients are found by optimizing the between class discrimination and the smoothness of result average. No other information on signals of interest is assumed to be available. Our simulations shows, that the proposed preprocessing significantly improves the classification rate, with respect to unprocessed data (simple sum of channels or choosing the best sen-

sor).

We have also developed misclassification rate lower bound, which is applicable for the experiments with synthesized signals. This bound shows how well can we perform signal reconstruction (and further classification) based on spatial integration only. Our simulations shows, that in majority of cases we reach the bound or stay very close to it. If we use *time-domain* filtering in addition to spatial integration, then we perform even better.

This paper is organized as follows. In Section 2 we describe the first stage of our preprocessing method, which is based on spatial integration. In Section 3 we develop additional method for spatial filtering based on Eigenvalue Decomposition. In Section 4 we develop a bound, which shows how well can we perform signal reconstruction (and further classification) based on spatial integration only. In section 5 we describe the second stage of our algorithm, which is based on time-domain filtering. Section 6 is devoted to computational experiments. Finally, conclusions are summarized in Section 7.

2 Spatial Integration Method

2.1 Data Description

In our simulation we use several data sets, recorded with different number of sensors, sampling rate, etc. Details on these data sets are available in Section 6. Here we provide the general description of the data format, which we use in our preprocessing method.

Suppose EEG data was recorded using S channels. Single trial signals, corresponding to one of the two possible mental tasks were taken from the raw data, synchronized by some external stimuli or cue, and they are T samples long. Each single trial is stored in the $T \times S$ matrix. Let us denote $X_l^1, 1 \leq l \leq L$ trials that belong to the first class, and X_m^2 , $1 \leq m \leq M$ trials that belong to the second class.

If we produce the averaging across the trials, we obtain

$$X_{avg}^1 = \frac{1}{L} \sum_{l=1}^{L} X_l^1 \qquad X_{avg}^2 = \frac{1}{M} \sum_{m=1}^{M} X_m^2$$

where X_{avg}^1 and X_{avg}^2 are $T \times S$ matrices.

2.2 The Method

In our model, we assume that **each sensor** records the following signal:

$$x_i(t) = a_i s^j(t) + n_i(t) \tag{1}$$

where a_i is the coupling coefficient for sensor i, $n_i(t)$ denotes the noise and background activity recorded by the sensor and $s^j(t)$, $j \subset \{1, 2\}$ is the response to one of the two possible mental tasks.

We will use a linear estimate of single trial signals

$$\hat{s}_i^1 = X_i^1 w \ , \ \hat{s}_j^2 = X_j^2 w \tag{2}$$

where w is the $S \times 1$ *weighting vector*.

The average of the estimated signals is:

$$\hat{s}_{avg}^1 = X_{avg}^1 w \ , \ \hat{s}_{avg}^2 = X_{avg}^2 w \tag{3}$$

Using the above notation, we can formulate our objective as finding the weighting vector w such, that will maximally discriminate between the average estimated signals $\hat{s}_{avg}^1$ and $\hat{s}_{avg}^2$, while keeping single trial estimated signals $\hat{s}_l^1$ and $\hat{s}_m^2$ $(1 \leq l \leq L \ , \ 1 \leq m \leq M)$ smooth. The smoothness can measured by the *total variation*, defined by

$$\begin{aligned} TV &= \sum_{l=1}^{L} \sum_{t=1}^{T-1} y_l^1(t) + \sum_{m=1}^{M} \sum_{t=1}^{T-1} y_m^2(t) \\ \text{where} \ \ y_l^1(t) &= |\hat{s}_l^1(t+1) - \hat{s}_l^1(t)| \ , \ 1 \leq t \leq T-1 \\ \text{and} \ \ y_m^2(t) &= |\hat{s}_m^2(t+1) - \hat{s}_m^2(t)| \ , \ 1 \leq t \leq T-1 \end{aligned} \tag{4}$$

This leads to following objective function:

$$\begin{aligned} &\min_w - \left\| \hat{s}_{avg}^1 - \hat{s}_{avg}^2 \right\|_2^2 + \mu TV \\ &s.t. \left\| w \right\|_2 = 1 \end{aligned} \tag{5}$$

where μ is a tradeoff parameter, which is intended to balance between smoothness of signals and between class discrimination. We are forcing the norm of weighting vector w to remain constant in order to prevent degenerate solution of $\|w\| \to \infty$ or $\|w\| \to 0$.

If we substitute expressions for $\hat{s}^1_{avg}$,$\hat{s}^2_{avg}$ and TV from equations (2),(3) and (4), the objective function (5) becomes:

$$\min_w -\left\|X^1_{avg}w - X^2_{avg}w\right\|_2^2 + \mu\left(\sum_{l=1}^{L}\left\|Y^1_l w\right\|_1 + \sum_{m=1}^{M}\left\|Y^2_m w\right\|_1\right) \qquad (6)$$
$$s.t.\ \|w\|_2 = 1$$

where $\|\cdot\|_1$ is the first norm and

$$Y^1_l(t,i) = X^1_l(t+1,i) - X^1_l(t,i)\ ,\ 1 \le t \le T-1$$

$$Y^2_m(t,i) = X^2_m(t+1,i) - X^2_m(t,i)\ ,\ 1 \le t \le T-1$$

note that $y^1_l = Y^1_l w$, $y^2_m = Y^2_m w$.

2.3 Getting Rid of the Tradeoff Parameter

In objective (6) there is a need to choose a value for a tradeoff parameter μ. Although we have found out by our simulations, that the optimization result is quite robust to the change of value of μ, the need to subjectively asses the tradeoff parameter is still an essential drawback. In this subsection, we propose an elegant way to rewrite the objective function in such a way, that it will contain no parameter any more.

For beginning, let's notice, that the norm and the sign of the vector w have no significance. We are interested only in *relative to each other* values of its elements. In other words, we want to find such w, which will satisfy two conditions. *First*, it will minimize a value of TV - the second term of (6), when a value of the first term is constant. *Second*, it will minimize a value of the first term, when a value of TV is constant. Now, let's write the objective function which will satisfy the above conditions[1]:

$$\min_w \sum_{l=1}^{L}\left\|Y^1_l w\right\|_1 + \sum_{m=1}^{M}\left\|Y^2_m w\right\|_1 \qquad (7)$$
$$s.t.\ \left\|X^1_{avg}w - X^2_{avg}w\right\|_2^2 = 1$$

[1]Alternatively, we may switch role of main term and constraint of objective (7)

One can notice, that above objective function satisfies the first condition at the solution point by definition. The second condition is also satisfied. This can be proved in the following way. Suppose w_{opt} is a solution of (7). Let's assume by contradiction, that there exist w_{new}, such that $TV(w_{new}) = TV(w_{opt})$ and $\left\|X^1_{avg}w_{new} - X^2_{avg}w_{new}\right\|_2 = c^2 < 1$. In such a case, $w = \frac{1}{c}w_{new}$ would satisfy the constraint, while $TV(w) < TV(w_{opt})$. This contradicts the assumption, that w_{opt} is a solution of (7). Thus, the second condition also holds.

Although the problems (6) and (7) are not completely equivalent, the problem (7) can be viewed as such, that optimally (and automatically) chooses the tradeoff parameter μ. The following example will explain, what do we mean by optimality. Suppose $\hat{w}$ is a solution of (6) for some value of μ and w_{opt} is a solution of (7). Then, according to what we have proven, w_{opt} (after re-scaling) will provide smaller or equal value of TV for the same degree of between class discrimination, and greater or equal between class discrimination for the same value of TV. This is true for any value of μ, thus w_{opt} is really the optimal solution.

The alternative view of the objective (7) is from basis pursuit perspective [14]. The TV term in (7) may be replaced by the l_1 norm of coefficients of signal representation in some basis (i.e. short-time Fourier transform, wavelet transform, etc.), which is expected to be sparse (see for example [15, 16]).

2.4 Optimization

Since it is a constrained optimization problem, it is convenient to minimize the objective function (7) by Lagrange Multipliers technique. For this purpose, we will need to calculate the gradient of the main and the constraint terms.

Before we proceed, we should notice, that the main term of (7) is not differentiable. Hence, for optimization, we will use the smooth approximation of absolute value function.

$$\psi(t) = c(\frac{|t|}{c} - \log(1 + \frac{|t|}{c})) \tag{8}$$

Note that $\psi'(t)$ is defined at $t = 0$:

$$\psi'(t) = \frac{t}{c + |t|} \tag{9}$$

The approximation becomes more accurate, when $c \to 0$.

The modified problem will receive the following form:

$$\begin{aligned} &\min_{w} \sum_{l=1}^{L} 1^T \psi\left(Y_l^1 w\right) + \sum_{m=1}^{M} 1^T \psi\left(Y_m^2 w\right) \\ &s.t. \left\| X_{avg}^1 w - X_{avg}^2 w \right\|_2 = 1 \end{aligned} \tag{10}$$

where 1 is a vector of ones, and the application of $\psi(\cdot)$ to a vector is element-wise.

Let's denote the the main term of (10) as $f(w)$, and rewrite the constraint to be

$$g(w) = \|X_{avg} w\|_2^2 = w^T X_{avg}^T X_{avg} w$$

where $X_{avg} = X_{avg}^1 - X_{avg}^2$.

Now, we can easily calculate the gradients of $f(w)$ and $g(w)$, using the matrix derivations and the chain rules:

$$\nabla f(w) = \sum_{l=1}^{L} \left(Y_l^1\right)^T \psi'\left(Y_l^1 w\right) + \sum_{m=1}^{M} \left(Y_m^2\right)^T \psi'\left(Y_m^2 w\right) \tag{11}$$

$$\nabla g(w) = X_{avg}^T X_{avg} w \tag{12}$$

This calculus is sufficient for minimizing the objective function (10) using Lagrange Multipliers method.

3 Learning Spatial Integration weights through Eigenvalue Decomposition

In this section, we propose a solution of objective function similar to (7) by Eigenvalue Decomposition (EVD). Let's have a look at the following problem:

$$\begin{aligned} &\max_{x} \|Ax\|_2^2 \\ &s.t. \|x\|_2^2 = 1 \end{aligned} \tag{13}$$

where A is matrix, and x is a vector. If we rewrite the main term as $\|Ax\|_2^2 = x^T A^T A x = x^T \left(A^T A\right) x$, then it can be easily seen, that a solution for above problem is an eigenvector, which corresponds to the largest eigenvalue of matrix $B = A^T A$.

Now, let's return to the objective function (7), more precisely to its alternative[2] :

$$\max_{w} \|X_{avg}w\|_2^2 \qquad (14)$$
$$s.t. \|Yw\|_2^2 = 1$$

where $X_{avg} = X^1_{avg} - X^2_{avg}$. Matrix Y is a block-matrix, composed of matrices $Y^1_l, 1 \leq l \leq L$ and $Y^2_m, 1 \leq m \leq M$ placed one under another; i.e. if matrices Y^1_l and Y^2_m have dimensions of $T-1 \times S$, then a matrix Y will have dimensions of $(L+M)(T-1) \times S$.

We can produce the change of variables in (14) in order to make it look like (13). Let's rewrite the constraint term: $\|Yw\|_2^2 = w^T Y^T Y w$. If matrix Y is a full rank (which is very likely for the noisy data), the matrix $C = Y^T Y$ has a Cholesky factorization[3] $C = U^T U$. Now, the constraint can be written as $\|Yw\|_2^2 = w^T Y^T Y w = w^T U^T U w$. If we introduce a new variable $x = Uw$ ($w = U^{-1}x$), then the objective (14) can be written as:

$$\max_{x} \left\|X_{avg}U^{-1}x\right\|_2^2 \qquad (15)$$
$$s.t. \|x\|_2^2 = 1$$

which exactly resembles the problem (13). Thus we know, that a solution of (15) is an eigenvector ν_{max}, which corresponds to the largest eigenvalue, λ_{max}, of matrix $B = \left(X_{avg}U^{-1}\right)^T X_{avg}U^{-1} = U^{-1T}X^T_{avg}X_{avg}U^{-1}$. Hence, the solution of (14) is $w = U^{-1}\nu_{max}$.

The important advantage of this approach, is that it doesn't require iterative optimization, but needs only a few simple algebraic steps. However, the difference of objective (14) is that in the TV term, the l_1 norm was replaced by l_2 norm. Thus it doesn't measure *Total Variation* any more. Nonetheless, our simulations shows, that using approach described in this subsection, we achieve similar results, with comparison to those, achieved using objective (10).

[2]Note, that objective (14) is quadratic, quadratically constrained. A TV term also appears with a l_2 norm, and thus can not represent *Total Variation* any more.

[3]If matrix Y is not full rank, we may use a regularization $C = Y^T Y + \alpha I$, where α is a small constant, and I is an identity matrix

4 Theoretically Best Spatial Integration

In some (unpractical) situations, we can evaluate the bound for the best theoretically achievable separations of signal from noise, using spatial filtering. This can be done if the 'sensor measurement' data is synthesized in the following way:

$$X = s_{art}a^T + N \tag{16}$$

In above formula, the 'sensor measurement' $T \times S$ matrix X is obtained by mixing artificial signal, represented in $T \times 1$ vector s_{art} with $S \times 1$ coupling vector a, and and adding $T \times S$ noise matrix N.

If we know both background noise covariance matrix N and the mixing vector a in (16), we can find the best weighting vector w solving the following problem[4]:

$$\begin{aligned} &\min_w \|Nw\|_2^2 = w^T R w \\ &s.t. \left\|s_{art}a^T w\right\|_2^2 = 1 \end{aligned} \tag{17}$$

where $R = N^T N$ is the noise covariance matrix.

The artificial signal s_{art} can be normalized $\|s_{art}\|_2 = 1$, hence the constraint in (17) can be simplified:

$$\begin{aligned} &\min_w w^T R w \\ &s.t.\ a^T w = 1 \end{aligned} \tag{18}$$

Note, that in above equation, we want to find w, which maximally suppress the noise, while keeping the norm of unmixing vector constant. And one can state, that this task differs from the task of the objective function (10). But actually in both cases we want to perform the de-noising of the signal of interest. In this sense, the solution of (18) can be viewed as theoretically best achievable limit and thus a good reference point for comparison.

We can solve the problem (18) using Lagrange Multipliers:

$$\min_w w^T R w - \lambda \left(a^T w - 1\right)$$

The gradient of above objective is given by: $g(w) = Rw - \lambda a^T$. The solution is obtained if $g(w) = 0 \ \Rightarrow w_{th} = \lambda R^{-1} a$. Note, that w_{th} can be found up to

[4]Note, that we want to perform the de-noising by weighted sum of channels. Thus, the idea of subtracting the noise matrix is not relevant.

scaling and sign, hence every real $\lambda \neq 0$ can be chosen. If we take $\lambda = 1$ we get the final formula for w_{th}:

$$w_{th} = R^{-1}a \tag{19}$$

5 *Time-Domain* Filtering

Signals, reconstructed by one of the spatial filtering methods, will still suffer from noise contamination, which can be further reduced by the second stage of preprocessing - *time-domain* filtering of the estimated signals (2).

The problem in applying filtering is that we do not know in advance which filter to use, because the signals of interest as well as background activity noise are unknown. Thus, we propose to find a suitable filter by learning, based on the same criteria used for finding spatial filter: maximize between class discrimination, while keeping the resulting signal smooth.

So, we want to find filter $h[n]$, $1 \leqslant n \leqslant N_{filt}$, which will further discriminate between reconstructed signals $\hat{s}^1_{avg}[n]$ and $\hat{s}^2_{avg}[n]$:

$$\max_{h[n]} \left\| \left(\hat{s}^1_{avg}[n] - \hat{s}^2_{avg}[n]\right) * h[n] \right\|_2^2 \tag{20}$$

where $*$ denotes convolution.

Since, we are working with discrete time, time-limited signals, lets define $\tilde{X}^1_{avg}$ to be $(T - N_{filt} + 1) \times N_{filt}$ matrix, j-th column of which contains $\hat{s}^1_{avg}[n]$, $j \leqslant n \leqslant (T - N_{filt} + j)$, e.i. in j-th column of $\tilde{X}$ there is a shifted by $(j-1)$ signal $\hat{s}^1_{avg}[n]$, which is also truncated by $(j-1)$ taps at the beginning and $(N_{filt} - j)$ taps at the end, in order to be $(T - N_{filt} + 1)$ taps in the length. In the same manner we can define matrix $\tilde{X}^2_{avg}$, columns of which will contain shifted replicas of $\hat{s}^2_{avg}[n]$. And finally, matrix $\tilde{X}_{avg} = \tilde{X}^1_{avg} - \tilde{X}^2_{avg}$.

Now, one can easily notice, that the following expression:

$$\max_{w_{filt}} \left\| \tilde{X}_{avg} w_{filt} \right\|_2^2 \tag{21}$$

is equivalent to equation (20) under following conditions:
$w_{filt}[n] = h[N_{filt} + 1 - n]$, and convolution in (20) is also appropriately truncated to be $(T - N_{filt} + 1)$ taps in the length. In this manner, we've succeeded in representing a convolution in an matrix form.

Now, let's continue with matrix manipulations, and represent TV term in the matrix form as well. Let's define *signle trial* matrices $\tilde{X}_i^1$ and $\tilde{X}_i^2$, exactly in the same manner as $\tilde{X}_{avg}^1$ and $\tilde{X}_{avg}^2$, using $\hat{s}_i^1[n]$ and $\hat{s}_j^2[n]$ (defined by (2)) instead of $\hat{s}_{avg}^1[n]$ and $\hat{s}_{avg}^2[n]$ respectively.

We will continue, by definition of

$$\tilde{Y}_l^1(t,i) = \tilde{X}_l^1(t+1,i) - \tilde{X}_l^1(t,i)\ ,\ 1 \le t \le T-1$$

$$\tilde{Y}_m^2(t,i) = \tilde{X}_m^2(t+1,i) - \tilde{X}_m^2(t,i)\ ,\ 1 \le t \le T-1$$

which closely resembles matrices Y_l^1 and Y_m^2 defined under (6).

Finally, we introduce a matrix $\tilde{Y}$, which is a block-matrix, composed of matrices $\tilde{Y}_l^1, 1 \le l \le L$ and $\tilde{Y}_m^2, 1 \le m \le M$ placed one under another; i.e. if matrices $\tilde{Y}_l^1$ and $\tilde{Y}_m^2$ have dimensions of $(T - N_{filt}) \times N_{filt}$, then a matrix $\tilde{Y}$ will have dimensions of $(L+M)(T - N_{filt}) \times N_{filt}$.

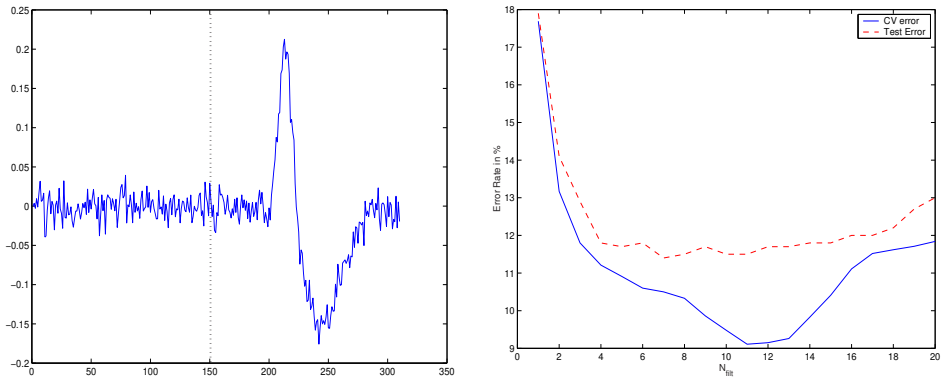
After those preparations, we can represent a problem of *time-domain* filtering in an familiar form:

$$\begin{aligned} &\max_{w_{filt}} \left\| \tilde{X}_{avg} w_{filt} \right\|_2^2 \\ &s.t. \left\| \tilde{Y} w_{filt} \right\|_2^2 = 1 \end{aligned} \tag{22}$$

which exactly resemble the problem (14), already solved in subsection 3. Thus, the solution developed in subsection 3 can be used to solve the problem (22).

There is an alternative choice of the matrix $\tilde{Y}$. It can represent an background activity noise, which we also want to minimize, instead of TV. This idea may be even more appealing, because we minimize the background noise directly, and not some measure of it - Total Variation. This is really a good approach, if the background activity is stationary. The only problem is how to obtain the noise, without desired signal? One of the ways to do it is illustrated in Figure 1(a): if we start to register the response well in advance, then at the beginning we will record an background activity only. Our simulations shows, that an alternative choice of the matrix $\tilde{Y}$ is better for real EEG recordings. On the synthetic data, we've preferred the initial choice - matrix $\tilde{Y}$ containing TV.

Finally, the only open question left, is how to choose the optimal order N_{filt} of FIR filter $h[n]$. We have no closed solution for this issue. In our



(a) (b)

Figure 1: (a) Signal left to the dashed vertical line can be treated as background noise. The actual response appear after the dashed vertical line. (b)Cross Validation error rate for different values of N_{filt}. One can notice, that real test error (dashed line) is highly correlated with CV error. This enables us to choose the optimal order of FIR filter.

experiments, we have chosen its value based on cross validation on the training data: we have calculated the CV error for different values of N_{filt}, and then have chosen the one, which gives the lowest error rate. Figure 1(b) illustrates, that the CV error rate and test error rate are highly correlated.

6 Computational Experiments

In order to show the feasibility of our approaches and make a comparison between them, we've conducted several experiments, using both synthetic and real signals. In artificial data synthesis we've used both white gaussian noise and real EEG signals as a background noise. In this section we provide a results of our simulations.

6.1 Real EEG Signals

In this experiment, we used the data obtained from two BCI competitions, held in years 2002 [17] and 2003 [18]. The goal of above competitions was to validate signal processing and classification methods for Brain Computer Interfaces. Data set consists of single-trials of spontaneous EEG activity,

one part labelled (training data) and another part unlabelled (test data). The goal was to infer labels for the test set, by prepossessing the training data. Inferred test labels should have maximally fit the true (but unknown to participants) test labels.

6.1.1 BCI competition 2002

This data set, obtained from public website [17], consists of EEG signals, that were recorded from one subject in sessions with few minute's breaks in between. The subject was sitting in a normal chair, relaxed arms resting on the table, fingers in a standard typing position at the computer keyboard (index fingers at 'f', 'j' and little fingers at 'a',';'). The task was to press two chosen keys with the corresponding fingers in a self-chosen order and timing ('self-paced key typing'). A total of 516 keystrokes was done at an average speed of 1 key every 2.1 seconds. Brain activity was measured with 27 Ag/AgCl electrodes at 1000 Hz using a band-pass filter from 0.05 to 200 Hz.

Further, windows 1500 ms long were cut out of the continuous raw signals each ending at 120 ms **before** the respective keystroke. The reason for choosing the endpoint at -120 ms is that before this point the classification based on measuring EMG activity only is still close to chance. 100 trials equally spaced over the whole experiment were defined to be the test set, leaving 413 labelled trials for training.

We used the training data for preprocessing. We've applied all our methods described in previous sections - w_{opt}, w_{EVD} and w_{filt}. We've tried several classifiers for classification (we used "pr-tools" classification toolbox, that can be obtained from the public web site [19]), including nearest mean, k nearest neighbor (k-nn) [20] and Support Vector Machines (SVM) with different kernels [21]. All classifiers have provided similar results. Under these circumstances, we've preferred to use the simplest one - nearest mean classifier. The result of classification error, both of *10-fold Cross-Validation* [22] and test error[5], are summarized in Table 1. The best result reported by competition organizers was 4% error rate.

[5]Test error was calculated when real labels were published by competition organizers.

	w_{opt}	w_{EVD}	w_{filt}	simple sum	best sensor
10-fold CV	15%	15%	14%	51%	40%
Test	9%	9%	7%	50%	47%

Table 1: Classification results (error rate in %) of BCI competition 2002 data set. Each column corresponds to a different method of preprocessing. Both*10-fold Cross-Validation* and test error results are provided.

	w_{opt}	w_{EVD}	w_{filt}	simple sum	best sensor
10-fold CV	27%	27%	27%	43%	37%
Test	26%	27%	26%	39%	33%

Table 2: Classification results (error rate in %) of BCI competition 2003 data set. Each column corresponds to a different method of preprocessing. Both*10-fold Cross-Validation* and test error results are provided.

6.1.2 BCI competition 2003

The data set for that competition [18] is similar to the previous one (self-paced key typing). This time the average typing rate was 1 key per second. Totally, there are 416 epochs of 500 ms length each ending 130 ms before a key press. 316 epochs are labelled (training set), the remaining 100 epoches are unlabelled (test set).

This data set was harder for classification than a previous one, because provided epochs are shorter (only 500 ms) and are cut earlier (130 ms) prior to keystroke. As result, all our methods (as well as results reported by organizers) showed higher error rates.

We used the training data for preprocessing, trying out all proposed approaches - w_{opt}, w_{EVD} and w_{filt}. Again, we used nearest mean classifier for classification. The result of classification error of *10-fold Cross-Validation* and test error are summarized in Table 2. The best result reported by competition organizers was error rate of 16%.

6.1.3 Response to Visual Stimuli

For this experiment we've used the data, that was recorded in the laboratory for Evoked Potentials in the Technion - Israel Institute of Technology. We have not conducted the new experiment for our purposes, but rather we have used the data that were already recorded for some other research [23]. We

	w_{opt}	w_{EVD}	w_{filt}	simple sum	best sensor
NM	7.7%	4.8%	4.8%	39.2%	30.2%
k-nn	2.2%	3.3%	2.8%	27.8%	18.9%
SVM	5.9%	4.6%	4.9%	50%	50%

Table 3: *10-fold Cross-Validation* error rate (in %) on Visual Stimuli data set. Each column corresponds to a different methods of preprocessing. Results of applying 3 different classifiers are shown: Nearest Mean (NM), k-nn (with k=3) and SVM with exponential kernel.

are grateful to Hillel Pratt for providing us with these EEG recordings.

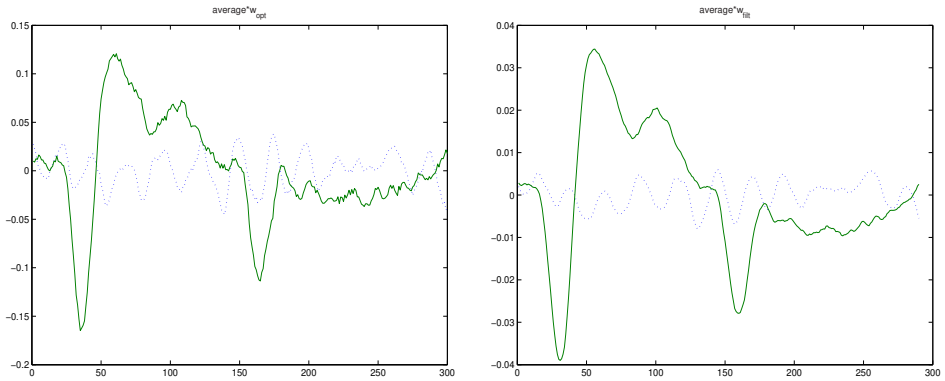
This EEG data was obtained during the following procedure. The subject was shown a sequence of 3 different images, at some predefined and constant over-trials pace. After the sequence of three images was shown, the subject had to respond, by pressing a button. The trials were repeated with periodicity of 7 seconds. The delay between the first and the second images in the sequence was 1.5 seconds, and 2.5 seconds between the second and the third images. Then, the subject was given 3 seconds for respond. Each session consisted of approximately 30 trials. There were several sessions, with some minutes break in between. The EEG data was recorded by 23 electrodes, with sampling rate of 256 samples per second.

In this experiment, we were interested in distinguishing a response to visual stimuli from the absence of response, i.e. regular background activity. We have built two classes of signals from the row data: the first class represented the response to visual stimuli (image was shown), and the second class represented the absence of visual stimuli (regular background activity). In order to build the first class, we've cut from the row data the segments, which start at the times when the first image is shown and are 300 time samples in length. The second class was built from segments, started 300 time samples before the third image is shown, and ended exactly at the time when the third image is displayed.

We have divided the data into the training and the test sets. Then, we have applied all our approaches. This time, the k-nn and SVM with exponential kernel classifiers demonstrated considerably better performance, with respect to nearest mean classifier. The results of 10-fold cross validation are summarized in the Table 3. The test error is provided in Table 4. Reconstructed signals are shown in Figure 2.

	w_{opt}	w_{EVD}	w_{filt}	simple sum	best sensor
NM	15%	10%	10%	40%	35%
k-nn	8.3%	5%	3.3%	35%	20%
SVM	6.7%	5%	3.3%	50%	41.7%

Table 4: Test error rate (in %) on Visual Stimuli data set. Each column corresponds to a different methods of preprocessing. Results of applying 3 different classifiers are shown: Nearest Mean (NM), k-nn (with k=3) and SVM with exponential kernel.



(a) (b)

Figure 2: Experiment with visual stimuli data set. One can easily tell, that the dotted line represents the signal with background activity only, while the solid line corresponds to the signal which contains the response. (a) - signals reconstructed by w_{EVD}; (b) - signals reconstructed by w_{EVD} and further filtered by w_{filt}

6.2 Artificial Signals

We decided not to stop the evaluation of our approach after two previous experiments, and generated synthetic data set, which should resemble the real experiments. The idea was to mix some smooth signal (two different signals, s^1_{art} and s^2_{art}, for two different classes) into background noise N.

We modelled our signals to be T time samples in length. Thus, artificial signals, s^1_{art} and s^2_{art}, are $T \times 1$ vectors. Moreover, we assumed S channel data, hence the dimensions of single trial and noise matrices (X_i and N_i respectfully) are $T \times S$. The weights, with which the artificial signal s_{art} reaches each channel (column of X_i), were randomly chosen and organized in $S \times 1$ *mixing vector* a. This leads to the following data synthesis model:

$$\begin{aligned} X^1_l &= s^1_{art} a^T + N_l \\ X^2_m &= s^2_{art} a^T + N_m \end{aligned} \tag{23}$$

6.2.1 White Gaussian Background Noise

In this experiment, the noise matrix N_i was generated as white gaussian noise (150×25 matrix, generated independently for each trial). The mixing vector m was randomly generated (each element in m is uniformly distributed between $[-1; 1]$). This experiment setup stands up for "real" problem of 25 sensors and 150 time samples in each trial. We have generated 1200 trials. First 200 trials (approximately 100 of each class) were used for preprocessing (finding unmixing vector w). Remaining 1000 trials were used for classification by the nearest mean algorithm [22].

The preprocessing was done by one of our methods - w_{opt}, w_{EVD} and w_{filt}. We have compared results of classification of data preprocessed by different methods. In addition we have compared our results with unprocessed data (simple sum over all sensors) and choosing the best sensor. Fortunately, in the artificial data experiments, we have the good reference point for comparison - results obtained by applying w_{th} (19). As shown in Section 4, this method serves as an upper bound of signal de-noising by weighted sum. Classification results (by Nearest Mean Classifier) are displayed in the Table 5. Three rows refer to three different SNR[6] of artificial signal.

[6]SNR refer to average signal-to-noise ratio at each sensor in single trial. Since mixing weights m_i of artificial signal s_{art} are randomly generated, we've taken the average value of $m_i = 0.5$

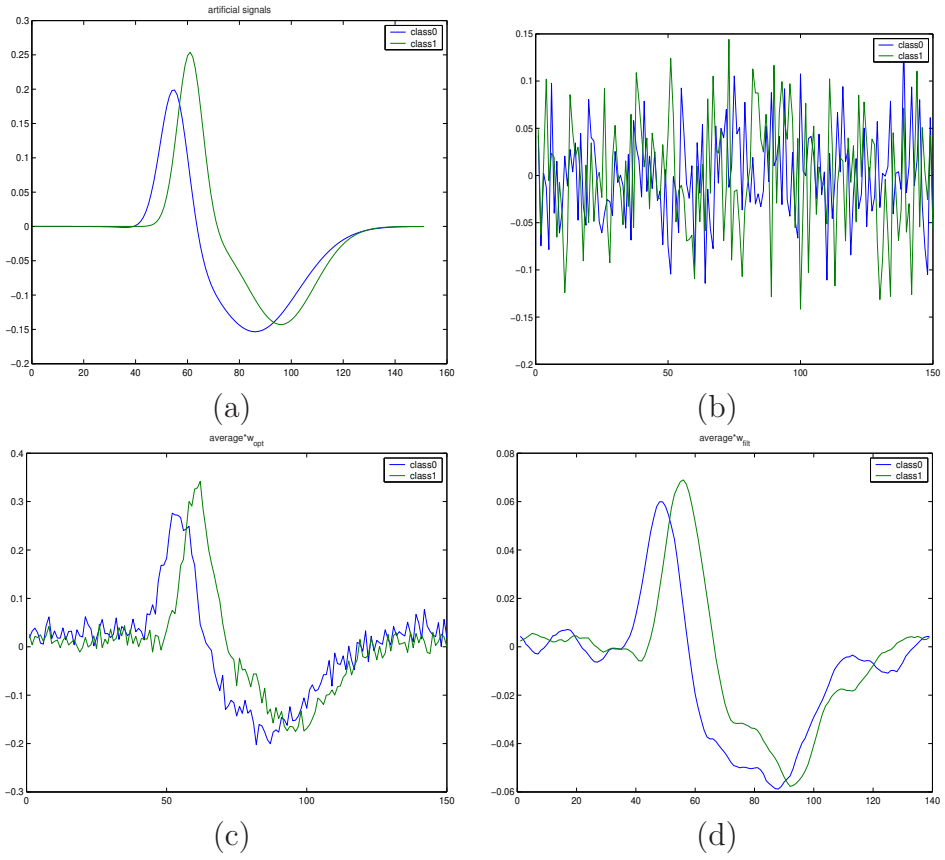


Figure 3: Experiment with artificial signals and Random Gaussian background noise: (a) - artificial signals; (b) - signals restored by simple sum of channels; (c) - signals restored by w_{opt}; (d) - signals restored by w_{opt} and further filtered by w_{filt}.

	w_{opt}	w_{EVD}	w_{filt}	w_{th}	sum of sensors	best sensor
SNR=-10dB	0.0%	0.0%	0.0%	0.0%	43.7%	14.0%
SNR=-15dB	3.5%	3.3%	1.4%	2.7%	44.1%	36.7%
SNR=-20dB	20.3%	20.4%	13.7%	17.1%	49.0%	41.7%

Table 5: Experiment with artificial signals and White Gaussian background noise: Classification Error Rate in % . The first column shows an average SNR, measured at each sensor.

	w_{opt}	w_{EVD}	w_{filt}	w_{th}	sum of sensors	best sensor
SNR=-20dB	2.5%	1.3%	1.1%	0.0%	51.6%	43.3%
SNR=-25dB	10.5%	10.3%	10.0%	0.7%	50.5%	50.1%
SNR=-30dB	25.7%	22.6%	21.9%	1.3%	52.9%	49.5%

Table 6: Experiment with artificial signals and real EEG recordings as a background noise: Classification Error Rate in %. The first column shows an average SNR, measured at each sensor.

6.2.2 Real EEG Background Noise

In the second experiment, we have used real EEG signals as the background noise N (150×22 matrix for each trial). The rest of the setup is identical to the previous case. Classification results are displayed in the Table 6.

7 Conclusions

We've presented an two-stage preprocessing algorithm. It extracts the desired response from multi-channel data, by means of spatial integration in the first stage, and time-domain filtering at the second stage. This preprocessing is essential for the classification. Our experiments shows, that the miss-classification rate achieved on the preprocessed data is significantly lower, than an error rate obtained by classifying unprocessed signals (simple sum of channels, best channel). In addition, in our simulations on synthetic data we show, that the error rate, achieved after the first stage of preprocessing, reaches (or is very close to) the lower bound, developed for spatial integration methods. Moreover, if we apply the second stage of proposed algorithm - time-domain filtering, we receive the error rate even lower than an above bound.

References

[1] H. Berger, "Uber das electrenkephalogramm des menchen," *Arch Psychiat Nervenkr*, vol. 87, pp. 527–570, 1929.

[2] J. Vidal, "Toward direct brain-computer communication," *Annu Rev Biophys Bioeng*, pp. 157–180, 1973.

[3] J. Kalcher, D. Flotzinger, C. Neuper, S. Golly, and G. Pfurtscheller, "Graz brain-computer interface ii: Toward communication between humans and computers based on online classification of three different EEG patterns," *Med Biol Eng Comp*, vol. 34, pp. 383–388, 1996.

[4] G. Pfurtscheller, C. Neuper, D. Flotzinger, and M. Pregenzer, "EEG-based discrimination between imagination of right and left hand movement," *Electroenceph. clin. Neurophysiology*, vol. 103, no. 6, pp. 642–651, 1997.

[5] C. Anderson, E. Stolz, and S. Shamsunder, "Multivariate autoregressive models for classification of spontaneous electroencephalogram during mental tasks," *IEEE Transactions on Biomedical Engineering*, vol. 45, no. 3, pp. 277–286, 1998.

[6] B. Obermaier, C. Neuper, C. Guger, and G. Pfurtscheller, "Information transfer rate in a five-classes brain-computer interface," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 9, no. 3, pp. 283–288, 2001.

[7] J. R. Wolpaw, D. J. McFarland, D. J. Neat, and C. A. Forneris, "An EEG-based brain-computer interface for cursor control," *Clinical Neurophysiology*, vol. 78, no. 3, pp. 252–259, 1991.

[8] A. Kubler, B. Kotchoubey, T. Hinterberger, N. Ghanayim, J. Perelmouter, M. Schauer, C. Fritsch, E. Taub, and N. Birbaumer, "The thought translation device: a neurophysiological approach to communication in total motor paralysis," *Experimental Brain Research*, vol. 124, pp. 223–232, 1999.

[9] J. R. Wolpaw, D. Flotzinger, G. Pfurtscheller, and D. F. McFarland, "A timing of EEG-based cursor control," *Clinical Neurophysiology*, vol. 146, pp. 529–538, 1997.

[10] N. Birbaumer, N. Ghanayim, T. Hinterberger, I. Iversen, B. Kotchoubey, A. Kubler, J. Perelmouter, E. Taub, and H. Flor, "A spelling device for the paralysed," *Nature*, vol. 398, pp. 297–298, 1999.

[11] B. Blankertz, G. Curio, and K.-R. Müller, "Classifying single trial EEG: Towards brain computer interfacing," in *Advances in Neural Information Processing Systems* (S. B. T.G. Dietterich and Z. Ghahramani, eds.), vol. 14, (Cambridge, MA), pp. 157–164, MIT Press, 2002.

[12] J. Müller-Gerking, G. Pfurtscheller, and H. Flyvbjerg, "Designing optimal spatial filters for single-trial EEG classification in a movement task," *Clinical Neurophysiology*, vol. 110, pp. 787–798, 1998.

[13] L. Parra, C. Alvino, A. Tang, B. Pearlmutter, N. Yeung, A. Osman, and P. Sajda, "Linear spatial integration for single trial detection in encephalography," *NeuroImage*, vol. 17, no. 1, pp. 223–230, 2002.

[14] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1998.

[15] M. Zibulevsky and B. A. Pearlmutter, "Blind separation of sources with sparse representations in a given signal dictionary," in *International Workshop on Independent Component Analysis and Blind Signal Separation*, (Helsinki, Finland), June 19–20 2000.

[16] M. Zibulevsky, P. Kisilev, Y. Y. Zeevi, and B. A. Pearlmutter, "Blind source separation via multinode sparse representation," in *Advances in Neural Information Processing Systems 12*, MIT Press, 2002.

[17] P. Sajda, A. Gerson, K.-R. Müller, B. Blankertz, and L. Parra, "A data analysis competition to evaluate machine learning algorithms for use in brain-computer interfaces," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 11, no. 2, pp. 184–185, 2003. Datasets and details about the results available at: http://newton.bme.columbia.edu/competition.htm.

[18] B. Blankertz, K.-R. Müller, G. Curio, T. M. Vaughan, G. Schalk, J. R. Wolpaw, A. Schlögl, C. Neuper, G. Pfurtscheller, T. Hinterberger, M. Schröder, and N. Birbaumer, "The BCI competition 2003: Progress and perspectives in detection and discrimination of EEG single trials," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 1044–1051, 2004. Datasets and details about the results available at: http://ida.first.fraunhofer.de/projects/bci/competition/.

[19] R. P. Duin, "Prtools, a pattern recognition toolbox for matlab." Pattern Recognition Group, Delft University of Technology, 2000. Download from http://www.prtools.org/.

[20] T. Cover and P. Hart., "Nearest neighbor pattern classification," *IEEE Trans. Info. Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[21] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[22] R. Duda, P. Hart, and D. Stork, *Pattern Classification*. New York, USA: John Wiley & Sons, second ed., 2001.

[23] Z. Bigman and H. Pratt, "Time course and nature of stimulus evaluation in category induction as revealed by visual event-related potentials," *Biol. Psychol.*, vol. 66, pp. 99–128, 2004.