RESEARCH ARTICLE

Penalized Multi-Way Partial Least Squares for Smooth Trajectory Decoding from Electrocorticographic (ECoG) Recording

Andrey Eliseyev*, Tetiana Aksenova

CEA-LETI-CLINATEC, Grenoble, France

* eliseyev.andrey@gmail.com

Abstract

In the current paper the decoding algorithms for motor-related BCI systems for continuous upper limb trajectory prediction are considered. Two methods for the smooth prediction, namely Sobolev and Polynomial Penalized Multi-Way Partial Least Squares (PLS) regressions, are proposed. The methods are compared to the Multi-Way Partial Least Squares and Kalman Filter approaches. The comparison demonstrated that the proposed methods combined the prediction accuracy of the algorithms of the PLS family and trajectory smoothness of the Kalman Filter. In addition, the prediction delay is significantly lower for the proposed algorithms than for the Kalman Filter approach. The proposed methods could be applied in a wide range of applications beyond neuroscience.

Introduction

The Brain-Computer Interface (BCI) is a system for converting the brain's neural activity into commands for external devices [1]. Motor-related BCI aims at improving the quality of life of individuals with severe motor disabilities [2]. Paralyzed or handicapped persons get the opportunity for mental control of wheelchairs, robotic arms, exoskeletons, etc. [3–8]. Besides that, motor-related BCI systems can be applied for rehabilitation purposes [9, 10].

Different methods of acquiring the brain's neural activity are currently used for BCI. Among the non-invasive approaches are electroencephalography (EEG) [11, 12], magnetoencephalography (MEG) [7, 13], functional magnet resonance imaging (fMRI) [14, 15], functional near-infrared spectroscopy (fNIRS) [16, 17], etc. In the case of invasive methods such as electrocorticography (ECoG) [18–22] and microelectrode arrays (ME) recordings [23, 24], the electrodes are implanted under the cranial bones. Whereas ME record intra-cortical neural activity, ECoG electrodes are placed on the surface of the brain. A chronic ECoG-recording implant, appropriate for a long-term use, has recently been reported [25]. Although non-invasive techniques ensure better safety, the invasive methods allow superior quality of the recordings, providing high spatial and temporal resolution of the signals [18]. Decoding of hand's continuous trajectories from subdural and epidural ECoG recordings was reported recently [22–24, 26–29].

In this paper, the task of reconstructing continuous 3D trajectories of the subject's hand from ECoG recordings is considered. This problem was studied in both monkeys and humans [22, 29–31]. A set of approaches for trajectory decoding from neuronal recordings were proposed [22–24, 28]. A common approach consists in extraction of informative features from spatial [32], frequency [33], or/and temporal [34] domains of brain signal. Decoder is identified from the trajectory information (response variable) and neural activity features (exploratory variable). Many standard methods for model identification from experimental data are designed for vector exploratory variables which generally represent only one domain (modality) of analysis [26, 35–37]. If analysis in one domain does not provide satisfactory results, several modalities could be combined sequentially in this paradigm. A tensor-based (Multi-Way) approach is intended for simultaneous treatment of several domains. Multi-electrodes brain signals recording is mapped to the spatial-temporal-frequency space [38] representing single time epoch by a cube (a third-order tensor). All cubes are stored in a tensor of observations (a fourth-order tensor). Tensor (multi-way array) data representation is convenient, since it save natural structure of the data. Detailed review of the tensor data representation as well as the tensor analysis is given in [39]. Multi-Way Analysis is reported as an efficient tool for the decoder identification [22, 40, 41].

The peculiarity of the tensor data analysis is their high dimension. The algorithms of the Partial Least Squares (PLS) regression family are known to be particularly appropriate for high dimension tasks [42]. PLS projects both exploratory and response variables to low dimensional spaces of latent variables. Contrary to the widespread Principal Component Analysis (PCA) [43], which projects only explanatory variables, both explanatory and response variables are taken into account by the PLS for projectors identification. Thus, the PLS provides projectors correlated to the task. However, the generic PLS approach is vector oriented and can be applied only for the tensors unfolded to the vectors (Unfolded PLS) [44]. To exploit the advantages of tensor data representation, generalizations of the PLS for the tensor data were invented (N-Way PLS [45], Higher Order PLS [40], etc.) and were used in BCI systems [40, 46, 47].

The smoothness of the predicted trajectory is an important property of motor-related BCI systems [29, 48, 49]. The particular importance of smoothness was reported for control of such BCI effectors as wheelchairs [50] and cars [51]. There are two main approaches to manage the problem of a smooth prediction. The first one consists in the smoothing post-processing of the predicted trajectories [29]. However, this post-processing increases the system's latency, depending on the required level of smoothness, the decision rate of the system, etc. Another solution consists in identifying the model, for which the smoothness of the predicted trajectories is an intrinsic property. A Kalman Filter (KF) [52] is used in BCI systems to generate smooth trajectories [53, 54]. However, the method is not adjusted for the high dimensional data due to the application of the matrix inversion operation in it. Moreover, the Kalman Filter weights the current prediction with the previous one, which also brings a latency to the system. Minimal system latency is an essential requirement for the comfortable use of real-time BCI systems [55]. Improvement of the prediction smoothness with minimal supplementary delay is the particular goal of the article.

In this paper, a new penalization approach is proposed to improve neural decoding smoothness. By means of penalization, one can impose different penalties related to the particular tasks. The paper novelty resides in two new ways of penalization (new penalization terms), which particularly address solution smoothness. Namely, Sobolev Penalized Multi-Way PLS (SNPLS) and Polynomial Penalized Multi-Way PLS (PNPLS), are proposed. They combine tensor representation of the data with the possibility to control the level of the smoothness of the predicted trajectories. In these methods, the smoothness is not provided by the post-processing or weighting with previous predictions, but is an internal parameter of the identified

model. Thus, no supplementary delay is introduced due to post-processing of the response. In the current paper, a set of state of the art BCI methods, namely Kalman Filter, generic PLS, and Multi-Way PLS (NPLS), are considered and compared with the proposed SNPLS and PNPLS on the set of publicly available ECoG recordings. In addition, response latencies are investigated and compared for the proposed approach and state of the art methods.

The proposed methods could be applied in a wide range of applications beyond neuroscience.

Prediction performance evaluation

Different criteria for performance evaluation have been applied in BCI studies [26, 29, 40, 56]. In this paper, Pearson correlation (r), Root Mean Squares Error (RMSE), and Mean Absolute Error (MAE) are used to compare the methods. A shortcoming of the above criteria is that they do not reflect the smoothness of the predicted trajectory. They are based on the sum of squares/absolute values of residuals which temporal sequence is not taken into account. Mean Absolute Differential Error (MADE) is a way to assess both prediction accuracy and smoothness [29]. Hence MADE is included to the list of criteria for performance evaluation.

The Pearson correlation is a commonly used criterion [22, 26, 30, 46] for the continued-value trajectories decoding,

$$r = \text{correlation}(\mathbf{y}, \hat{\mathbf{y}}), \tag{1}$$

Where $\mathbf{y} = (y(t_1), \ldots, y(t_N))^T = (y_1, \ldots, y_N)^T$ and $\hat{\mathbf{y}} = (\hat{y}(t_1), \ldots, \hat{y}(t_N))^T = (\hat{y}_1, \ldots, \hat{y}_N)^T$ are the observed and predicted trajectories. RMSE represents the L_2-norm distance between the predicted and observed trajectories [22, 29, 40]:

$$\text{RMSE} = \|\mathbf{y} - \hat{\mathbf{y}}\|_2 / \|\mathbf{y} - \bar{\mathbf{y}}\|_2, \quad \|\mathbf{y}\|_2 = \sqrt{\sum_{i=1}^{N} y_i^2}, \quad \bar{\mathbf{y}} = \sum_{i=1}^{N} y_i / N. \tag{2}$$

Less sensitive to the outliers, MAE [29, 56] is based on the L_1-norm distance:

$$\text{MAE} = \|\mathbf{y} - \hat{\mathbf{y}}\|_1 / \|\mathbf{y} - \bar{\mathbf{y}}\|_1, \quad \|\mathbf{y}\|_1 = \sum_{i=1}^{N} |y_i|. \tag{3}$$

MADE criterion compares derivatives $\mathbf{y}'$ and $\hat{\mathbf{y}}'$:

$$\text{MADE} = \|\mathbf{y}' - \hat{\mathbf{y}}'\|_1 / \|\mathbf{y}' - \bar{\mathbf{y}}'\|_1, \quad \bar{\mathbf{y}}' = \sum_{i=1}^{N} y_i' / N. \tag{4}$$

MADE characterizes the prediction smoothness [29] and, being based on the L_1-norm, is robust to the outliers.

Methods

Kalman Filter

The KF [52, 57] is a vector-oriented method, which recursively operates on streams of input data to produce an estimate of the underlying system state. KF is widely used to decode the neural activity in BCIs [53, 58–61]. One advantage of KF is the smoothness of the prediction in time [53, 59]. Thus, KF was applied for prediction of the hand's trajectory from the brain's neuronal activity, such as ECoG [60] and spikes [53, 61]. Brain activity data $\mathbf{x}_k \in \Re^n$, observed at the moment t_k, are used to predict the system state $\mathbf{y}_k \in \Re^m$ ($m = 9$ represents 3D coordinates, velocity, and acceleration of the hand at the moment t_k). The generative model of the KF is formulated as a linear dependency between the brain activity and system state: $\mathbf{x}_k = \mathbf{H}_k \mathbf{y}_k + \mathbf{q}_k$, where $\mathbf{H}_k \in \Re^{n \times m}$ is a matrix of linear coefficients, and $\mathbf{q}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$ is the zero-mean noise of the observation, $\mathbf{Q}_k \in \Re^{n \times n}$. The model expects that the system states are linearly propagated

in time: $\mathbf{y}_{k+1} = \mathbf{A}_k\mathbf{y}_k + \mathbf{w}_k$, where $\mathbf{A}_k \in \Re^{m \times m}$ is a matrix of linear coefficients, $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{W}_k)$, $\mathbf{W}_k \in \Re^{m \times m}$. If matrices $\mathbf{H}_k = \mathbf{H}$, $\mathbf{A}_k = \mathbf{A}$, $\mathbf{Q}_k = \mathbf{Q}$, and $\mathbf{W}_k = \mathbf{W}$ are constants in time, they can be estimated from the training data $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N) \in \Re^{n \times N}$, $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_N) \in \Re^{m \times N}$ by means of the least squares:

$$\mathbf{A} = \mathbf{Y}_2\mathbf{Y}_1^T(\mathbf{Y}_1\mathbf{Y}_1^T)^{-1}, \ \mathbf{H} = \mathbf{X}\mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T)^{-1}, \ \mathbf{W} = (\mathbf{Y}_2 - \mathbf{A}\mathbf{Y}_1)(\mathbf{Y}_2 - \mathbf{A}\mathbf{Y}_1)^T/(N-1),$$

$$\mathbf{Q} = (\mathbf{X} - \mathbf{H}\mathbf{Y})(\mathbf{X} - \mathbf{H}\mathbf{Y})^T/N, \ \mathbf{Y}_1 = (\mathbf{y}_1, \ldots, \mathbf{y}_{N-1}), \ \mathbf{Y}_2 = (\mathbf{y}_2, \ldots, \mathbf{y}_N). \quad (5)$$

The KF algorithm consists of *a priori* and *a posteriori* steps. In the first step, the method *a priori* estimates the state of the system as $\hat{\mathbf{y}}_k^- = \mathbf{A}\hat{\mathbf{y}}_{k-1}$. In the second step, *a posteriori* estimation $\hat{\mathbf{y}}_k$ is found as a linear combination of the *a priori* estimation and the weighted difference between the observed and predicted signals:

$$\hat{\mathbf{y}}_k = \hat{\mathbf{y}}_k^- + \mathbf{K}_k(\mathbf{x}_k - \mathbf{H}\hat{\mathbf{y}}_k^-). \quad (6)$$

Here $\mathbf{K}_k = \mathbf{P}_k^-\mathbf{H}^T(\mathbf{H}\mathbf{P}_k^-\mathbf{H}^T + \mathbf{Q})^{-1}$ is the gain matrix, $\mathbf{P}_k^- = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{W}$ is the *a priori* estimate error, and $\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H})\mathbf{P}_k^-$ is the *a posteriori* estimate error [52]. More detailed information about the KF can be found in [57].

Generic PLS

A linear regression model between the response variable $\mathbf{y} \in \Re^m$ and the explanatory variable $\mathbf{x} \in \Re^n$ is represented as $E[\mathbf{y}|\mathbf{x}] = \sum_{i=1}^{n} \boldsymbol{\beta}_i x_i = \mathbf{B}\mathbf{x}$, where $\mathbf{B} = (\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_n) \in \Re^{m \times n}$, $\mathbf{x} = (x_1, \ldots, x_N)^T$ [62]. For the data set $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)^T$, $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_N)^T$, the Ordinary Least Squares (OLS) gives [63]:

$$\mathbf{B} = \arg\min_{\mathbf{B}} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{B}\mathbf{x}_i\|_2^2 = \arg\min_{\mathbf{B}} \|\mathbf{Y} - \mathbf{B}\mathbf{X}\|_2^2. \quad (7)$$

However, the OLS procedure is unstable for the case of the high dimension task [63]. Partial Least Squares regression is a statistical method for linear vector-based data analysis [42]. Due to its efficient dimension reduction technique even in the case of highly correlated variables, PLS is particularly appropriate for the high dimensional data. The PLS algorithm iteratively estimates a linear relation between the matrices of observations of input and output variables $\mathbf{X} \in \Re^{N \times n}$ and $\mathbf{Y} \in \Re^{N \times m}$: $\mathbf{Y} = \mathbf{X}\mathbf{B} + \mathbf{D}$, where $\mathbf{B} \in \Re^{n \times m}$ is the matrix of linear coefficients, and $\mathbf{D} \in \Re^{N \times m}$ is the matrix of noise. On each iteration, $\mathbf{X}$ and $\mathbf{Y}$ are projected in low dimension spaces (latent variables) in such a way as to explain the maximum of their variation simultaneously: $\mathbf{X} = \mathbf{T}\mathbf{P}^T + \mathbf{E}$, $\mathbf{Y} = \mathbf{U}\mathbf{Q}^T + \mathbf{F}$, where $\mathbf{T} = [\mathbf{t}_1, \ldots, \mathbf{t}_F] \in \Re^{N \times F}$ and $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_F] \in \Re^{N \times F}$ are the matrix of the latent variables (scores), $\mathbf{P} = [\mathbf{p}_1, \ldots, \mathbf{p}_F] \in \Re^{n \times F}$ and $\mathbf{Q} = [\mathbf{q}_1, \ldots, \mathbf{q}_F] \in \Re^{m \times F}$ are the matrix of the loading vectors (projectors), $\mathbf{E}$ and $\mathbf{F}$ are residual matrices, and F is the number of iterations (factors). Estimation of the factors number F can be done through the cross-validation procedure [64]. A detailed description of the PLS approach can be found in [42].

The PLS regression was applied in the BCI systems for hand trajectory reconstruction [22, 26, 40].

Multi-way PLS

Multi-way (N-way) PLS is a generalization of the PLS method to the tensor input/output data [45]. Tensors (multi-way arrays) allow the representation of data of a high order. Vectors and matrices are tensors of orders one and two, respectively. In this paper, a tensor is denoted as $\underline{\mathbf{X}} \in \Re^{I_1 \times \ldots \times I_n}$, where n is the order of the tensor. More information about the tensors can be

found in [39]. NPLS inherits the robustness of PLS and allows simultaneous analysis of data in several domains (e.g. time-frequency-space). Examples of the application of multi-way tensor-based methods in BCI research are given in [40, 46, 65–67].

Similarly to PLS regression, NPLS builds the linear relation between the dependent and independent tensors of observations $\underline{\mathbf{X}} \in \Re^{N \times I_1 \times \ldots \times I_n}$ and $\underline{\mathbf{Y}} \in \Re^{N \times J_1 \times \ldots \times J_m}$. The model is constructed iteratively by projection of the tensors to the space of latent variables:

$$\underline{\mathbf{X}} = \sum_{f=1}^{F} \mathbf{t}_f \circ \mathbf{w}_1^f \circ \ldots \circ \mathbf{w}_n^f + \underline{\mathbf{E}}, \; \underline{\mathbf{Y}} = \sum_{f=1}^{F} \mathbf{u}_f \circ \mathbf{q}_1^f \circ \ldots \circ \mathbf{q}_m^f + \underline{\mathbf{F}}, \; \mathbf{u}_f = \mathbf{T}_f \mathbf{B}_f, \quad (8)$$

where the operator "∘" is the outer product [39], $\mathbf{T}_f = [\mathbf{t}_1, \ldots, \mathbf{t}_f] \in \Re^{N \times f}$ and $\mathbf{U}_f = [\mathbf{u}_1, \ldots, \mathbf{u}_f] \in \Re^{N \times f}$ are the matrices of the latent variables after $f = 1, \ldots, F$ iterations, $\mathbf{w}_i^f \in \Re^{I_i}$ and $\mathbf{q}_j^f \in \Re^{J_j}$ are the projection vectors, $\mathbf{B}_f$ is the matrix of the linear coefficients, and $\underline{\mathbf{E}}$ and $\underline{\mathbf{F}}$ are the residual tensors.

In contrast to the OLS approach, NPLS, as well as PLS, is a biased estimator. However, it provides the insensitivity of the predictive model to ill-conditioning and noise [68].

Penalized regression

Modification of the cost function (optimization functional) in the model identification procedure could be used to improve the prediction/regression properties. Among other reasons, penalizations are widely applied to achieve desirable characteristics (sparsity, robustness, etc.). Additional terms are added to the optimization task, basically performing the standard L_1-norm or L_2-norm optimization. For linear regression $E[\mathbf{y}|\mathbf{x}] = \mathbf{Bx}$ the least squares estimation is given in (7). To impose the additional restrictions, the regularization (penalization) term is introduced [69–74].

$$\mathbf{B} = \arg\min_{\mathbf{B}} \|\mathbf{Y} - \mathbf{BX}\|_2^2 + P(\mathbf{Y}, \mathbf{B}, \mathbf{X}), \quad (9)$$

$P_{L_1}(\mathbf{B}, \lambda) = \lambda \|\mathbf{B}\|_1^2$ (LASSO) and $P_{L_2}(\mathbf{B}, \lambda) = \lambda \|\mathbf{B}\|_2^2$ (RIDGE) are the most frequently used, $\lambda \geq 0$ represents the penalization parameter. Whereas the P_{L_1} penalization encourages sparse solution (9), the P_{L_2} penalization reduces the variation of the coefficients $\mathbf{B}$, which increases the robustness of the model [71]. In addition, the P_{L_2} penalization as well as Fused LASSO [75] are often applied to improve the smoothness of the coefficients $\mathbf{B}$ [70, 73, 74]. However, neither P_{L_1} nor P_{L_2} penalizations improve the smoothness of prediction of $\mathbf{y}(t)$ analyzing the data flow $\mathbf{x}(t)$. In this paper, we propose two penalizations to improve the smoothness of the continuous trajectories prediction.

SNPLS. The Sobolev NPLS method is based on the use of the Sobolev norm W_s [76] instead of the L_2-norm in the optimization task:

$$W_s \text{ norm}: \; \|\phi\|_{W_s}^2 = \|\phi\|_2^2 + \|\phi^{(s)}\|_2^2. \quad (10)$$

Here ϕ and $\phi^{(s)}$ are a function and its s-th derivation, respectively. Optimization in Sobolev space corresponds to the optimization problem (9), where the penalization term is added with the parameter $\lambda \geq 0$ to control the influence of the derivative part:

$$\mathbf{B} = \arg\min_{\mathbf{B}} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{Bx}_i\|^2 + \lambda \sum_{i=2}^{N} \|\mathbf{Bx}^{(s)}_i\|^2. \quad (11)$$

The problem can be represented in the matrix form:

$$\mathbf{B} = \arg\min_{\mathbf{B}} \|\mathbf{Y} - \mathbf{BX}\|^2 + \|\sqrt{\lambda}\mathbf{BX}^{(s)}\|^2 = \arg\min_{\mathbf{B}} \|\widetilde{\mathbf{Y}} - \mathbf{B}\widetilde{\mathbf{X}}\|^2, \qquad (12)$$

where $\widetilde{\mathbf{X}} = \begin{bmatrix} \mathbf{X} \\ \sqrt{\lambda}\mathbf{X}^{(s)} \end{bmatrix}$, $\widetilde{\mathbf{Y}} = \begin{bmatrix} \mathbf{Y} \\ \mathbf{0} \end{bmatrix}$.

The optimization problem (12) corresponds to the regression estimation task with the matrices of independent and dependent variables $\widetilde{\mathbf{X}}$ and $\widetilde{\mathbf{Y}}$, respectively. Moreover, for the tensor case, the same approach can be applied to obtain the tensors $\underline{\widetilde{\mathbf{X}}}$ and $\underline{\widetilde{\mathbf{Y}}}$. To identify the regression between the tensors $\underline{\widetilde{\mathbf{X}}}$ and $\underline{\widetilde{\mathbf{Y}}}$, we use the NPLS algorithm, since it is suitable for tensor-based variables and has demonstrated its efficiency for BCI tasks.

PNPLS. Polynomial Penalized NPLS considers the optimization problem:

$$\mathbf{B} = \arg\min_{\mathbf{B}} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{Bx}_i\|^2 + \lambda \sum_{i=l+1}^{N-l} \|g_{p,l}(\mathbf{x}_i, \mathbf{B})\|^2,$$

$$g_{p,l}(\mathbf{x}_i, \mathbf{B}) = \mathbf{Bx}_i - \mathbf{B}\begin{pmatrix} P_{p,(x_{i-l}^1,\ldots,x_{i+l}^1)}(x_i^1) \\ \cdots \\ P_{p,(x_{i-l}^n,\ldots,x_{i+l}^n)}(x_i^n) \end{pmatrix} = \mathbf{Bx}_i - \mathbf{B}\mathbf{P}_{p,(\mathbf{x}_{i-l},\ldots,\mathbf{x}_{i+l})}(\mathbf{x}_i) = \mathbf{Bx}_i - \mathbf{B}\widehat{\mathbf{x}}_i, \quad (13)$$

where $\mathbf{x}_i = (x_i^1, \ldots, x_i^n)^T$, and $P_{p,(x_{i-l},\ldots,x_{i+l})}(\cdot)$ is a polynomial function of the power p, which provides the minimal least squares approximation of the set of points $(x_{i-l},\ldots,x_{i+l})$, $\widehat{\mathbf{x}}_i = \mathbf{P}_{p,(\mathbf{x}_{i-l},\ldots,\mathbf{x}_{i+l})}(\mathbf{x}_i)$. The procedure of $\widehat{\mathbf{x}}$ identification can be represented as a result of the power p polynomial filtration in a sliding window of size $(2l + 1)$, where polynomial coefficients are fitted independently for each window.

In the matrix notation, the optimization problem is represented as

$$\mathbf{B} = \arg\min_{\mathbf{B}} \|\mathbf{Y} - \mathbf{BX}\|^2 + \lambda\|\mathbf{BX} - \mathbf{B}\widehat{\mathbf{X}}\|^2, \qquad (14)$$

Penalizing the difference between the results of application of the model $\mathbf{B}$ to the initial ($\mathbf{X}$) and "smoothed" ($\widehat{\mathbf{X}}$) explanatory variables, this algorithm is looking for the model which provides similar results being applied to the original and "smoothed" data.

For the new variables $\widetilde{\mathbf{X}}$ and $\widetilde{\mathbf{Y}}$:

$$\mathbf{B} = \arg\min_{\mathbf{B}} \|\widetilde{\mathbf{Y}} - \mathbf{B}\widetilde{\mathbf{X}}\|^2, \ \widetilde{\mathbf{X}} = \begin{bmatrix} \mathbf{X} \\ \sqrt{\lambda}(\mathbf{X} - \widehat{\mathbf{X}}) \end{bmatrix}, \ \widetilde{\mathbf{Y}} = \begin{bmatrix} \mathbf{Y} \\ \mathbf{0} \end{bmatrix}. \qquad (15)$$

Similar to (12), the task (15) corresponds to the problem of regression estimation with the matrix of the explanatory and response variables $\widetilde{\mathbf{X}}$ and $\widetilde{\mathbf{Y}}$. It can also be generalized to the case of the tensors $\underline{\widetilde{\mathbf{X}}}$ and $\underline{\widetilde{\mathbf{Y}}}$. The NPLS algorithm is used for the following regression estimation.

Both SNPLS and PNPLS algorithms improve solutions without increasing the complexity of the optimization problem (sum-of-squares optimization).

Application

Evaluation and comparison

The BCI problem of the reconstruction of 3D trajectories of a monkey hand from its ECoG brain activity is considered to evaluate the accuracy and smoothness of the proposed approaches. Two proposed methods (SNPLS and PNPLS) are evaluated and compared to three state-of-the-art methods (Kalman Filter, generic PLS, and tensor-based NPLS).

The publicly available dataset recorded and distributed by the Laboratory for Adaptive Intelligence, BSI, RIKEN (http://neurotycho.org) was used to evaluate the methods. During the experiments, the 3D position of the monkey's right hand was recorded simultaneously with the epidural ECoG signal of the brain [26, 77]. The dataset used in the current paper consists of 20 recordings, corresponding to two Japanese macaques (10 recordings of each monkey), denoted as 'B' and 'C' in the database [22]. The monkeys were trained to receive pieces of food with their right hands. The pieces were demonstrated by the experimenter at random locations at a distance of about 20 cm from the monkeys. The new trial started when the monkey finished eating the previous piece of food. The ECoG data were acquired with 64 electrodes (Blackrock Microsystems, Salt Lake City, UT, USA) implanted in the epidural space of the left hemisphere (see Fig 1). A sampling rate of 1000 Hz per channel was provided. To record the hand trajectories, an optical motion capture system was used
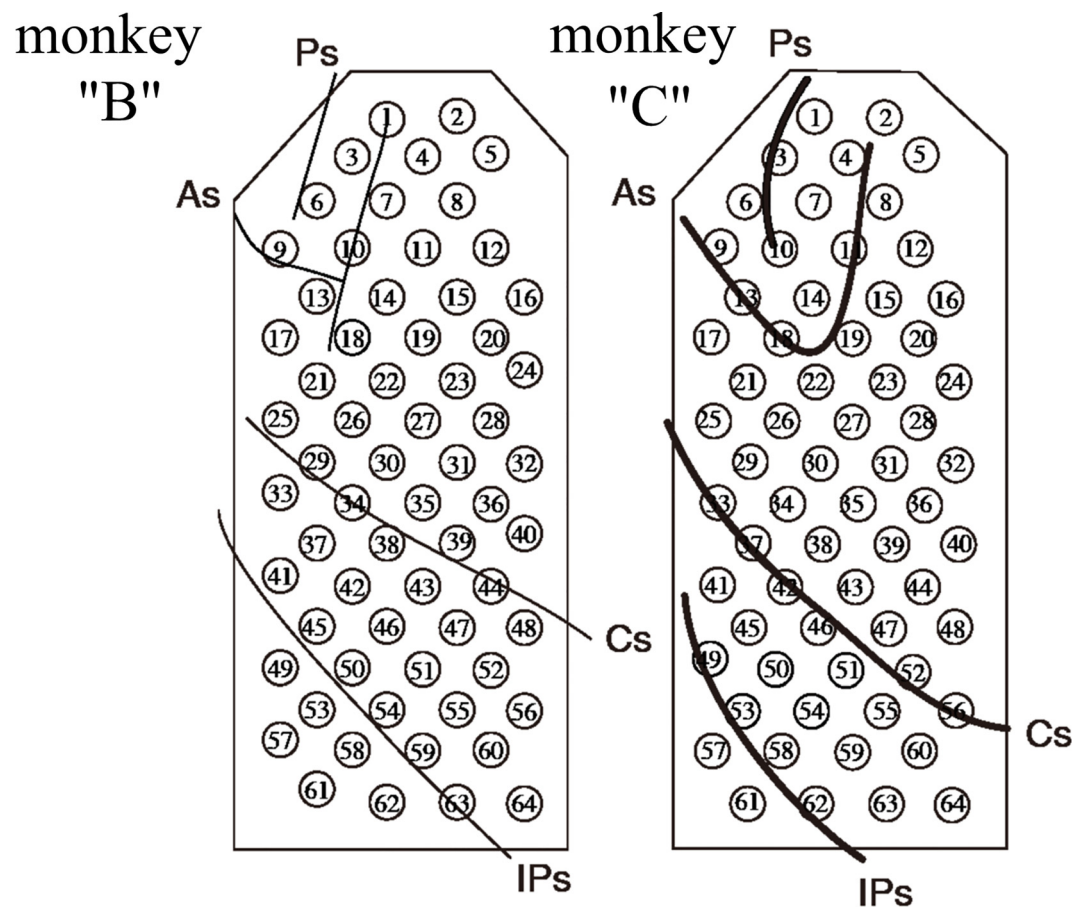

Fig 1. The schemes of the implantations. 64 electrodes were implanted in the epidural space of the left hemispheres of two monkeys. Ps: principal sulcus; As: arcuate sulcus; Cs: central sulcus; IPs: intraparietal sulcus [22].

doi:10.1371/journal.pone.0154878.g001

(Vicon Motion Systems, Oxford, UK) with a sampling rate of 120 Hz. The experiment scheme is represented in Fig 2A. Additional description of the experiments and analyzed data is given in [22, 77].

The duration of each of 20 experiment sessions was 15 minutes. The first 10 minutes of each recording were used for model identification and the last 5 minutes were employed for testing of the corresponding model.

Feature extraction. The input data tensor $\underline{X}$ was formed from the ECoG epochs. Each epoch contains 1 second of the signal, mapped to the temporal-spatial-frequency space by
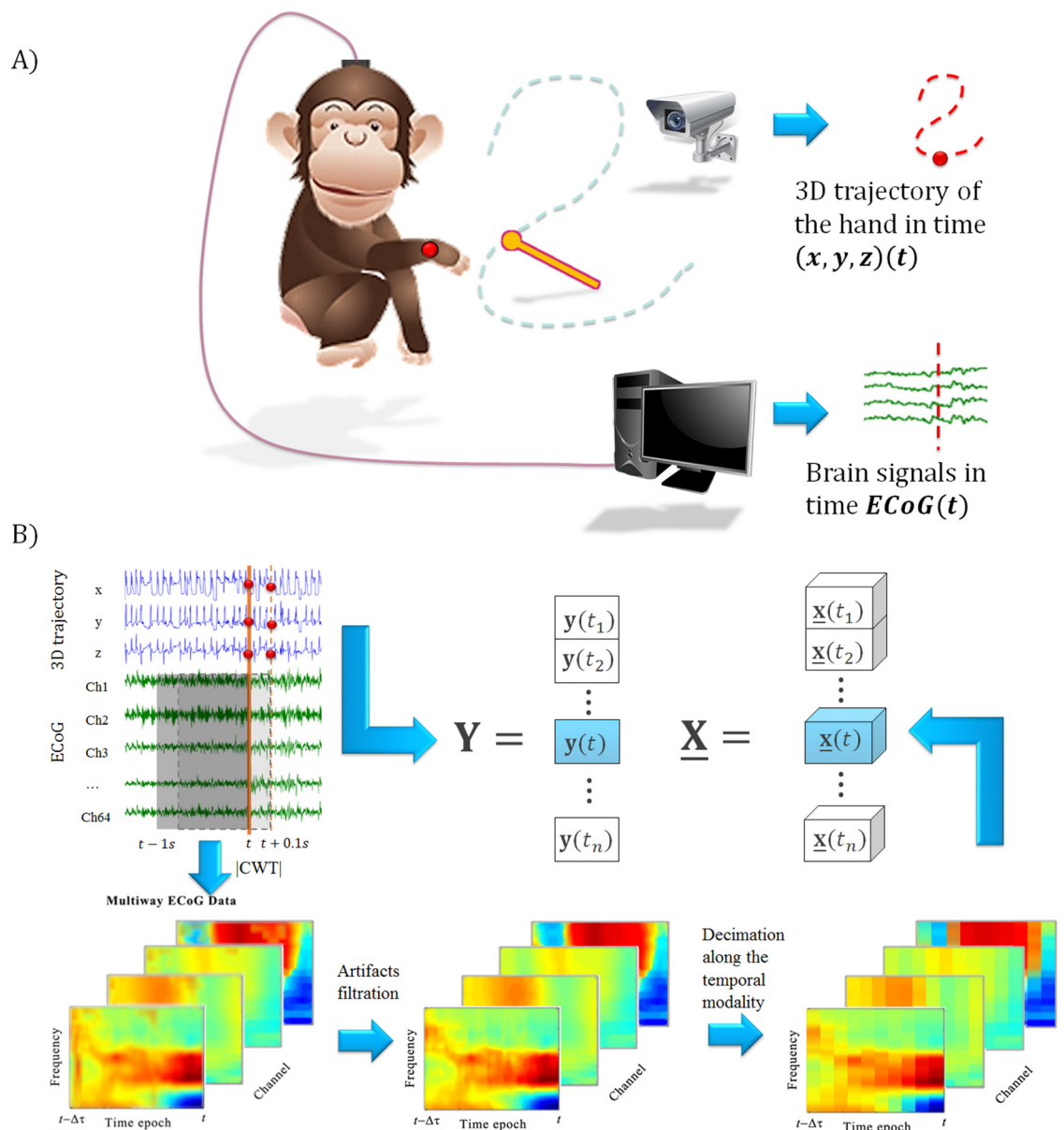
Fig 2. Experimental scheme and data formation. (A) The scheme of the experiment. The monkey is following the food with its hand. 3D coordinates of the hand are recorded simultaneously with monkey ECoG brain activity [22]. (B) For each moment t, the response variable $y(t)$ is formed from the 3D coordinates of the monkey's hand at this moment. To form the explanatory variable $\underline{x}(t)$, the ECoG epoch $[t–\Delta\tau, t]$, $\Delta\tau = 1$ s is mapped by the Continuous Wavelet Transform (CWT). Then, the logarithm of the absolute values of the wavelet coefficients is taken, the procedure of the artifact filtration is applied, and the tensor is decimated along the temporal modality 100 times (1000 points to 10 points). Then the whole procedure is repeated for the next time moment $t_i + 1 = t_i + \Delta t$, $\Delta\tau = 0.1$ s.

doi:10.1371/journal.pone.0154878.g002

means of the wavelet transform. The complex Morlet wavelet was chosen as a mother wavelet due to its widespread application for BCI data analysis [22, 26, 30, 40, 46]. The epochs were taken continuously with time shifts equal to 100 ms. Based on [29], frequencies from 10 to 150 Hz with a step of 10 Hz were chosen. The logarithm of the absolute values of the wavelet coefficients was taken. After formation of the input tensor $\underline{\mathbf{X}}$, the chewing artifacts [22] were filtered in the way described in [29]. Then, the feature tensor was decimated along the temporal modality 100 times, by averaging the data in 10 sliding windows of 100 ms length. A detailed description of the analyzed data formation procedure is given in [29]. The matrix of the response variables $\mathbf{Y}$ was formed from the corresponding 3D coordinates of the monkey's hand. Whereas for PLS, NPLS, SNPLS, PNPLS methods matrix $\mathbf{Y}$ represents 3D coordinates, the Kalman Filter uses the extended matrix of the response variables which additionally includes velocity and acceleration $\mathbf{Y}_{KF} = [\mathbf{Y}, \mathbf{Y}', \mathbf{Y}'']$. The explanatory variables tensor $\underline{\mathbf{X}}$ is identical for all the methods. Fig 2B represents the general scheme of the data preparation procedure.

To validate the methods, the explanatory variables tensor $\underline{\mathbf{X}}$ and the matrix of the response variables $\mathbf{Y}$ were split into training (70%) and test (30%) subsets for each recording:

$$\text{for training data} \begin{cases} \underline{\mathbf{X}}_{\text{training}} \in \Re^{6021 \times 15 \times 10 \times 64}, & \mathbf{Y}_{\text{training}} \in \Re^{6021 \times 3}, & \text{for PLS} - \text{family methods} \\ & [\mathbf{Y}_{\text{training}}, \mathbf{Y}'_{\text{training}}, \mathbf{Y}''_{\text{training}}] \in \Re^{6021 \times 9}, & \text{for KF} \end{cases}_{i=1}^{20}, \quad (16)$$

$$\text{for test data} \begin{cases} \underline{\mathbf{X}}_{\text{test}} \in \Re^{2580 \times 15 \times 10 \times 64}, & \mathbf{Y}_{\text{test}} \in \Re^{2580 \times 3}, & \text{for PLS} - \text{family methods} \\ & [\mathbf{Y}_{\text{test}}, \mathbf{Y}'_{\text{test}}, \mathbf{Y}''_{\text{test}}] \in \Re^{2580 \times 9}, & \text{for KF} \end{cases}_{i=1}^{20}. \quad (17)$$

Parametrization. Three state-of-the-art methods (Kalman Filter, generic PLS, and tensor-based NPLS) and two proposed tensor-based approaches (SNPLS and PNPLS) were compared using a set of optimized parameters. The optimal number of factors F (PLS, NPLS, SNPLS, PNPLS) as well as the smoothing parameter λ (SNPLS, PNPLS) were chosen on the training set by the 10-fold cross-validation procedure [64] for each recording. The preliminary study on one recording gave the polynomic parameters for the PNPLS algorithm: the number of points chosen for the polynomial coefficients estimation was equal to $L = 2l + 1 = 9$ and the polynomic power was $p = 2$. Then, these parameters were fixed for all other recordings. Similarly, the optimal derivative order chosen for the SNPLS approach was $s = 3$.

Results. The models were calibrated separately on each training recording and tested on the corresponding testing data. An example of the application of the compared methods to the same interval of data is illustrated in Fig 3.

Fig 4 presents the criteria for comparing the methods' performance (Pearson correlation, RMSE, MAE, MADE) for the Kalman Filter, PLS, NPLS, SNPLS, and PNPLS, averaged over all the recordings. Table 1 represents the p-values for all the pairs of the methods. In the present study, the significance level of $\alpha = 0.05$ is chosen. The algorithms of the PLS family, and in particular the proposed SNPLS and PNPLS, significantly outperform the KF in correlation, RMSE, and MAE. The difference between the generic methods (PLS, NPLS) and the proposed approaches (SNPLS, PNPLS) for these criteria is insignificant (Table 1) and does not exceed 5% (Fig 4). In general, the presented prediction accuracy corresponds to the accuracy reported for this database [22, 26, 29, 46].

KF, SNPLS and PNPLS demonstrated better smoothness than PLS and NPLS, which is reflected by the MADE criterion: the improvement is significant (Table 1) and varies from 30% to 47% (Fig 4). At the same time, although KF showed the best smoothness, the difference between KF and the proposed PNPLS approach is less than 6% and is insignificant.
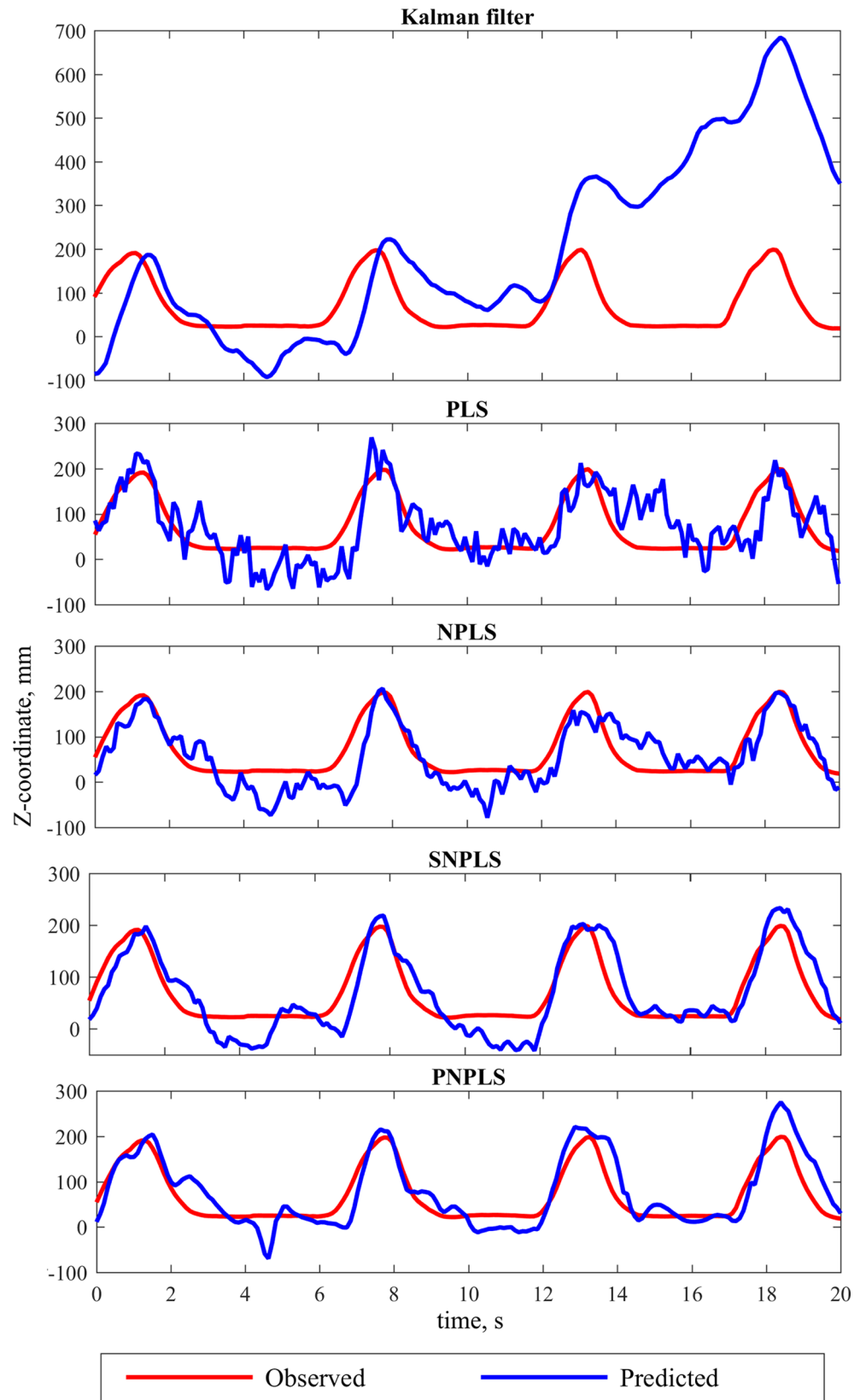
Fig 3. Example of the observed and predicted data. Examples of the observed and predicted (Kalman Filter, PLS, NPLS, PNPLS, and SNPLS) trajectories of Z-coordinate of the monkey's wrist.

doi:10.1371/journal.pone.0154878.g003
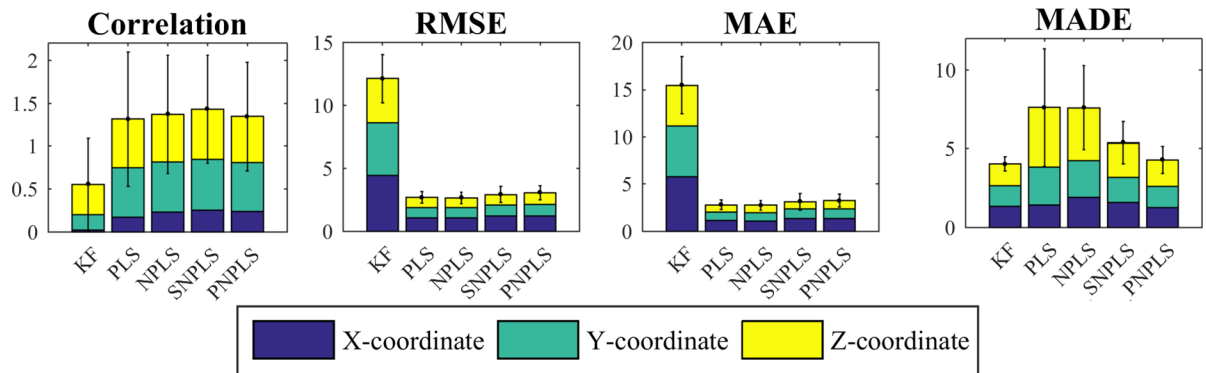
Fig 4. Prediction quality characteristics. The criteria of prediction quality, estimated for 3D trajectories for the models given by KF, PLS, NPLS, SNPLS, and PNPLS. The results are averaged over 20 recordings for two monkeys.

To illustrate PLS-family predictive models, the averaged influence [78] of the frequency, temporal, as well as spatial modalities estimated for the monkey "B" are shown in Fig 5 (the results for the monkey "C" are consistent). For each modality, the influences of its elements are evaluated as the weights of sum of the absolute values of the related predictive model's coefficients.

Prediction delay comparison

System latency is an important characteristic for BCIs [55]. Smoothing generally introduce the delay to the prediction. An additional study was carried out to assess and compare prediction delays of the analyzed methods.

To compare latencies, the predicted trajectories generated by the analyzed approaches were shifted in time to maximize the correlation between predicted and observed trajectories. Prediction delays were estimated as minimal shifts of the predicted signals, which maximizes the correlation between predicted and tested trajectories. The shift interval $S = [0, 2]$ s was studied. Fig 6 shows averaged delays for the analyzed methods over monkeys, sessions, and coordinates. The delay of the Kalman filter is 0.47±0.21 s. Significantly lower ($\alpha = 0.05$) prediction delays are demonstrated by PLS-family algorithms (0.10±0.09 s, 0.11±0.11 s, 0.15±0.12 s, and 0.03 ±0.07 s for PLS, NPLS, SNPLS, and PNPLS, respectively). Differences between the PLS, NPLS, SNPLS, and PNPLS latencies are not significant.

Discussion

Motor-related BCI systems are of great importance since they can help people with severe motor disabilities to recover their functionality. However, decoding the neural activity is a

Table 1. *p*-values of the difference between the quality evaluation criteria of the methods (ANOVA test, significance level $\alpha = 0.05$).

| Compared methods		*p*-value			
		Correlation	RMSE	MAE	MADE
KF	SNPLS	0.000	0.000	0.000	0.000
	PNPLS	0.000	0.000	0.0l00	0.132
PLS	SNPLS	0.456	0.126	0.108	0.002
	PNPLS	0.848	0.007	0.010	0.000
NPLS	SNPLS	0.678	0.093	0.073	0.000
	PNPLS	0.853	0.005	0.005	0.000
SNPLS	PNPLS	0.529	0.409	0.548	0.000
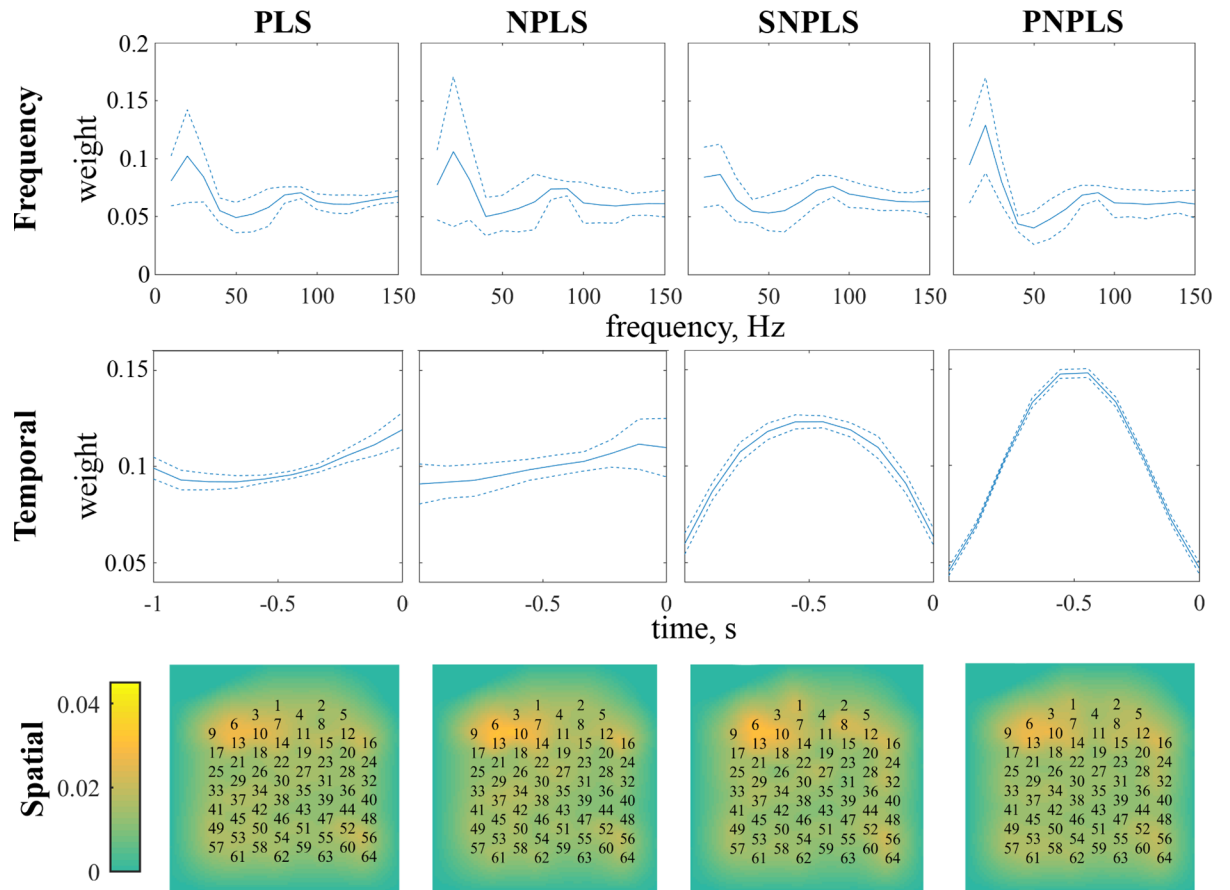
Fig 5. Modalities influences. The average influences of the elements of frequency, temporal, and spatial modalities on the prediction model for PLS-family methods (generic PLS, NPLS, SNPLS, and PNPLS) for monkey "B".

challenging task [1]. BCI control of external devices like wheelchairs, artificial arms, exoskeletons, etc. in real life is especially difficult, due to safety issues [24]. This imposes additional restrictions on BCI decoders, as well as on the smoothness of the predicted trajectories, in particular. The standard approach for improving smoothness consists of post-filtering of the obtained prediction [55, 79]. Such filters (e.g. sliding window averaging) bring a delay to the
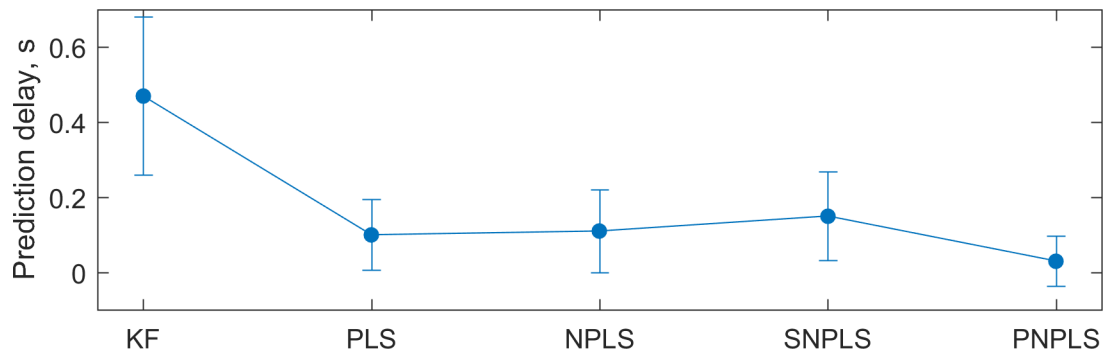
Fig 6. Prediction delays. Averages prediction delays of the Kalman Filter, PLS, NPLS, SNPLS, and PNPLS approaches. PLS, NPLS, SNPLS, and PNPLS significantly outperform KF, whereas the differences between PLS, NPLS, SNPLS, and PNPLS are not significant (significance level $\alpha = 0.05$)

system response. More complex nonlinear filters, such as KF, are labor-consuming, which decreases the decision rate of real-time BCI systems, especially in the case of high dimensional data, and also introduces a prediction delay. Both a high decision rate and minimal system latency are essential requirements for the comfortable use of real-time BCI systems [55].

The methods proposed in the current paper, namely SNPLS and PNPLS, are based on multi-way data analysis, which is known to be efficient for neural data treatment [40, 47, 80]. These methods construct the decoders with the possibility of controlling the smoothness of the predicted trajectories. The proposed algorithms try to improve the smoothness of prediction without increasing the system latency. Penalization integrated into the linear decoder identification procedure can be interpreted, as the penalization of informative but noisy variables. The proposed methods improve solutions stability by penalizing unsmooth predictions without increasing the complexity of the optimization problem (sum-of-squares optimization). The proposed methods do not change the predictive model complexity as well. Solutions remains linear, and the resulted model can be efficiently applied for high dimensional data flow decoding in real-time on standard computer (Intel Dual Core, 3.16 GHz; RAM 3.25 Gb) [80]. The proposed methods could be applied in a wide range of applications beyond neuroscience.

To validate the efficiency of SNPLS and PNPLS for BCI tasks, the algorithms were compared with a set of state-of-the-art approaches. The Kalman Filter has been chosen as a method which generates a smooth prediction and is widely applied in BCIs. However, it contains the inversion of the matrix. This limits the application of the KF for high dimensional tasks [53]. Unlike the KF, the algorithms of the Partial Least Squares family are particularly suited for the high dimensional data and are also often used in BCI systems [22, 46]. Improvement of the prediction smoothness of these methods, with minimal loss of accuracy as well as supplementary delay, was the particular goal of the article. The prediction accuracy of the compared methods was evaluated by means of Pearson correlation, RMSE, and the more robust to outliers MAE criteria. To evaluate the smoothness, the MADE criterion was chosen. It combines the robustness of the L_1-norm and allows assessment of the smoothness.

The algorithms of the PLS family significantly outperform the KF in prediction accuracy estimated with Correlation, RMSE, MAE, whereas the difference among the PLS-based approaches is insignificant. Due to matrix inversion, KF could be more sensitive to high dimensional and noisy explanatory variable in comparison to PLS-family algorithms. Additional pre- and/or post-processing of data or Kalman algorithm's modifications could improve its prediction accuracy for the current task and this is the subject for future researches.

From the smoothness point of view, KF, PNPLS, and SNPLS significantly outperform generic PLS and NPLS. KF demonstrates the best smoothness. However, the results of KF and new methods are comparable. Thus, PNPLS and SNPLS algorithms combined the prediction accuracy of the PLS-family algorithms with the smoothness approaching the KF. In addition, SNPLS and PNPLS provide lower prediction delay in comparison with the KF. The KF approach latency reaches half a second (470±210 ms) that could be noticeable and irritable for subjects. As it was reported in [79] for spikes decoding methods in monkeys experiments, whereas delays up to 200 ms provide minimal influence, every 100 ms of additional delay added about 200 ms to the reaching time. The studies in human also demonstrated that delays about 200 ms slow down reaching tasks and decrease accuracy [81]. For simulated human experiments, increasing of average reaching time by over 5 s due to visual feedback delay of 300 ms is reported in [55]. Contrary to the KF, the PLS, NPLS, SNPLS, and PNPLS methods provide a latency of about 100 ms that is equal to the decision rate of the system. However, additional real-time closed-loop BCI experiments should be carried out in human to clarify the latency's tolerable level.

Generally, criteria values appropriated for the practical applications are derived according to the subject satisfaction in practical experiments. The influence of different parameters such as error magnitude, processing delays, etc. on closed-loop BCI in human was studied, in particular in [55]. The comparison of decoding approaches during the clinical closed-loop BCI trials will be the next step of our research.

Fig 5 demonstrates the modality influence analysis results [78] for the PLS-family methods, in frequency, spatial and temporal domains. In the frequency modality, frequencies influence distributions are similar for all the considered approaches. The distributions have two maximums around 20 Hz and 90 Hz. In the temporal modality, the PLS and NPLS models tend to give close weights to all elements of the analyzing interval, with the maximums near 0 moments. Contrarily, coefficient distribution of the SNPLS and PNPLS models have maximums in the middle of the analyzing intervals. It is possible that weight shapes are responsible for the predicted trajectories smoothness. On the other hand, prediction delay evaluation demonstrated that there is no additional latency due to these weights distributions, as one might expect. Additional study is needed to clarify this effect. In the spatial modality, electrode weights are comparable for all the considered approaches. The most informative electrodes (6, 10, and 13) are the same for all the methods. This could be explained by detecting the readiness potentials, i.e. cortical contribution to the pre-motor planning of volitional movement in the supplementary motor area. Additional information about the readiness potentials could be found in [82].

The influences analysis allows estimating the stability of the prediction models by means of assessment of the weights variability in each modality. Standard deviations through all the modalities are $\sigma_{PLS} = 0.005\pm0.005$, $\sigma_{NPLS} = 0.005\pm0.008$, $\sigma_{SNPLS} = 0.005\pm0.005$, $\sigma_{PNPLS} = 0.004\pm0.005$. The differences of the standard deviations are not significant ($\alpha = 0.05$). Thus, the proposed PNPLS and SNPLS methods keep the robustness of the generic PLS and NPLS approaches.

Comparing the proposed methods, both SNPLS and PNPLS allow improving of the smoothness of the predicted trajectories. The PNPLS prediction smoothness surpasses the SNPLS one, while the prediction latencies for both approaches are comparable. On the other hand, SNPLS provided slightly a better accuracy (Fig 4). The advantage of SNPLS is that it depends on fewer external parameters than PNPLS, thus it is preferable in the case of restricted computation time for model identification.

Limitations of the present study and future work

The main limitation and disadvantage of the proposed approaches is the time-consuming estimation of a set of the external parameters. The smoothness parameter, the polynomial parameters for PNPLS, the derivative order for SNPLS, as well as the number of factors for SNPLS and PNPLS, need to be evaluated. In this study, a time-consuming cross-validation procedure is applied to identify the external parameters of the algorithms. Additional investigation is needed to optimize parameters estimation procedure.

Parametrization and calibration is the offline procedure. Generally, recalibration is not required for ECoG-based BCIs during a long period [80]. Nevertheless, the fast online calibration would be a strong point for the BCI system. Recently, the Recursive NPLS algorithm was proposed [46] for adaptive BCI system calibration. The combination of the SNPLS and PNPLS methods with RNPLS will allow online adjustment of the decoder to the non-stationary brain activity.

As it was reported in [49, 83], methods that demonstrate good performance for offline decoding could be less efficient for online applications, while other methods provide better performance in closed-loop conditions. The next step of the study will be closed-loop application

and the evaluation of decoding approaches, and in particular the application of the algorithms in clinical BCI to calibrate the BCI decoder. The fully-implantable device WIMAGINE® [25] for chronic measurement and wireless transmission of ECoG data, as well as the full body exo-skeleton EMY® [84], are currently developed within the framework of the CEA-LETI-CLINA-TEC® BCI project. Clinical trials of the exoskeleton control by tetraplegic subjects is in preparation.

Acknowledgments

Author Contributions

References

1. Wolpaw JR, Birbaumer N, McFarland DJ, Pfurtscheller G, Vaughan TM. Brain-computer interfaces for communication and control. Clinical neurophysiology: official journal of the International Federation of Clinical Neurophysiology. 2002; 113(6):767–91. PMID: 12048038.

2. Donoghue JP. Bridging the brain to the world: a perspective on neural interface systems. Neuron. 2008; 60(3):511–21. doi: 10.1016/j.neuron.2008.10.037 PMID: 18995827.

3. Daly JJ, Wolpaw JR. Brain-computer interfaces in neurological rehabilitation. Lancet neurology. 2008; 7(11):1032–43. doi: 10.1016/S1474-4422(08)70223-0 PMID: 18835541.

4. Benabid AL, Costecalde T, Torres N, Moro C, Aksenova T, Eliseyev A, et al. Deep brain stimulation: BCI at large, where are we going to? Progress in brain research. 2011; 194:71–82. doi: 10.1016/B978-0-444-53815-4.00016–9 PMID: 21867795.

5. McMullen DP, Hotson G, Katyal KD, Wester BA, Fifer MS, McGee TG, et al. Demonstration of a Semi-Autonomous Hybrid Brain–Machine Interface Using Human Intracranial EEG, Eye Tracking, and Computer Vision to Control a Robotic Upper Limb Prosthetic. Neural Systems and Rehabilitation Engineering, IEEE Transactions on. 2014; 22(4):784–96.

6. Pfurtscheller G, Allison BZ, Brunner C, Bauernfeind G, Solis-Escalante T, Scherer R, et al. The hybrid BCI. Frontiers in neuroscience. 2010; 4.

7. Amiri S, Fazel-Rezai R, Asadpour V. A review of hybrid brain-computer interface systems. Advances in Human-Computer Interaction. 2013; 2013:1.

8. Cao L, Li J, Ji H, Jiang C. A hybrid brain computer interface system based on the neurophysiological protocol and brain-actuated switch for wheelchair control. Journal of neuroscience methods. 2014; 229:33–43. doi: 10.1016/j.jneumeth.2014.03.011 PMID: 24713576

9. Silvoni S, Ramos-Murguialday A, Cavinato M, Volpato C, Cisotto G, Turolla A, et al. Brain-computer interface in stroke: a review of progress. Clinical EEG and Neuroscience. 2011; 42(4):245–52. PMID: 22208122

10. Bermudez i Badia S, Garcia Morgade A, Samaha H, Verschure P. Using a hybrid brain computer interface and virtual reality system to monitor and promote cortical reorganization through motor activity and motor imagery training. Neural Systems and Rehabilitation Engineering, IEEE Transactions on. 2013; 21(2):174–81.

11. Birbaumer N, Ghanayim N, Hinterberger T, Iversen I, Kotchoubey B, Kübler A, et al. A spelling device for the paralysed. Nature. 1999; 398(6725):297–8. PMID: 10192330

12. Wolpaw JR, McFarland DJ, Neat GW, Forneris CA. An EEG-based brain-computer interface for cursor control. Electroencephalography and clinical neurophysiology. 1991; 78(3):252–9. PMID: 1707798

13. Mellinger J, Schalk G, Braun C, Preissl H, Rosenstiel W, Birbaumer N, et al. An MEG-based brain–computer interface (BCI). NeuroImage. 2007; 36(3):581–93. PMID: 17475511

14. Weiskopf N, Mathiak K, Bock SW, Scharnowski F, Veit R, Grodd W, et al. Principles of a brain-computer interface (BCI) based on real-time functional magnetic resonance imaging (fMRI). Biomedical Engineering, IEEE Transactions on. 2004; 51(6):966–70.

15. Sitaram R, Caria A, Veit R, Gaber T, Rota G, Kuebler A, et al. FMRI brain-computer interface: a tool for neuroscientific research and treatment. Computational intelligence and neuroscience. 2007; 2007.

16. Leff DR, Orihuela-Espina F, Elwell CE, Athanasiou T, Delpy DT, Darzi AW, et al. Assessment of the cerebral cortex during motor task behaviours in adults: a systematic review of functional near infrared spectroscopy (fNIRS) studies. NeuroImage. 2011; 54(4):2922–36. doi: 10.1016/j.neuroimage.2010.10.058 PMID: 21029781

17. Fazli S, Mehnert J, Steinbrink J, Curio G, Villringer A, Müller K-R, et al. Enhanced performance by a hybrid NIRS–EEG brain computer interface. NeuroImage. 2012; 59(1):519–29. doi: 10.1016/j.neuroimage.2011.07.084 PMID: 21840399

18. Wang W, Collinger JL, Degenhart AD, Tyler-Kabara EC, Schwartz AB, Moran DW, et al. An electrocorticographic brain interface in an individual with tetraplegia. PloS one. 2013; 8(2):e55344. doi: 10.1371/journal.pone.0055344 PMID: 23405137; PubMed Central PMCID: PMC3566209.

19. Leuthardt EC, Schalk G, Wolpaw JR, Ojemann JG, Moran DW. A brain–computer interface using electrocorticographic signals in humans. Journal of neural engineering. 2004; 1(2):63. PMID: 15876624

20. Schalk G, Leuthardt EC. Brain-computer interfaces using electrocorticographic signals. IEEE reviews in biomedical engineering. 2011; 4:140–54. doi: 10.1109/RBME.2011.2172408 PMID: 22273796.

21. Anderson NR, Blakely T, Schalk G, Leuthardt EC, Moran DW. Electrocorticographic (ECoG) correlates of human arm movements. Experimental brain research. 2012; 223(1):1–10. doi: 10.1007/s00221-012-3226-1 PMID: 23001369.

22. Shimoda K, Nagasaka Y, Chao ZC, Fujii N. Decoding continuous three-dimensional hand trajectories from epidural electrocorticographic signals in Japanese macaques. Journal of neural engineering. 2012; 9(3):036015. doi: 10.1088/1741-2560/9/3/036015 PMID: 22627008.

23. Hochberg LR, Bacher D, Jarosiewicz B, Masse NY, Simeral JD, Vogel J, et al. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. Nature. 2012; 485(7398):372–5. doi: 10.1038/nature11076 PMID: 22596161; PubMed Central PMCID: PMC3640850.

24. Velliste M, Perel S, Spalding MC, Whitford AS, Schwartz AB. Cortical control of a prosthetic arm for self-feeding. Nature. 2008; 453(7198):1098–101. doi: 10.1038/nature06996 PMID: 18509337

25. Mestais C, Charvet G, Sauter-Starace F, Foerster M, Ratel D, Benabid AL. WIMAGINE®: Wireless 64-channel ECoG recording implant for long term clinical applications. IEEE TNSRE. 2015; 23(1).

26. Chao ZC, Nagasaka Y, Fujii N. Long-term asynchronous decoding of arm motion using electrocorticographic signals in monkeys. Frontiers in neuroengineering. 2010; 3:3. doi: 10.3389/fneng.2010.00003 PMID: 20407639; PubMed Central PMCID: PMC2856632.

27. Ifft PJ, Shokur S, Li Z, Lebedev MA, Nicolelis MA. A brain-machine interface enables bimanual arm movements in monkeys. Science translational medicine. 2013; 5(210):210ra154. doi: 10.1126/scitranslmed.3006159 PMID: 24197735.

28. Shin D, Watanabe H, Kambara H, Nambu A, Isa T, Nishimura Y, et al. Prediction of muscle activities from electrocorticograms in primary motor cortex of primates. PloS one. 2012; 7(10):e47992. doi: 10.1371/journal.pone.0047992 PMID: 23110153; PubMed Central PMCID: PMC3480494.

29. Eliseyev A, Aksenova T. Stable and artifact-resistant decoding of 3D hand trajectories from ECoG signals using the generalized additive model. Journal of neural engineering. 2014; 11(6):066005. doi: 10.1088/1741-2560/11/6/066005 PMID: 25341256

30. van Gerven MA, Chao ZC, Heskes T. On the decoding of intracranial data using sparse orthonormalized partial least squares. Journal of neural engineering. 2012; 9(2):026017. doi: 10.1088/1741-2560/9/2/026017 PMID: 22414639.

31. Nakanishi Y, Yanagisawa T, Shin D, Fukuma R, Chen C, Kambara H, et al. Prediction of three-dimensional arm trajectories based on ECoG signals recorded from human sensorimotor cortex. PloS one. 2013; 8(8):e72085. doi: 10.1371/journal.pone.0072085 PMID: 23991046; PubMed Central PMCID: PMC3749111.

32. Rakotomamonjy A, Guigue V, Mallet G, Alvarado V. Ensemble of SVMs for improving brain computer interface P300 speller performances. Artificial Neural Networks: Biological Inspirations–ICANN 2005: Springer; 2005. p. 45–50.

33. Schlögl A, Lee F, Bischof H, Pfurtscheller G. Characterization of four-class motor imagery EEG data for the BCI-competition 2005. Journal of neural engineering. 2005; 2(4):L14. PMID: 16317224

34. Vidaurre C, Krämer N, Blankertz B, Schlögl A. Time domain parameters as a feature for EEG-based brain–computer interfaces. Neural Networks. 2009; 22(9):1313–9. doi: 10.1016/j.neunet.2009.07.020 PMID: 19660908

35. Kayser J, Bruder GE, Tenke CE, Stuart BK, Amador XF, Gorman JM. Event-related brain potentials (ERPs) in schizophrenia for tonal and phonetic oddball tasks. Biological psychiatry. 2001; 49(10):832–47. PMID: 11343680

36. Zhao Q, Zhang L, Cichocki A, Li J, editors. Incremental common spatial pattern algorithm for BCI. Neural Networks, 2008 IJCNN 2008(IEEE World Congress on Computational Intelligence) IEEE International Joint Conference on; 2008: IEEE.

37. Scherer R, Müller GR, Neuper C, Graimann B, Pfurtscheller G. An asynchronously controlled EEG-based virtual keyboard: improvement of the spelling rate. Biomedical Engineering, IEEE Transactions on. 2004; 51(6):979–84.

38. Acar E, Aykut-Bingol C, Bingol H, Bro R, Yener B. Multiway analysis of epilepsy tensors. Bioinformatics. 2007; 23(13):i10–i8. PMID: 17646285

39. Kolda TG, Bader BW. Tensor decompositions and applications. SIAM review. 2009; 51(3):455–500.

40. Zhao Q, Caiafa CF, Mandic DP, Chao ZC, Nagasaka Y, Fujii N, et al. Higher order partial least squares (HOPLS): a generalized multilinear regression method. IEEE transactions on pattern analysis and machine intelligence. 2013; 35(7):1660–73. doi: 10.1109/TPAMI.2012.254 PMID: 23681994.

41. Cong F, Lin Q-H, Kuang L-D, Gong X-F, Astikainen P, Ristaniemi T. Tensor decomposition of EEG signals: a brief review. Journal of neuroscience methods. 2015; 248:59–69. doi: 10.1016/j.jneumeth.2015.03.018 PMID: 25840362

42. Geladi P, Kowalski BR. Partial least-squares regression: a tutorial. Analytica Chimica Acta. 1986; 185 (0):1–17. http://dx.doi.org/10.1016/0003-2670(86)80028-9.

43. Lee JA, Verleysen M. Nonlinear Dimensionality Reduction: Springer Publishing Company, Incorporated; 2007. 310 p.

44. Cichocki A, Zdunek R, Phan AH, Amari S-i. Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation: John Wiley & Sons; 2009.

45. Bro R. Multiway calidration. Multilinear PLS. Journal of Chemometrics. 1996; 10:47–61.

46. Eliseyev A, Aksenova T. Recursive N-way partial least squares for brain-computer interface. PloS one. 2013; 8(7):e69962. doi: 10.1371/journal.pone.0069962 PMID: 23922873; PubMed Central PMCID: PMC3724854.

47. Zhao Q, Zhou G, Adali T, Zhang L, Cichocki A. Kernelization of tensor-based models for multiway data analysis: Processing of multidimensional structured data. IEEE Signal Processing Magazine. 2013; 30:137–48.

48. Poli R, Salvaris M, Cinel C. Evolutionary synthesis of a trajectory integrator for an analogue brain-computer interface mouse. Applications of Evolutionary Computation: Springer; 2011. p. 214–23.

49. Koyama S, Chase SM, Whitford AS, Velliste M, Schwartz AB, Kass RE. Comparison of brain–computer interface decoding algorithms in open-loop and closed-loop control. Journal of computational neuroscience. 2010; 29(1–2):73–87. doi: 10.1007/s10827-009-0196-9 PMID: 19904595

50. Galán F, Nuttin M, Lew E, Ferrez PW, Vanacker G, Philips J, et al. A brain-actuated wheelchair: asynchronous and non-invasive brain–computer interfaces for continuous control of robots. Clinical Neurophysiology. 2008; 119(9):2159–69. doi: 10.1016/j.clinph.2008.06.001 PMID: 18621580

51. Göhring D, Latotzky D, Wang M, Rojas R. Semi-autonomous car control using brain computer interfaces. Intelligent Autonomous Systems 12: Springer; 2013. p. 393–408.

52. Gelb A. Applied optimal estimation: MIT press; 1974.

53. Wu W, Black M, Gao Y, Bienenstock E, Serruya M, Donoghue J, editors. Inferring hand motion from multi-cell recordings in motor cortex using a Kalman filter. SAB'02-workshop on motor control in humans and robots: On the interplay of real brains and artificial devices; 2002.

54. Su Y, Qi Y, Luo J-x, Wu B, Yang F, Li Y, et al. A hybrid brain-computer interface control strategy in a virtual environment. Journal of Zhejiang University SCIENCE C. 2011; 12(5):351–61.

55. Marathe AR, Taylor DM. The impact of command signal power distribution, processing delays, and speed scaling on neurally-controlled devices. Journal of neural engineering. 2015; 12(4):046031. Epub 2015/07/15. doi: 10.1088/1741-2560/12/4/046031 PMID: 26170261; PubMed Central PMCID: PMCPmc4547796.

56. Hyndman RJ, Koehler AB. Another look at measures of forecast accuracy. International Journal of Forecasting. 2006; 22(4):679–88.

57. Welch G, Bishop G. An Introduction to the Kalman Filter. University of North Carolina at Chapel Hill, 1995.

58. Tsui CSL, Gan JQ, Roberts SJ. A self-paced brain–computer interface for controlling a robot simulator: an online event labelling paradigm and an extended Kalman filter based algorithm for online training. Medical & biological engineering & computing. 2009; 47(3):257–65.

59. Li Z, O'doherty JE, Hanson TL, Lebedev MA, Henriquez CS, Nicolelis MA. Unscented Kalman filter for brain-machine interfaces. PloS one. 2009; 4(7):e6243. doi: 10.1371/journal.pone.0006243 PMID: 19603074

60. Pistohl T, Ball T, Schulze-Bonhage A, Aertsen A, Mehring C. Prediction of arm movement trajectories from ECoG-recordings in humans. Journal of neuroscience methods. 2008; 167(1):105–14. PMID: 18022247

61. Gowda S, Orsborn AL, Overduin S, Moorman HG, Carmena JM. Designing dynamical properties of brain–machine interfaces to optimize task-specific performance. Neural Systems and Rehabilitation Engineering, IEEE Transactions on. 2014; 22(5):911–20.

62. Seber GAF, Lee AJ. Linear regression analysis. 2nd ed. Hoboken, N.J.: Wiley-Interscience; 2003. xvi, 557 p. p.

63. Hastie T, Tibshirani R, Friedman J, Franklin J. The elements of statistical learning: data mining, inference and prediction. The Mathematical Intelligencer. 2005; 27(2):83–5.

64. Kohavi R. A study of cross-validation and bootstrap for accuracy estimation and model selection. Proceedings of the 14th international joint conference on Artificial intelligence—Volume 2; Montreal, Quebec, Canada. 1643047: Morgan Kaufmann Publishers Inc.; 1995. p. 1137–43.

65. Eliseyev A, Faber J, Aksenova T, editors. Classification of multi-modal data in a self-paced binary BCI in freely moving animals. Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE; 2011: IEEE.

66. Li J, Zhang L. Regularized tensor discriminant analysis for single trial EEG classification in BCI. Pattern Recognition Letters. 2010; 31(7):619–28.

67. Nazarpour K, Sanei S, Shoker L, Chambers JA. Parallel space-time-frequency decomposition of EEG signals for brain computer interfacing. Proc EUSIPCO06. 2006.

68. Frank LE, Friedman JH. A statistical view of some chemometrics regression tools. Technometrics. 1993; 35(2):109–35.

69. Bro R. Multi-way analysis in the food industry: models, algorithms, and applications: Københavns Universitet'Københavns Universitet', LUKKET: 2012 Det Biovidenskabelige Fakultet for Fødevarer, Veterinærmedicin og NaturressourcerFaculty of Life Sciences, LUKKET: 2012 Institut for FødevarevidenskabDepartment of Food Science, 2012 Institut for Fødevarevidenskab, 2012 Kvalitet og TeknologiDepartment of Food Science, Quality & Technology; 1998.

70. Lin Y, Zhang HH. Component selection and smoothing in multivariate nonparametric regression. The Annals of Statistics. 2006; 34(5):2272–97.

71. Schmidt M, Fung G, Rosales R. Fast optimization methods for l1 regularization: A comparative study and two new approaches. Machine Learning: ECML 2007: Springer; 2007. p. 286–97.

72. Eliseyev A, Moro C, Faber J, Wyss A, Torres N, Mestais C, et al. L1-penalized N-way PLS for subset of electrodes selection in BCI experiments. Journal of neural engineering. 2012; 9(4):045010. doi: 10.1088/1741-2560/9/4/045010 PMID: 22832155.

73. Yokota T, Zhao Q, Li C, Cichocki A. Smooth PARAFAC Decomposition for Tensor Completion. arXiv preprint arXiv:150506611. 2015.

74. Krämer N, Boulesteix A-L, Tutz G. Penalized Partial Least Squares with applications to B-spline transformations and functional data. Chemometrics and Intelligent Laboratory Systems. 2008; 94(1):60–9.

75. Tibshirani R, Saunders M, Rosset S, Zhu J, Knight K. Sparsity and smoothness via the fused lasso. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2005; 67(1):91–108.

76. Adams RA, Fournier JJ. Sobolev spaces: Academic press; 2003.

77. Nagasaka Y, Shimoda K, Fujii N. Multidimensional recording (MDR) and data sharing: an ecological open research and educational platform for neuroscience. PloS one. 2011; 6(7):e22561. doi: 10.1371/journal.pone.0022561 PMID: 21811633; PubMed Central PMCID: PMC3141074.

78. Cook RD, Weisberg S. Residuals and influence in regression. 1982.

79. Willett FR, Suminski AJ, Fagg AH, Hatsopoulos NG. Improving brain–machine interface performance by decoding intended future movements. Journal of neural engineering. 2013; 10(2):026011. doi: 10.1088/1741-2560/10/2/026011 PMID: 23428966

80. Eliseyev A, Moro C, Costecalde T, Torres N, Gharbi S, Mestais C, et al. Iterative N-way partial least squares for a binary self-paced brain-computer interface in freely moving animals. Journal of neural engineering. 2011; 8(4):046012. doi: 10.1088/1741-2560/8/4/046012 PMID: 21659695.

81. Shimada S, Hiraki K, Matsuda G, Oda I. Decrease in prefrontal hemoglobin oxygenation during reaching tasks with delayed visual feedback: a near-infrared spectroscopy study. Cognitive brain research. 2004; 20(3):480–90. PMID: 15268925

82. Kornhuber HH, Deecke L. Hirnpotentialänderungen bei Willkürbewegungen und passiven Bewegungen des Menschen: Bereitschaftspotential und reafferente Potentiale. Pflüger's Archiv für die gesamte Physiologie des Menschen und der Tiere. 284(1):1–17. doi: 10.1007/bf00412364

83. Marathe A, Taylor D. Decoding continuous limb movements from high-density epidural electrode arrays using custom spatial filters. Journal of neural engineering. 2013; 10(3):036015. doi: 10.1088/1741-2560/10/3/036015 PMID: 23611833

84. B. Morinière, Verney A, Abroug N, Garrec P, Perrot Y, editors. EMY: a dual arm exoskeleton dedicated to the evaluation of Brain Machine Interface in clinical trials. Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on; 2015 Sept. 28 2015-Oct. 2 2015.