

Principal component based covariate shift adaption to reduce non-stationarity in a MEG-based Brain-Computer Interface

Martin Spüler · Wolfgang Rosenstiel ·
Martin Bogdan

Received: date / Accepted: date

Abstract One of the biggest problems in today's BCI research is the non-stationarity of the recorded signals. This non-stationarity can cause the BCI performance to deteriorate over time or drop significantly when transferring data from one session to another. To reduce the effect of non-stationaries, we propose a new method for covariate shift adaption that is based on Principal Component Analysis to extract non-stationaries and alleviate them. We show the proposed method to significantly increase BCI performance for an MEG-based BCI in an offline analysis as well as an online experiment with 10 subjects. We also show the method to be superior to other covariate shift adaption methods and present examples of identified non-stationaries to show the effect of the proposed method.

Keywords BCI · non-stationarity · covariate shift adaption · PCA

1 Introduction

A Brain-Computer Interfaces (BCI) enables a user to communicate or control a computer by means of pure brain activity without the need for muscle control. Its primary field of application is to help people who have lost voluntary muscle control due to diseases or traumatic injuries. While BCIs can be used for rehabilitation after stroke, they are most prominently used as a communication device for patients suffering from locked-in syndrome. The locked-in syndrome can be caused by different neurodegenerative diseases (like amyotrophic lateral sclerosis), brain-stem stroke or traumatic brain injuries. The locked-in state describes a condition in which the patient is aware and awake but paralysed and therefore unable to move or to communicate verbally or by any other means of muscle activity. A BCI can enable such patients to communicate or to control a computer and interact with their environment[1].

Martin Spüler
Sand 14, 72076 Tübingen, Germany
Tel.: +49-7071-2970490
E-mail: spueeler@informatik.uni-tuebingen.de

The basic principle of a BCI relies on the user being able to voluntarily alter his brain activity. These changes in the recorded brain activity can be detected and used as an input signal for a computer. There are different signal acquisition techniques that allow to measure the brain activity of a user.

While electro-encephalography (EEG) is the most popular non-invasive method, we concentrate on data recorded by magneto-encephalography (MEG) in this paper. Typically MEG is associated with higher costs, which may be the reason for seldomly being used, but it also provides a higher spatial resolution (due to the larger amount of sensors) and more information in the higher frequency range above 40 Hz. While it has been shown to work well with BCI[2], it still suffers from the same problem as EEG-based BCIs, namely the non-stationarity of the recorded signals.

Reasons for non-stationarity include changes in the mental state over time (increasing fatigue or losing concentration), the transfer from training without feedback to online usage with feedback or head movements in the MEG, which cause the generating brain areas to be under a different sensor. These non-stationaries especially are a problem when a classifier trained on data of a previous session is used for classification in a current session, which is often referred to as the session-transfer problem. From the machine learning point of view this phenomenon is termed covariate shift and describes the fact that the training data follows a different distribution than the test data[3].

There have been different ways how to approach the problem of non-stationarity by adaptive classification[4, 5], adaptive spatial filters[6, 7] or using covariate shift adaption[8–11]. In this paper we propose a method that first uses Principal Component Analysis to extract non-stationaries and then minimize the effect of these non-stationaries by covariate shift adaption of the principal components. We present results from offline data and prove the benefit of this method in an online experiment with 10 subjects.

2 Method

2.1 Covariate shift adaption by normalized principal components

In this section we propose a method for covariate shift adaption which uses Principal Component Analysis (PCA)[12] to extract the most important principal components and normalizes these components by shifting a window over the data to reduce the effect of non-stationarity. The normalization is similar to the method described in [8, 9], but normalizes each feature individually instead of normalizing a linear combination of all features.

PCA is a method that uses an orthogonal transformation to convert a set of possibly correlated variables into a set of uncorrelated variables. These uncorrelated variables are called principal components and are sorted by the amount of variance that the principal components account for in the original data. The first principal components accounts for the highest proportion of variance in the original data.

The proposed method is applied after feature extraction when the power spectrum for each channel has been estimated. When having n trials of training data,

the dataset consists of a matrix D with dimension $n \times p$, with the number of features $p = (\text{channels} \cdot \text{bins})$ and $D_{(i,j)}$ being the value for trial i and feature j . For the covariate shift adaption, first a PCA is applied for dimensionality reduction and extraction of non-stationary components. As a next step the m principal components with the highest variance are selected, resulting in a $p \times m$ transformation matrix W and a matrix $P = D \cdot W$ that represents the m principal components. For the data presented here $m = 100$ was used, which seems to be a robust value giving good results. As a next step a rectangular window of length w is defined which is shifted over the data and the value of each $P_{(i,j)}$ is normalized by the preceding w trials with

$$\hat{P}_{(ij)} = P_{(ij)} - \text{mean}(P_{(i-w,j)}, \dots, P_{(i-1,j)})$$

For all $\hat{P}_{(ij)}$ with $i \leq w$ the window $(P_{(1,j)}, \dots, P_{(w,j)})$ is used. We also experimented with different types of windows, e.g. a half hanning window, but found the rectangular window to give best results.

When using the method online, the last w trials $(P_{t-w}, \dots, P_{t-1})$ are kept in a buffer B and the principal components for a new trial D_t are calculated by $P_t = D_t \cdot W$ and P_t is normalized by the mean of the buffer

$$\hat{P}_{(tj)} = P_{(tj)} - \text{mean}(B_{(1,j)}, \dots, B_{(w,j)})$$

Then P_t is added to the end of B and the first trial in B is removed, to keep the latest w trials in B . \hat{P}_t can then be used for classification.

2.2 Covariate shift adaption methods

In the following we give an overview of the different methods for covariate shift adaption that are tested in this paper. The covariate shift adaption methods were applied after the signal processing as explained in section 2.3.2.

Satti et al.: in [13] a method for covariate shift adaption was proposed, that uses a polynomial function for estimating the covariate shift of the next trial and adapt the data accordingly. In the following we used a polynomial of order 3 like stated in [13] and used the previous w trials to fit the polynomial to the data.

baseline: as a reference method we use results without covariate shift adaption.

pcanorm: this is the method for covariate shift adaption by normalized principal components as proposed in this paper in section 2.1.

pcapoly: with this method we propose a slightly different approach than the one presented in section 2.1. Also a PCA is used, but instead of shifting a window over the data and normalizing by the last w trials, a method similar to [13] is used: a polynomial is fitted to the content of the window and the next trial is adjusted by the value, the polynomial function would estimate. Again a polynomial of order 3 is used.

pcaonly: for this method, PCA was used for extraction of non-stationaries without any further covariate shift adaption.

Although different w were tested, for preparing the results w was kept constant at $w = 15$, to provide a fair comparison between the methods.

2.3 Offline analysis

2.3.1 Data

To evaluate the advantage of different covariate shift adaption methods we performed an offline analysis on data recorded for another study[14]. In this study 10 subjects performed motor imagery of right hand movement and a subtraction task. In the subtraction task the subject had to do subtractions by choosing a random number (around 100) and subtract 7. The result should not be communicated but it should be continued by subtracting 7 from the result and doing this all over until the end of a trial. Two sessions were recorded on different days with 51 trials per task and session. Recording was done with a 275-channel whole-head MEG-system (VSM MedTech Ltd.) at a sampling rate of 586 Hz. Each trial lasted 4.05 seconds with about 6 seconds of break between the trials. Instructions were given on a screen and a fixation cross was displayed during trials to minimize eye movement.

2.3.2 Signal processing and classification

The signals were filtered and resampled to 200 Hz. For spatial filtering a small laplacian derivation was applied. To reduce the number of channels we only used the 185 inner channels, which should also reduce the influence of possible artifacts, which are most prominent on the outer channels. After the preprocessing the power spectrum was estimated by an autoregressive model computed with the Burg Method, as it was used in a previous MEG-BCI[2]. A model order of 16 was used, since we obtained best results with this model order in previous MEG-BCI experiments. We used the frequency range from 1 to 40 Hz with a bin width of 2 Hz. The logarithm function was applied to each value. Before classification we used r^2 -ranking[15] for feature selection. **The number of features was not estimated individually on the training data, which would have introduced overfitting in our experience. Instead a fixed number of 1000 features was used, which gave on average the best results when evaluated by cross-validation.** Each feature was normalized to have zero mean and unit variance for the training dataset. The test dataset was scaled according to the mean and standard deviation of the training dataset.

For classification we used LibSVM[16] with $C = 1$ and a linear kernel. We decided against a parameter estimation by gridsearch and cross-validation because it introduced overfitting in previous experiments.

2.3.3 Offline accuracy evaluation

To evaluate the performance of the method proposed in section 2.1 and to compare it to other covariate shift adaption methods (see section 2.2) we trained the classifier after using the respective covariate shift adaption method on session 1 of the data and tested it on session 2, referenced to as S1S2-validation later. This method especially addresses the benefits of the covariate shift adaption methods in context of the session-transfer problem.

For comparison reasons we also performed a 5x10-fold crossvalidation on all data, in which the data was permuted and partitioned into 10 blocks with equal

size. In each fold 9 blocks were used for training the classifier (**including feature selection and PCA**) and tested on the one remaining block. Each block was used for testing once. This procedure was repeated 5 times and the accuracy was averaged over all folds.

While non-stationaries have a great effect in the S1S2-validation, since the test set has an unknown data distribution, this effect should be minimized when using a crossvalidation (CV), because of the data being permuted and the classifier knowing the data distribution from both sessions. Using both validation methods allows for a direct estimation by how much non-stationaries are alleviated by the covariate shift adaption method proposed in section 2.1. To specifically adress this issue and due to the fact that the proposed method wouldn't make sense on permuted data, no covariate shift adaption was performed for the crossvalidation. For a fair comparison, we not only used the baseline method but also the baseline method combined with PCA for a dimensionality reduction to the $m = 100$ principal components with the highest variance. Although the number of features used for PCA-based methods differs from the number of features used for the baseline-method, we always used the number of features that gave best results in a cross-validation for the specific method.

2.4 Online experiment

To confirm the results from the offline analysis, we integrated the test of the proposed method in an ongoing online experiment with 10 subjects, who had to perform motor imagery and mental subtraction. To explicitly evaluate the covariate shift adaption in context of the session-transfer problem, each subjects should participate in two sessions. In the first session 200 trials training data were recorded. In the second session the classifier was trained on the training data from the first session and the proposed method was tested with online feedback in 200 trials. Each session was seperated into runs with 40 trials and a short break after each run.

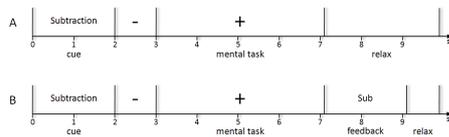


Fig. 1 Online paradigm. (A) training phase without feedback. (B) test phase with feedback

Recording was done again with a 275-channel whole-head MEG-system (VSM MedTech Ltd.) at a sampling rate of 586 Hz. During measurement the head position was continuously recorded. A Notebook with an Intel Core i7 720QM and 4GB memory running BCI2000[17] was used for signal acquisition, signal processing, feedback presentation and classification. The design of the paradigm and the corresponding time intervals are shown in figure 1. During the test phase feedback was given after every trial, which indicated the result of the classifier. Since the online test of the proposed method was integrated during the ongoing experiment, it should be noted that the first 4 subjects (B01,B03,B07,B08) did receive feedback

without the covariate shift adaption method and the results of these 4 subjects shown below are from a simulated online experiment. The other 6 subjects received online feedback with the covariate shift adaption method proposed in this paper.

To test how the performance deterioration during one session is affected by the proposed covariate shift adaption method, we did a linear regression (least squares regression by Matlab’s *polyfit* function) on the accuracy throughout a session and used the slope of the regression line as a measure for performance deterioration.

To compare the online results with the *pcanorm*-method to the baseline-method, the baseline-method was applied offline to simulate the online experiment with the same data and same parameters but different covariate shift adaption method.

3 Results

3.1 Offline analysis

The comparison of the crossvalidation results and the results from the S1S2-validation are shown in table 1. It can be seen that the method proposed in section 2.1 performs significantly better ($p < 0.005$, paired t-test) than the baseline method without covariate shift adaption. The comparison between the results obtained by crossvalidation and the results for the S1S2-validation shows that the baseline method still suffers a significant performance decrease ($p < 0.05$) due to the session-transfer, while there is no significant difference when comparing the proposed method with the crossvalidation results from the baseline method with PCA ($p > 0.1$).

Since robust PCA has been proposed as a better method to model non-stationaries[18], we also investigated if the use of robust PCA would increase the results with the method proposed in this paper. Due to the low amount of trials (102 training trials in the S1S2-validation) it was not possible to calculate robust PCA with $k=100$ principal components[19]. Since it is also advised to choose a $k \leq \frac{n}{2}$, we used robust PCA with $k = 50$ to compare both methods with the first 50 principal components. Since the results deviate very little, they are not shown here in detail, but it should be noted that the proposed method with robust PCA achieved an average accuracy of 90.3 % while the proposed method with PCA achieved an average accuracy of 90.0 % which is not significantly different ($p > 0.5$, paired t-test). Although robust PCA achieved slightly better results, we continued with the use of PCA due to the limitation in the number of principal components when using robust PCA and a higher computation time.

The results from the comparison of different covariate shift adaption methods can be seen in table 2. It shows, that the proposed method with a mean accuracy of 92.2 % is superior to all other methods tested. While all covariate shift adaption methods perform significantly better ($p < 0.05$) than the baseline method without covariate shift adaption, the accuracies with the proposed method are significantly higher than with the other two tested covariate shift adaption methods ($p < 0.005$).

Since the number of features differed between the *pcanorm*-method (100 features) and the *baseline*-method (1000 features), it should be stated, that the *baseline*-method with 100 features resulted in an average accuracy of 76.8%. The *pcanorm*-method with 1000 features is not significantly better than chance level

Table 1 Offline classification accuracies for the baseline method and the pca-normalization. Results on the left side(CV) are estimated by a 5x10-fold cross-validation. For results on the right(S1S2), the classifier was trained on session 1 and tested on session 2. Best results are printed bold.

subject	CV		S1S2	
	baseline	pca	baseline	pcanorm
A01	88.5 %	90.9 %	86.3 %	88.2 %
A02	86.7 %	90.6 %	82.4 %	94.1 %
A03	92.4 %	93.6 %	77.5 %	96.1 %
A04	92.9 %	94.1 %	90.2 %	90.2 %
A05	94.8 %	97.4 %	80.4 %	95.1 %
A06	89.9 %	90.1 %	63.7 %	88.2 %
A07	96.1 %	97.8 %	91.4 %	98.0 %
A08	99.3 %	98.6 %	91.4 %	98.0 %
A09	69.0 %	81.4 %	69.6 %	74.5 %
A10	98.6 %	98.2 %	96.1 %	99.0 %
mean	90.8 %	93.3 %	83.4 %	92.2 %

Table 2 Offline classification accuracies for different methods for covariate shift adaption. Best results are printed bold.

	baseline	pcanorm	pcapoly	pcaonly	Satti et al.
A01	86.3 %	88.2 %	83.3 %	80.4 %	84.3 %
A02	82.4 %	94.1 %	89.2 %	91.2 %	82.4 %
A03	77.5 %	96.1 %	98.0 %	95.1 %	90.2 %
A04	90.2 %	90.2 %	87.2 %	86.3 %	88.2 %
A05	80.4 %	95.1 %	94.1 %	91.2 %	92.2 %
A06	63.7 %	88.2 %	81.4 %	79.4 %	82.4 %
A07	91.4 %	98.0 %	95.1 %	97.1 %	94.1 %
A08	91.4 %	98.0 %	96.1 %	97.1 %	95.1 %
A09	69.6 %	74.5 %	73.5 %	72.6 %	72.6 %
A10	96.1 %	99.0 %	96.1 %	97.1 %	98.0 %
mean	83.4 %	92.2 %	89.4 %	88.7 %	87.9 %

(50 %), since it is highly affected by the curse of dimensionality, due to too many principal components with a variance ≈ 0 .

3.2 Online experiment

The results from the online experiment are shown in table 3. With a mean accuracy of 80.9 % the proposed method performs better ($p < 0.1$) than the baseline method with a mean accuracy of 76.4 %. There is no significant correlation between the performance improvement by the proposed covariate shift adaption method and the number of days between the sessions.

The results from the linear regression analysis, in which the slope of the regression line is used as a measure for performance deterioration is not shown in detail. But it is worth mentioning, that the slope averaged over all subjects with the proposed method was -0.06 ± 0.62 , while it was -0.32 ± 0.61 for the baseline method without covariate shift adaption. Although there is less performance dete-

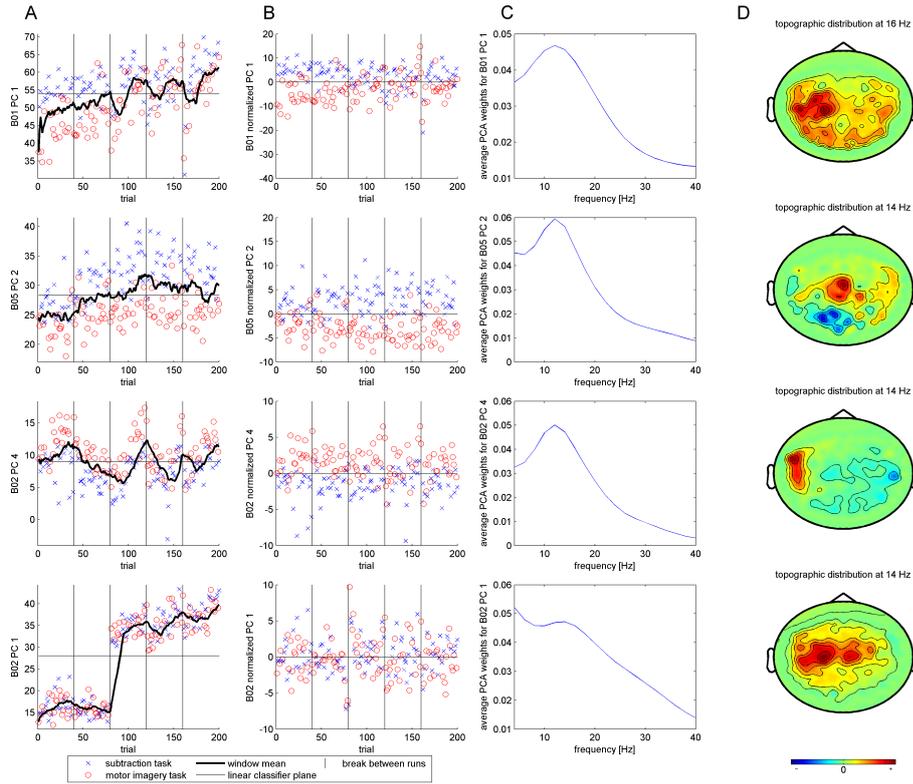


Fig. 2 Examples of different principal components. (A) raw principal component (B) normalized principal components (C) average PCA weights per frequency bin (D) topographic distribution

rioration with the proposed covariate shift adaption method, the difference is not significant ($p > 0.1$).

3.3 Example of different principal components

In figure 2 four examples of different principal components are given. In figure 2(A) the raw principal component is shown as well as the linear classifier plane (horizontal line) and the mean of the window with the $w = 15$ preceding trials. The vertical lines depict the break between two runs. In figure 2(B) the principal component after the covariate shift adaption as well as the corresponding linear classifier plane is shown. In figure 2(C) and (D) the frequency range and the topographic distribution of the principal component can be seen. The first three principal components are an example of the proposed method improving classification accuracy by removing temporal fluctuations in the principal components and thereby making the two classes easier separable. The second principal component is an interesting case in which one can also see an topographic shift of activation over time. With an increasing value for this principal component the power in the alpha band increases in central regions, while it decreases in the posterior region.

Table 3 Accuracies during the online experiment with the proposed method and the baseline method as well as the days between session 1 session 2.

	online (pcanorm)	baseline	d. between S1S2
B01	90.5 %	68.0 %	2
B02	85.0 %	83.5 %	1
B03	96.5 %	94.0 %	1
B04	86.5 %	94.0 %	1
B05	97.0 %	97.5 %	1
B06	86.0 %	77.5 %	6
B07	63.5 %	59.0 %	1
B08	81.5 %	68.0 %	1
B09	70.0 %	60.0%	0
B10	52.0 %	62.0 %	0
mean	80.9 %	76.4 %	1.4

The fourth component is an example for a big non-stationary effect that happend during a longer (than usual) break between two runs.

4 Discussion

The offline results show that the method proposed in this paper is a useful tool to reduce the effect of non-stationaries and improves classification accuracies for BCI. In the offline analysis it has been shown that the proposed method significantly increases classification performance. It also performs significantly better than the other tested methods.

To validate the offline results and to show that the proposed method can be used in online BCI, we integrated the proposed method in an online experiment with 10 subjects. Although 2 of the 10 subjects had a noticeable performance decrease, the average performance could be increased by 4.5 % through the use of the PCA-based covariate shift adaption method proposed in this paper. In addition there is less performance deterioration during a session when using the proposed covariate shift adaption method. Since the results in the online experiment are on the verge to significance ($p < 0.1$), more subjects may be needed to show a significant result. With the online experiment we have shown the method to be feasible and computationally efficient enough to be used in an online BCI. While the use of PCA increases the time needed for calibration by some seconds, there is no noticeable increase in the computational cost in the online case and as we have shown, the method is fast enough to be used online even with high-dimensional MEG data.

Although PCA is not specifically designed to extract non-stationaries, the examples in section 3.3 underline the conclusion drawn in [18], that PCA is a useful method to model non-stationaries and may help to understand the underlying processes or neurophysiological changes. By analysing the frequency spectrum and the topographical distribution of the principal components, one might be able to draw conclusions about the origin of the non-stationaries and find new methods to alleviate them. Methods like Stationary Subspace Analysis (SSA)[20], that are tailored to decompose multivariate signals in stationary and non-stationary parts

might give even better results. But in our case SSA could not be applied since the number of dimensions was much higher than the number of trials.

Since changes in alpha power are known to reflect changing levels of fatigue[21], this might explain some of the patterns that are shown in section 3.3. Especially a pattern that repeats every run or changes constantly over the whole session is likely to be associated with increasing fatigue, since many subjects reported becoming more tired or losing concentration over the course of a session and over the course of a run. When recording with EEG other types of components may arise that for example may reflect changing impedances of the electrodes.

5 Conclusion

In this paper we have proposed a new method for covariate shift adaption, which is based on Principal Component Analysis to model non-stationaries. We have shown it to significantly increase BCI performance in an offline analysis and an online experiment with 10 subjects. With the online experiment we have also shown the proposed method to be efficient enough to be used in an online BCI.

The proposed covariate shift adaption method is a step towards a more robust BCI. By reducing the effects of non-stationaries it alleviates the session-transfer problem and keeps the performance from deteriorating during a session.

Competing interests

The authors declare that they have no competing interests.

Acknowledgements

This study was partly granted by the German Federal Ministry of Education and Research (BMBF, BFNT F*T, Grant UTü 01 GQ 0831) and the DFG (Grant RO 1030/15-1, KOME G).

References

1. A. Kübler, F. Nijboer, J. Mellinger, T. Vaughan, H. Pawelzik, G. Schalk, D. McFarland, N. Birbaumer, J. Wolpaw, *Neurology* **64**(10), 1775 (2005)
2. J. Mellinger, G. Schalk, C. Braun, H. Preissl, W. Rosenstiel, N. Birbaumer, A. Kübler, *NeuroImage* **36**(3), 581 (2007)
3. M. Sugiyama, M. Krauledat, K.R. Müller, J. Mach. *Learn. Res.* **8**, 985 (2007). URL <http://dl.acm.org/citation.cfm?id=1314498.1390324>
4. Y. Li, H. Kambara, Y. Koike, M. Sugiyama, *IEEE Transactions on Biomedical Engineering* **57**, 1318 (2010)
5. C. Vidaurre, M. Kawanabe, P. von Bünau, B. Blankertz, K. Müller, *Biomedical Engineering, IEEE Transactions on* **58**(3), 587 (2011). DOI 10.1109/TBME.2010.2093133
6. R. Tomioka, J. Hill, B. Blankertz, K. Aihara, in *IBIS 2006* (Max-Planck-Gesellschaft, 2006), pp. 65–70
7. Q. Zhao, L. Zhang, A. Cichocki, J. Li, in *Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on* (2008), pp. 2656 –2659. DOI 10.1109/IJCNN.2008.4634170

8. H. Ramoser, J.R. Wolpaw, G. Pfurtscheller, *Biomedizinische Technik. Biomedical engineering* **42**(9), 226 (1997)
9. G. Fabiani, D. McFarland, J. Wolpaw, G. Pfurtscheller, *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* **12**(3), 331 (2004). DOI 10.1109/TNSRE.2004.834627
10. B. Reuderink, J. Farquhar, M. Poel, A. Nijholt, in *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE* (2011), pp. 4600 – 4604. DOI 10.1109/IEMBS.2011.6091139
11. B. Reuderink, J. Farquhar, M. Poel, *International journal of bioelectromagnetism* **13**(2), 78 (2011)
12. I.T. Jolliffe, *Principal Component Analysis* (Springer-Verlag, New York, 1986)
13. A. Satti, C. Guan, D. Coyle, G. Prasad, in *Proceedings of the 20th International Conference on Pattern Recognition* (2010), pp. 105–108
14. M. Bensch, J. Mellinger, M. Bogdan, W. Rosenstiel, in *Proceedings of the 4th Int. Brain-Computer Interface Workshop* (Graz, 2008), pp. 191–196
15. M. Spüler, W. Rosenstiel, M. Bogdan, in *Proceedings of the 5th Int. Brain-Computer Interface Conference* (Graz, 2011), pp. 24–27
16. C.C. Chang, C.J. Lin, *LIBSVM: a library for support vector machines* (2001). Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
17. G. Schalk, D.J. Mcfarland, T. Hinterberger, N. Birbaumer, J.R. Wolpaw, *IEEE Transactions on Biomedical Engineering* **51**(6), 1034 (2004). DOI 10.1109/TBME.2004.827072. URL <http://dx.doi.org/10.1109/TBME.2004.827072>
18. J. Pascual, M. Kawanabe, C. Vidaurre, in *Hybrid Artificial Intelligent Systems, Lecture Notes in Computer Science*, vol. 6679, ed. by E. Corchado, M. Kurzynski, M. Wozniak (Springer Berlin / Heidelberg, 2011), pp. 51–58
19. M. Hubert, P.J. Rousseeuw, *Technometrics* **47**(1), pp. 64 (2005). URL <http://www.jstor.org/stable/25470935>
20. P. von Bünau, F.C. Meinecke, F.C. Király, K.R. Müller, *Physical Review Letters* **103**(21), 214101 (2009). DOI 10.1103/PhysRevLett.103.214101
21. W. Klimesch, *Brain Research Reviews* **29**(2-3), 169 (1999). DOI 10.1016/S0165-0173(98)00056-3. URL [http://dx.doi.org/10.1016/S0165-0173\(98\)00056-3](http://dx.doi.org/10.1016/S0165-0173(98)00056-3)