HAL
open science

A Review of Classification Algorithms for EEG-based Brain-Computer Interfaces: A 10-year Update

Fabien Lotte, Laurent Bougrain, Andrzej Cichocki, Maureen Clerc, Marco Congedo, Alain Rakotomamonjy, Florian Yger

To cite this version:

Fabien Lotte, Laurent Bougrain, Andrzej Cichocki, Maureen Clerc, Marco Congedo, et al.. A Review of Classification Algorithms for EEG-based Brain-Computer Interfaces: A 10-year Update. Journal of Neural Engineering, 2018, 15 (3), pp.55. 10.1088/1741-2552/aab2f2 . hal-01846433

HAL Id: hal-01846433

https://inria.hal.science/hal-01846433v1

Submitted on 21 Jul 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Review of Classification Algorithms for EEG-based Brain-Computer Interfaces: A 10-year Update

F Lotte[1,2], L Bougrain[3,4], A Cichocki[2,5,10], M Clerc[6], M Congedo[7], A Rakotomamonjy[8], and F Yger[9]

[1] Inria, LaBRI (CNRS/Univ. Bordeaux /INP), Talence, France
[2] RIKEN Brain Science Insitute, Wakoshi, Japan
[3] Univ. Lorraine, Nancy, France
[4] Inria Nancy Grand-Est / LORIA, Nancy, France
[5] Skolkowo Institute of Science and Technology, Moscow, Russia
[6] Inria, Université Côte d'Azur, France
[7] Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, Grenoble, France
[8] Univ. Rouen / LITIS, Rouen, France
[9] Univ. Paris-Dauphine, PSL Research Univ. / CNRS, LAMSADE, Paris, France
[10] Nicolaus Copernicus University, Torun, Poland

E-mail: `fabien.lotte@inria.fr`

Abstract.

Objective: Most current Electroencephalography (EEG)-based Brain-Computer Interfaces (BCIs) are based on machine learning algorithms. There is a large diversity of classifier types that are used in this field, as described in our 2007 review paper. Now, approximately 10 years after this review publication, many new algorithms have been developed and tested to classify EEG signals in BCIs. The time is therefore ripe for an updated review of EEG classification algorithms for BCIs.

Approach: We surveyed the BCI and machine learning literature from 2007 to 2017 to identify the new classification approaches that have been investigated to design BCIs. We synthesize these studies in order to present such algorithms, to report how they were used for BCIs, what were the outcomes, and to identify their pros and cons.

Main results: We found that the recently designed classification algorithms for EEG-based BCIs can be divided into four main categories: adaptive classifiers, matrix and tensor classifiers, transfer learning and deep learning, plus a few other miscellaneous classifiers. Among these, adaptive classifiers were demonstrated to be generally superior to static ones, even with unsupervised adaptation. Transfer learning can also prove useful although the benefits of transfer learning remain unpredictable. Riemannian geometry-based methods have reached state-of-the-art performances on multiple BCI problems and deserve to be explored more thoroughly, along with tensor-based methods. Shrinkage linear discriminant analysis and random forests also appear particularly useful for small training samples settings. On the other hand, deep learning methods have not yet shown convincing improvement over state-of-the-art BCI methods.

Significance: This paper provides a comprehensive overview of the modern classification algorithms used in EEG-based BCIs, presents the principles of these

methods and guidelines on when and how to use them. It also identifies a number of challenges to further advance EEG classification in BCI.

Keywords: Brain-Computer Interfaces, BCI, EEG, Electroencephalography, signal processing, spatial filtering, machine learning, feature extraction, classification, adaptive classifiers, deep learning, Riemannian geometry, transfer learning, tensors.

Submitted to: *J. Neural Eng.*

1. Introduction

A Brain-Computer Interface (BCI) can be defined as a system that translates the brain activity patterns of a user into messages or commands for an interactive application, this activity being measured and processed by the system [229, 139, 44]. A BCI user's brain activity is typically measured using Electroencephalography (EEG). For instance, a BCI can enable a user to move a cursor to the left or to the right of a computer screen by imagining left or right hand movements, respectively [230]. As they make computer control possible without any physical activity, EEG-based BCIs promise to revolutionize many applications areas, notably to enable severely motor-impaired users to control assistive technologies, e.g., text input systems or wheelchairs [181], as rehabilitation devices for stroke patients [8], as new gaming input devices [52], or to design adaptive human-computer interfaces that can react to the user's mental states [237], to name a few [216, 45].

In order to use a BCI, two phases are generally required: 1) an offline training phase during which the system is calibrated and 2) the operational online phase in which the system can recognize brain activity patterns and translate them into commands for a computer [136]. An online BCI system is a closed-loop, starting with the user producing a specific EEG pattern (e.g., using motor imagery) and these EEG signals being measured. Then, EEG signals are typically pre-processed using various spatial and spectral filters [23], and features are extracted from these signals in order to represent them in a compact form [140]. Finally, these EEG features are classified [141] before being translated into a command for an application [45] and before feedback is provided to users to inform them whether a specific mental command was recognized or not [170].

Although much effort is currently under way towards calibration-free modes of operation, an off-line calibration is currently used and is necessary in most BCIs to obtain a reliable system. In this stage, the classification algorithm is calibrated and the optimal features from multiple EEG channels are selected. For this calibration, a training data set needs to be pre-recorded from the user. EEG signals are highly user-specific, and as such, most current BCI systems are calibrated specifically for each user.

This training data set contains EEG signals recorded while the user performed each mental task of interest several times, according to given instructions.

There are various key elements in the BCI closed-loop, one being the classification algorithms a.k.a *classifiers* used to recognize the users' EEG patterns based on EEG features. There was, and still is, a large diversity of classifier types that are used and have been explored to design BCIs, as presented in our 2007 review of classifiers for EEG-based BCIs [141]. Now, approximately 10 years after this initial review was published, many new algorithms have been designed and explored in order to classify EEG signals in BCI, and BCIs are more popular than ever. We therefore believe that the time is ripe to update this review of EEG classifiers. Consequently, in this paper, we survey the literature on BCI and machine learning from 2007 to 2017 in order to identify which new EEG classification algorithms have been investigated to design BCI, and which appear to be the most efficient‡. Note that we also include in the present review machine learning methods for EEG feature extraction, notably to optimize spatial filters, which have become a key component of BCI classification approaches. We synthesize these readings in order to present these algorithms, to report how they were used for BCIs and what were the outcomes. We also identify their pros and cons in order to provide guidelines regarding how and when to use a specific classification method, and propose some challenges that must be solved to enable further progress in EEG signal classification.

This paper is organized as follows. Section 2 briefly presents the typically used EEG feature extraction and selection techniques, as these features are usually the input to classifiers. It also summarizes the classifier performance evaluation metrics. Then, Section 3.1 provides a summary of the classifiers that were used for EEG-based BCIs up to 2007, many of which are still in use today, as well as the challenges faced by current EEG classification methods. Section 4 describes the core of the paper, as it reviews the classification algorithms for BCI that have been explored since 2007 to address these various challenges. These algorithms are discussed in Section 5, where we also propose guidelines on how and when to use them, and identify some remaining challenges. Finally, Section 6 concludes the paper.

2. Feature extraction and selection, and performance measures in brief

The present paper is dedicated to classification methods for BCI. However, most pattern recognition/machine learning pipelines, and BCIs are no exception, not only use a classifier, but also apply feature extraction/selection techniques to represent EEG signals in a compact and relevant manner. In particular for BCI, EEG signals are typically filtered both in the time domain (band-pass filter), and spatial domain (spatial filter)

‡ This updated review describes more advanced classification concepts and algorithms than the ones presented in the initial review in [141]. We thus advise our readers new to the EEG classification field to start by reading [141], as that paper is more accessible, and the concepts it presented will not be explained again in the current manuscript.

before features are extracted from the resulting signals. The best subsets of features are then identified using feature selection algorithms, and these features are used to train a classifier. This process is illustrated in Figure 1. In this chapter, we briefly discuss which features are typically used in BCI, how to select the most relevant features amongst these and how to evaluate the resulting pattern recognition pipeline.

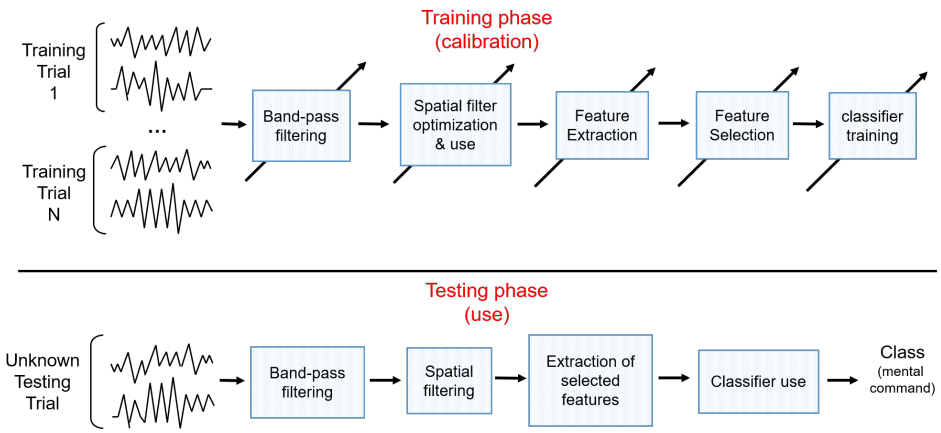

Figure 1. Typical classification process in EEG-based BCI systems. The oblique arrow denotes algorithms that can be or have to be optimized from data. A training phase is typically necessary to identify the best filters and features and to train the classifier. The resulting filters, features and classifier are then used online to operate the BCI.

2.1. Feature Extraction

While there are many ways in which EEG signals can be represented (e.g. [16, 136, 155]), the two most common types of features used to represent EEG signals are frequency band power features and time point features.

Band power features represent the power (energy) of EEG signals for a given frequency band in a given channel, averaged over a given time window (typically 1 second for many BCI paradigms). Band power features can be computed in various ways [28, 87], and are extensively used for BCIs exploiting oscillatory activity, i.e. changes in EEG rhythm amplitudes. As such, band power features are the gold standard features for BCI based on motor and mental imagery for many passive BCI aiming at decoding mental states such as mental workload or emotions, or for Steady State Visual Evoked Potential (SSVEP)-based BCIs.

Time point features are a concatenation of EEG samples from all channels. Typically, such features are extracted after some pre-processing, notably band-pass or low-pass filtering and down-sampling. They are the typical features used to classify Event Related Potentials (ERP), which are temporal variations in EEG signals amplitudes time-locked to a given event/stimulus [22, 136]. These are the features used in most P300-based BCI.

Both types of features benefit from being extracted after spatial filtering [22, 188, 185, 136]. Spatial filtering consists of combining the original sensor signals, usually linearly, which can result in a signal with a higher signal-to-noise ratio than that of individual sensors. Spatial filtering can be data independent, e.g., based on physical consideration regarding how EEG signals travel through the skin and skull, leading to spatial filters such as the well-known Laplacian filter [160] or inverse solution based spatial filtering [101, 18, 173, 124]. Spatial filters can also be obtained in a data-driven and unsupervised manner with methods such as Principal Component Analysis (PCA) or Independent Component Analysis (ICA) [98]. Finally, spatial filters can be obtained in a data-driven manner, with supervised learning, which is currently one of the most popular approaches. Supervised spatial filters include the well-known Common Spatial Patterns (CSP) [185, 23], dedicated to band-power features and oscillatory activity BCI, and spatial filters such as xDAWN [188] or Fisher spatial filters [92] for ERP classification based on time point features. Owing to the good classification performances obtained by such supervised spatial filters in practice, many variants of such algorithms have been developed that are more robust to noise or non-stationary signals, using regularization approaches, robust data averaging, and/or new divergence measures, (e.g. [194, 143, 187, 211, 233]). Similarly, extensions of these approaches have been proposed to optimize spectral and spatial filters simultaneously (e.g. the popular Filter Bank CSP (FBCSP) method [7] and others [61, 88, 161]). Finally, some approaches have combined both physically-driven spatial filters based on inverse models with data-driven spatial filters (e.g. [49, 148]).

While spatial filtering followed by either band power or time points feature extraction are by far the most common features used in current EEG-based BCIs, it should be mentioned that other feature types have been explored and used. Firstly, an increasingly used type is connectivity features. Such features measure the correlation or synchronization between signals from different sensors and/or frequency bands. This can be measured using features such as spectral coherence, phase locking values or directed transfer functions, among many others [31, 79, 167, 110, 225, 240]. Researchers have also explored various EEG signal complexity measures or higher order statistics as features of EEG signals (e.g. [29, 135, 11, 248]). Finally, rather than using vectors of features, recent research has also explored how to represent EEG signals by covariance matrices or by tensors (i.e. arrays and multi-way arrays, with two or more dimensions), and how to classify these matrices or tensors directly [232, 47, 38]. Such approaches are discussed in Section 4.2. It should be mentioned that when using matrix or tensor decompositions, the resulting features are linear combinations of various sensors data, time points or frequencies (among others). As such they may not have an obvious physical/physiological interpretation, but nonetheless prove useful for BCI design.

Finally, it is interesting to note that several BCI studies have reported that combining various types of features, e.g. time points with band powers or band powers with connectivity features, generally leads to higher classification accuracies as compared to using a single feature type (e.g. [60, 29, 70, 166, 191, 93]). Combining multiple feature

types typically increases dimensionality; hence it requires the selection of the most relevant features to avoid the curse-of-dimensionality. Methods to reduce dimensionality are described in the following section.

2.2. Feature Selection

A feature selection step can be applied after the feature extraction step to select a subset of features with various potential benefits [82]. Firstly, among the various features that one may extract from EEG signals, some may be redundant or may not be related to the mental states targeted by the BCI. Secondly, the number of parameters that the classifier has to optimize is positively correlated with the number of features. Reducing the number of features thus leads to fewer parameters to be optimized by the classifier. It also reduces possible overtraining effects and can thus improve performance, especially if the number of training samples is small. Thirdly, from a knowledge extraction point of view, if only a few features are selected and/or ranked, it is easier to observe which features are actually related to the targeted mental states. Fourthly, a model with fewer features and consequently fewer parameters can produce faster predictions for a new sample, as it should be computationally more efficient. Fifthly, collection and storage of data will be reduced. Three feature selection approaches have been identified [106]: the filter, wrapper and embedded approaches. Many alternative methods have been proposed for each approach.

Filter methods rely on measures of relationship between each feature and the target class, independently of the classifier to be used. The coefficient of determination, which is the square of the estimation of the Pearson correlation coefficient, can be used as a feature ranking criterion [85]. The coefficient of determination can also be used for a two-class problem, labelling classes as -1 or +1. The correlation coefficient can only detect linear dependencies between features and classes though. To exploit non-linear relationships, a simple solution is to apply non-linear pre-processing, such as taking the square or the log of the features. Ranking criteria based on information theory can also be used e.g. the mutual information between each feature and the target variable [82, 180]. Many filter feature selection approaches require estimations of the probability densities and the joint density of the feature and class label from the data. One solution is to discretize the features and class labels. Another solution is to approximate their densities with a non-parametric method such as Parzen windows [179]. If the densities are estimated by a normal distribution, the result obtained by the mutual information will be similar to the one obtained by the correlation coefficient. Filter approaches have a linear complexity with respect to the number of features. However, this may lead to a selection of redundant features [106].

Wrapper and embedded approaches solve this problem at the cost of a longer computation time. These approaches use a classifier to obtain a subset of features. Wrapper methods select a subset of features, present it as input to a classifier for training, observe the resulting performance and stop the search according to a stopping criterion or

propose a new subset if the criterion is not satisfied. Embedded methods integrate the features selection and the evaluation in a unique process, e.g. in a decision tree [27, 184] or a multilayer perceptron with optimal cell damage [37].

Feature selection has provided important improvements in BCI, e.g., the stepwise Linear Discriminant Analysis (embedded method) for P300-BCI [111] and frequency bands selection for motor imagery using maximal mutual information (filtering methods) [7]. Let us also mention the Support Vector Machine for channel selection [115], linear regressor for knowledge extraction [123], genetic algorithms for spectral feature selection [50] and P300-based feature selection [201], or evolutionary algorithms for feature selection based on multiresolution analysis [176] (all being wrapper methods). Indeed, metaheuristic techniques (also including ant colony, swarm search, tabu search and simulated annealing) [152] are becoming more and more frequently used for feature selection in BCI [174] in order to avoid the curse-of-dimensionality.

Other popular methods used in EEG-based BCIs notably include filter methods such as maximum Relevance Minimum Redundancy (mRMR) feature selection [180, 166] or R^2 feature selection [217, 169]. It should be mentioned that five feature selection methods, namely information gain ranking, correlation-based feature selection, Relief (an instance-based feature ranking method for multiclass problems), consistency-based feature selection and 1R Ranking (one-rule classification) have been evaluated on the BCI competition III data sets [107]. Amongst 10 classifiers, the top three feature selection methods were correlation-based feature selection, information gain and 1R ranking, respectively.

2.3. Performance Measures

To evaluate BCI performance, one must bear in mind that different components of the BCI loop are at stake [212]. Regarding the classifier alone, the most basic performance measure is the classification accuracy. This is valid only if the classes are balanced [66], i.e. with the same number of samples per class and if the classifier is unbiased, i.e. it has the same performance for each class [199]. If these conditions are not met, the Kappa metric or the confusion matrix are more informative performance measures [66]. The sensitivity-specificity pair, or precision, can be computed from the confusion matrix. When the classification depends on a continuous parameter (e.g. a threshold), the Receiver Operating Characteristic (ROC) curve, and the Area Under the Curve (AUC) are often used.

Classifier performance is generally computed offline on pre-recorded data, using a hold-out strategy: some datasets are set aside to be used for the evaluation, and are not part of the training dataset. However, some authors also report cross-validation measures estimated on training data, which may over-rate the performance.

The contribution of classifier performance to overall BCI performance strongly depends on the orchestration of the BCI subcomponents. This orchestration is highly

variable given the variety of BCI systems (co-adaptive, hybrid, passive, self- or system-paced). The reader is referred to [212] for a comprehensive review of evaluation strategies in such BCI contexts.

3. Past methods and current challenges

3.1. A brief overview of methods used 10 years ago

In our original review of classification algorithms for EEG-based BCIs published ten years ago, we identified five main families of classifiers that had been explored: linear classifiers, neural networks, non-linear Bayesian classifiers, nearest neighbour classifiers and classifier combinations [141].

Linear classifiers gather discriminant classifiers that use linear decision boundaries between the feature vectors of each class. They include Linear Discriminant Analysis (LDA), regularized LDA and Support Vector Machines (SVMs). Both LDA and SVM were, and still are, the most popular types of classifiers for EEG based-BCIs, particularly for online and real-time BCIs. The previous review highlighted that in terms of performances, SVM often outperformed other classifiers.

Neural Networks (NN) are assemblies of artificial neurons, arranged in layers, which can be used to approximate any non-linear decision boundary. The most common type of NN used for BCI at that time was the Multi-Layer Perceptron (MLP), typically employing only one or two hidden layers. Other NN types were explored more marginally, such as the Gaussian classifier NN or Learning Vector Quantization (LVQ) NN.

Non-linear Bayesian classifiers are classifiers modeling the probability distributions of each class and use Bayes rule to select the class to assign to the current feature vector. Such classifiers notably include Bayes quadratic classifiers and Hidden Markov Models (HMMs).

Nearest neighbour classifiers assign a class to the current feature vector according to its nearest neighbours. Such neighbours could be training feature vectors or class prototypes. Such classifiers include the k-Nearest Neighbour (kNN) algorithm or Mahalanobis distance classifiers.

Finally, classifier combinations are algorithms combining multiple classifiers, either by combining their outputs and/or by training them in ways that maximize their complementarity. Classifier combinations used for BCI at the time included boosting, voting or stacking combination algorithms. Classifier combination appeared to be amongst the best performing classifiers for EEG based BCIs, at least in offline evaluations.

3.2. Challenges faced by current EEG signal classification methods

Ten years ago, most classifiers explored for BCI were rather standard classifiers used in multiple machine learning problems. Since then, research efforts have focused on

identifying and designing classification methods dedicated to the specificities of EEG-based BCIs. In particular, the main challenges faced by classification methods for BCI are the low signal-to-noise ratio of EEG signals [172, 228], their non-stationarity over time, within or between users, where same-user EEG signals varying between or even within runs [202, 80, 109, 164, 145, 56], the limited amount of training data that is generally available to calibrate the classifiers [108, 137], and the overall low reliability and performance of current BCIs [139, 138, 229, 109].

Therefore, most of the algorithms studied these past 10 years aimed at addressing one or more of these challenges. More precisely, adaptive classifiers whose parameters are incrementally updated online were developed to deal with EEG non-stationarity in order to track changes in EEG properties over time. Adaptive classifiers can also be used to deal with limited training data by learning online, thus requiring fewer offline training data. Transfer learning techniques aim at transferring features or classifiers from one domain, e.g., BCI subjects or sessions, to another domain, e.g., other subjects or other sessions from the same subject. As such they also aim at addressing within or between-subjects non-stationarity and limited training data by complementing the few training data available with data transferred from other domains. Finally in order to compensate for the low EEG signal-to-noise ratio and the poor reliability of current BCIs, new methods were explored to process and classify signals in a single step by merging feature extraction, feature selection and classification. This was achieved by using matrix (notably Riemannian methods) and tensor classifiers as well as deep learning. Additional methods explored were targeted specifically at learning from limited amount of data and at dealing with multiple class problems. We describe these new families of methods in the following.

4. New EEG classification methods since 2007

4.1. Adaptive classifiers

4.1.1. Principles

Adaptive classifiers are classifiers whose parameters, e.g. the weights attributed to each feature in a linear discriminant hyperplane, are incrementally re-estimated and updated over time as new EEG data become available [202, 200]. This enables the classifier to track possibly changing feature distribution, and thus to remain effective even with non-stationary signals such as an EEG. Adaptive classifiers for BCI were first proposed in the mid-2000's, e.g., in [72, 202, 30, 209, 163], and were shown to be promising in offline analysis. Since then, more advanced adaptation techniques have been proposed and tested, including online experiments.

Adaptive classifiers can employ both supervised and unsupervised adaptation, i.e. with or without knowledge of the true class labels of the incoming data, respectively. With supervised adaptation, the true class labels of the incoming EEG signals is known

and the classifier is retrained on the available training data augmented with these new, labelled incoming data, or is updated based on this new data only [202, 200]. Supervised BCI adaptation requires guided user training, for which the users' commands are imposed and thus the corresponding EEG class labels are known. Supervised adaptation is not possible with free BCI use, as the incoming EEG data true label is unknown. With unsupervised adaptation, the label of the incoming EEG data is unknown. As such, unsupervised adaptation is based on an estimation of the data class labels for retraining/updating, as discussed in [104], or is based on class-unspecific adaptation, e.g. the general all classesEEG data mean [219, 24] or a covariance matrix [238] is updated in the classifier model. A third type of adaptation, in between supervised and unsupervised methods, has also been explored: semi-supervised adaptation [122, 121]. Semi-supervised adaptation consists of using both initial labelled data and incoming unlabelled data to adapt the classifier. For BCI, semi-supervised adaptation is typically performed by 1) initially training a supervised classifier on available labelled training data, then 2) by estimating the labels of incoming unlabelled data with this classifier, and 3) by adapting/retraining the classifier using these initially unlabelled data assigned to their estimated labels combined with the known available labelled training data. This process is repeated as new batches of unlabelled incoming EEG data become available.

4.1.2. State-of-the-art

So far, the majority of the work on adaptive classifiers for BCI has been based on supervised adaptation. Multiple adaptive classifiers were explored offline, such as LDA or Quadratic Discriminant Analysis (QDA) [200] for motor imagery-based BCI. An adaptive LDA was also proposed based on Kalman Filtering to track the distribution of each class [96]. In order to deal with possibly imperfect labels in supervised adaptation, [236] proposed and evaluated offline an adaptive Bayesian classifier based on Sequential Monte Carlo sampling that explicitly models uncertainty in the observed labels. For ERP-based BCI, [227] explored an offline adaptive Support Vector Machine (SVM), adaptive LDA, a stochastic gradient-based adaptive linear classifier, and online Passive-Aggressive (PA) algorithms. Interestingly, McFarland and colleagues demonstrated in offline analysis of EEG data over multiple sessions that continuously retraining the weights of linear classifiers in a supervised manner improved the performance of Sensori-Motor Rhythms (SMR)-based BCI, but not of the P300-based BCI speller [159]. However, results presented in [197] suggested that continuous adaption was beneficial for the asynchronous P300-BCI speller, and [227] suggested the same for passive BCI based on the P300.

Online, still using supervised adaptation, both adaptive LDA and QDA have been explored successfully in [222]. In [86], an adaptive probabilistic Neural Network was also used for online adaptation with a motor imagery-BCI. Such a classifier models the feature distributions of each class in non-parametric fashion, and updates them as new trials become available. Classifier ensembles were also explored to create adaptive

classifiers. In [119], a dynamic ensemble of five SVM classifiers was created by training a new SVM for each batch of new incoming labelled EEG trials, adding it to the ensemble and removing the oldest SVM. Classification was performed using a weighted sum of each SVM output. This approach was shown online to be superior to a static classifier.

Regarding supervised adaptation, it should be mentioned that adaptive spatial filters were also proposed, notably several variants of adaptive CSP [247, 204], but also adaptive xDAWN [227].

Unsupervised adaptation of classifiers is obviously much more difficult, as the class labels, hence the class-specific variability, is unknown. Thus, unsupervised methods have been proposed to estimate the class labels of new incoming samples before adapting the classifier based on this estimation. This technique was explored offline in [24] and [129], and online in [83] for an LDA classifier and Gaussian Mixture Model (GMM) estimation of the incoming class labels, with motor imagery data. Offline, Fuzzy C-means (FCM) were also explored instead of GMM to track the class means and covariance for an LDA classifier [130]. Similarly, a non-linear Bayesian classifier was adapted using either unsupervised or semi-supervised learning (i.e. only some of the incoming trials were labelled) using extended Kalman filtering to track the changes in the class distribution parameters with Auto-Regressive (AR) features [149]. Another simple unsupervised adaptation of the LDA classifier for motor imagery data was proposed and evaluated for both offline and online data [219]. The idea was to not incrementally adapt all of the LDA parameters, but only its bias, which can be estimated without knowing the class labels if we know that the data is balanced, i.e. with the same number of trials per class on average. This approach was extended to the multiclass LDA case, and evaluated in an offline scenario in [132].

Adaptation can be performed according to reinforcement signals (RS), indicating whether a trial was erroneously classified by the BCI. Such reinforcement signals can be deduced from Error-related Potentials (ErrP), potentials appearing following a perceived error which may have been committed by either the user or the machine [68]. In [133], an incremental logistic regression classifier was proposed, which was updated along the error gradient when a trial was judged to be misclassified according to the detection of an ErrP. The strength of the classifier update was also proportional to the probability of this ErrP. A Gaussian probabilistic classifier incorporating an RS was later proposed in [131], in which the update rules of the mean and covariance of each class depend on the probability of the RS. This classifier could thus incorporate a supervised, unsupervised or semi-supervised adaptation mode, according to whether the probability of the RS is always correct as either 0 or 1 (supervised case), uniform, i.e. uninformative (unsupervised case) or with a continuous probability with some uncertainty (partially supervised case). Using simulated supervised RS, this method was shown to be superior to static LDA and the other supervised and unsupervised adaptive LDA discussed above [131]. Evaluations with real-world data remain to be performed. Also using ErrP in offline simulations of an adaptive movement-related potential (MRP)-BCI, [9] augmented the training set with incoming trials, but only with those that were classified

correctly, as determined by the absence of an ErrP following feedback to the user. They also removed the oldest trials from the training set as new trials became available. Then, the parameters of the classifier, an incremental SVM, were updated based on the updated training set. ErrP-based classifier adaptation was explored online for code-modulated visual evoked potential (c-VEP) classification in [206]. In this work, the label of the incoming trial was estimated as the one decided by the classifier if no ErrP was detected, the opposite label otherwise (for binary classification). Then, this newly labelled trial was added to the training set, and the classifier and spatial filter, a one-class SVM and Canonical Correlation Analysis (CCA), respectively, were retrained on the new data. Finally, [239] demonstrated that classifier adaptation based on RS could also be performed using classifier confidence, and that such adaptation was beneficial to P300-BCI.

For ERP-based BCI, semi-supervised adaptation was explored with SVM and enabled the calibration of a P300-speller with less data as compared to a fixed, non-adaptive classifier [122, 151]. This method was later tested and validated online in [81]. For P300-BCI, a co-training semi-supervised adaptation was performed in [178]. In this work, two classifiers were used: a Bayesian LDA and a standard LDA. Each was initially trained on training labelled data, and then used to estimate the labels of unlabelled incoming data. The latter were labelled with their estimated class label and used as additional training data to retrain the other classifier, hence the co-training. This semi-supervised approach was shown offline to lead to higher bit-rates than a fully supervised method, which requires more supervised training data. On the other hand, offline semi-supervised adaptation with an LDA as classifier failed on mental imagery data, probably owing to the poor robustness of the LDA to mislabelling [137]. Finally, both for offline and online data, [104, 105] proposed a probabilistic method to adaptively estimate the parameters of a linear classifier in P300-based spellers, which led to a drastic reduction in calibration time, essentially removing the need for the initial calibration. This method exploited the specific structure of the P300-speller, and notably the frequency of samples from each class at each time, to estimate the probability of the most likely class label. In a related work, [78] proposed a generic method to adaptively estimate the parameters of the classifier without knowing the true class labels by exploiting any structure that the application may have. Semi-supervised adaptation was also used offline for multi-class motor imagery with a Kernel Discriminant Analysis (KDA) classifier in [171]. This method has shown its superiority over non-adaptive methods, as well as over adaptive unsupervised LDA methods.

Vidaurre *et al.*, also explored co-adaptive training, where both the machine and the user are continuously learning, by using adaptive features and an adaptive LDA classifier [221, 220]. This enabled some users who were initially unable to control the BCI to achieve better than chance classification performances. This work was later refined in [64] by using a simpler but fully adaptive setup with auto-calibration, which proved to be effective both for healthy users and for users with disabilities [63]. Co-adaptive training, using adaptive CSP patches, proved to be even more efficient [196].

Adaptive classification approaches used in BCI are summarized in Tables 1 and 2, for supervised and unsupervised methods, respectively.

4.1.3. Pros and cons Adaptive classifiers were repeatedly shown to be superior to non-adaptive ones for multiple types of BCI, notably motor-imagery BCI, but also for some ERP-based BCI. To the best of our knowledge, adaptive classifiers have apparently not been explored for SSVEP-BCI. Naturally, supervised adaptation is the most efficient type of adaptation, as it has access to the real labels. Nonetheless unsupervised adaptation has been shown to be superior to static classifiers in multiple studies [24, 130, 149, 219, 132]. It can also be used to shorten or even remove the need for calibration [122, 151, 81, 105, 78]. There is a need for more robust unsupervised adaptation methods, as the majority of actual BCI applications do not provide labels, and thus can only rely on unsupervised methods.

For unsupervised adaptation, reward signals, and notably ErrP, have been exploited in multiple papers (e.g. [206, 239, 9]). Note however, that ErrP decoding from EEG signals may be a difficult task. Indeed, [157] demonstrated that the decoding accuracy of ErrP was positively correlated with the P300 decoding accuracy. This means that people who make errors in the initial BCI task (here a P300), for whom error correction and ErrP-based adaptation would be the most useful, have a lesser chance that the ErrP will be correctly decoded. There is thus a need to identify robust reward signals.

Only a few of the proposed methods were actually used online. For unsupervised methods, a simple and effective one that demonstrated its value online in several studies is adaptive LDA, proposed by Vidaurre *et al.* [219]. This and other methods that are based on incremental adaptation (i.e., updating the algorithms parameters rather than fully re-optimizing them) generally have a computational complexity that is low enough to be used online. Adaptive methods that require fully retraining the classifier with new incoming data generally have a much higher computationnal complexity (e.g., regularly retraining an SVM from scratch in real-time requires a lot of computing power) which might prevent them from being actually used online.

However, more online studies are clearly necessary to determine how adaptation should be performed in practice, with a user in the loop. This is particularly important for mental imagery BCI in which human-learning is involved [170, 147]. Indeed, because the user is adapting to the BCI by learning how to perform mental imagery tasks so that they are recognized by the classifier, adaptation may not always help and may even be confusing to the user, as it may lead to continuously-changing feedback. Both machine and human learning may not necessarily converge to a suitable and stable solution. A recent theoretical model of this two-learner problem was proposed in [168], and indicated that adaptation that is either too fast or too slow can actually be detrimental to user learning. There is thus a need to design adaptive classifiers that ensure and favour human learning.

Table 1. Summary of adaptive supervised classification methods explored offline

EEG Pattern	Features	Classifier	References
Motor Imagery	band power	adaptive LDA/QDA	[200]
Motor Imagery	Fractal Dimension	adaptive LDA	[96]
Motor Imagery	band power	adaptive LDA/QDA	[222]
Motor Imagery	band power	adaptive probabilistic NN	[86]
Motor Imagery	CSP	dynamic SVM ensemble	[119]
Motor Imagery	adaptive CSP	SVM	[247, 204]
Motor execution	AR parameters	adaptive Gaussian classifier	[236]
P300	Time points with adaptive xDAWN	adaptive LDA/SVM online PA classifier	[227]

Table 2. Summary of adaptive unsupervised classification methods explored

EEG Pattern	Features	Classifier	References
Motor Imagery	band power	adaptive LDA with GMM	[24, 129, 83]
Motor Imagery	band power	adaptive LDA with FCM	[130]
Motor Execution	AR parameters	adaptive Gaussian classifier	[149]
Motor Imagery	Band Power	adaptive LDA	[219][132]
Motor Imagery	Band Power	Adaptive Gaussian classifier	[131]
Motor Imagery	Band Power	semi-supervised CSP+LDA	[137]
Motor Imagery	adaptive Band Power	adaptive LDA	[221, 220, 64, 63]
Motor Imagery	adaptive CSP patches	adaptive LDA	[196]
Covert Attention	Band Power	incremental logistic regression	[133]
MRP	Band Power	incremental SVM	[9]
c-VEP	CCA	adaptive One-class SVM	[206]
P300	Time Points	SWLDA	[239]
P300	Time Points	semi-supervised SVM	[122, 151, 81]
P300	Time Points	co-training LDA	[178]
P300	Time Points	unsupervised Linear classifier	[104, 105]
ErrP	Time Points	unsupervised Linear classifier	[78]

4.2. Classifying EEG matrices and tensors

4.2.1. Riemannian geometry-based classification

Principles:

The introduction of Riemannian geometry in the field of BCI has challenged some of the conventions adopted in the classic classification approaches; instead of estimating spatial filters and/or select features, the idea of a Riemannian geometry classifier (RGC) is to map the data directly onto a geometrical space equipped with a suitable metric.

In such a space, data can be easily manipulated for several purposes, such as averaging, smoothing, interpolating, extrapolating and classifying. For example, in the case of EEG data, mapping entails computing some form of *covariance matrix* of the data. The principle of this mapping is based on the assumption that the power and the spatial distribution of EEG sources can be considered fixed for a given mental state and such information can be coded by a covariance matrix. Riemannian geometry studies smooth curved spaces that can be locally and linearly approximated. The curved space is named a *manifold* and its linear approximation at each point is the *tangent space*. In a Riemannian manifold the tangent space is equipped with an inner product (metric) smoothly varying from point to point. This results in a non-Euclidean notion of distance between any two points (*e.g.* each point may be a trial) and a consequent notion of centre of mass of any number of points (Fig. 2). Therefore, instead of using the Euclidean distance, called the *extrinsic* distance, an *intrinsic* distance is used, which is adapted to the geometry of the manifold, and thus to the manner in which the data have been mapped [47, 232].

Amongst the most common matrix manifolds used for BCI applications, we encountered the manifold of Hermitian or symmetric positive definite (SPD) matrices [19] when dealing with covariance matrices estimated from EEG trials, and the Stiefel and Grassmann manifolds [62] when dealing with subspaces or orthogonal matrices. Several machine learning problems can be readily extended to those manifolds by taking advantage of their geometrical constraints (*i.e. learning on manifold*). Furthermore, optimization problems can be formulated specifically on such spaces, which is leading to several new optimization methods and to the solution of new problems [2]. Although related, manifold learning, which consists of empirically attempting to locate the non-linear subspace in which a dataset is defined, is different in concept and will not be covered in this paper. To illustrate these notions, consider the case of SPD

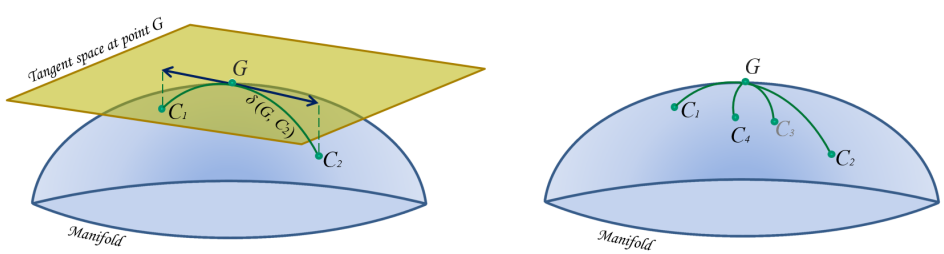

Figure 2. Schematic representation of a Riemannian manifold. EEG trials are represented by points. Left: Representation of the tangent space at point $\mathbf{G}$. The shortest path on the manifold relying on two points $\mathbf{C}_1$ and $\mathbf{C}_2$ is named the geodesic and its length is the Riemannian distance between them. Curves on the manifolds through a point are mapped on the tangent space as straight lines (local approximation). Right: $\mathbf{G}$ represents the centre of mass (mean) of points $\mathbf{C}_1$, $\mathbf{C}_2$, $\mathbf{C}_3$ and $\mathbf{C}_4$. It is defined as the point minimizing the sum of the squared distance between itself and the four points. The centre of mass is often used in RGCs as a representative for a given class.

matrices. The square of the intrinsic distance between two SPD matrices $\mathbf{C}_1$ and $\mathbf{C}_2$ has a closed-form expression given by

$$\delta^2\left(\mathbf{C}_1, \mathbf{C}_2\right) = \sum_n log^2 \lambda_n \left(\mathbf{C}_1^{-1}\mathbf{C}_2\right), \tag{1}$$

where $\lambda_n(\mathbf{M})$ denotes the nth eigenvalue of matrix $\mathbf{M}$. For $\mathbf{C}_1$ and $\mathbf{C}_2$ SPDs, this distance is non-negative, symmetric and is equal to zero if and only if $\mathbf{C}_1 = \mathbf{C}_2$. Interestingly, when $\mathbf{C}_1$ and $\mathbf{C}_2$ are the means of two classes, the eigenvectors of matrix $(\mathbf{C}_1^{-1}\mathbf{C}_2)$ are used to define CSP filters, while its eigenvalues are used for computing their Riemannian distance [47]. Using the distance in Eq. 1, the centre of mass $\mathbf{G}$ of a set $\{\mathbf{C}_1, ..., \mathbf{C}_K\}$ of K SPD matrices (Fig. 3), also called the *geometric mean*, is the unique solution to the following optimization problem

$$\underset{\mathbf{G}}{\text{argmin}} \sum_k \delta^2(\mathbf{C}_k, \mathbf{G}). \tag{2}$$

As discussed thoroughly in [47], this definition is analogous to the definition of the arithmetic mean $1/K \sum_k \mathbf{C}_k$, which is the solution of the optimization problem (2) when the Euclidean distance is used instead of the Riemannian one. In contrast to the arithmetic mean, the geometric mean does not have a closed-form solution. A fast and robust iterative algorithm for computing the geometric mean has been presented in [48]. The simplest RGC methods allow immediate classification of trials (mapped via

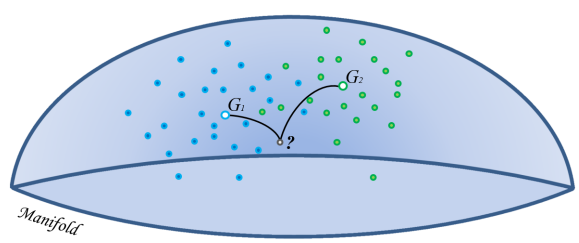

Figure 3. Schematic of the Riemannian minimum distance to mean (RMDM) classifier for a two-class problem. From training data a centre of mass for each class is computed ($\mathbf{G}_1$ and $\mathbf{G}_1$). An unlabelled trial (question mark) is then assigned to the class whose centre of mass is the closest, $\mathbf{G}_1$ in this example. The RMDM works in the same manner for any dimension of the data, any number of classes and any BCI paradigm. It does not require any spatial filtering and feature selection, nor any parameter tuning (see text).

some form of covariance matrix) by simple nearest neighbour methods, using exclusively the notion of Riemannian distance (Eq. 1), and possibly with the notion of geometric mean (2). For instance, the Riemannian minimum distance to mean (RMDM) classifier [13, 15] computes a geometric mean for each class using training data and then assigns an unlabelled trial to the class corresponding to the closest mean (Fig. 3). Another class of RGCs consists of methods projecting the data points to a tangent space followed by a classification, thereafter using standard classifiers such as LDA, SVM, logistic regression, etc. [13, 14]. These methods take advantage of both the Riemannian geometry and

the possibility of executing complex decision functions using dedicated classifiers. An alternative approach is to project the data in the tangent space, filter the data there (for example by LDA), and map the data back onto the manifold to finally carry out the RMDM.

State-of-the-art:

As described above, Riemannian classifiers either operate directly on the manifold (e.g., the RMDM) or by the projection of the data in the tangent space. Simple RGCs on the manifold have been shown to be competitive as compared to previous state-of-the-art classifiers used in BCI as long as the number of electrodes is not very large, providing better robustness to noise and better generalization capabilities, both on healthy users [13, 46, 100] and clinical populations [158]. RGCs based on tangent space projection clearly outperformed the other state-of-the-art methods in terms of accuracy [13, 14], as demonstrated by the first place they have been awarded in five recent international BCI predictive modelling data competitions, as reported in [47]. For a comprehensive review of the Riemannian approaches in BCI, the reader can refer to [47, 232]. The various approaches using Riemannian Geometry classifiers for EEG-based BCIs are summarized in Table 3.

Pros and cons:

As highlighted in [232], the processing procedures of Riemannian approaches such as RMDM is simpler and involves fewer stages than more classic approaches. Also, Riemannian classifiers apply equally well to all BCI paradigms (*e.g.* BCIs based on mental imagery, ERPs and SSVEP); only the manner in which data points are mapped in the SPD manifold differs (see [47] for details). Furthermore, in contrast to most classification methods, the RMDM approach is parameter-free, that is, it does not require any parameter tuning, for example by cross-validation. Hence, Riemannian geometry provides new tools for building simple, more robust and accurate prediction models.

Several reasons have been proposed to advocate the use of the Riemannian geometry. Due to its logarithmic nature the Riemannian distance is robust to extreme values, that is, noise. Also, the intrinsic Riemannian distance for SPD matrices is invariant both to matrix inversion and to any linear invertible transformation of the data, *e.g.* any mixing applied to the EEG sources does not change the distances among the observed covariance matrices. These properties in part explain why Riemannian classification methods provide a good generalization capability [238, 224], which enabled researchers to set up calibration-free adaptive ERP-BCIs using simple subject-to-subject and session-to-session transfer learning strategies [6].

Interestingly, as illustrated in [94], it is possible to not only interpolate along geodesics (Fig. 2) on the SPD manifolds, but also to extrapolate (*e.g.* forecast) without

leaving the manifold and respecting the geometrical constraints. For example, in [99] interpolation has been used for data augmentation by generating artificial covariance matrices along geodesics but extrapolation could also have been used. Often, the Riemannian interpolation is more relevant than its Euclidean counterpart as it does not suffer from the so-called *swelling effect* [232]. This effect describes the fact that a Euclidean interpolation between two SPD matrices does not involve the determinant of the matrix as it should (i.e. the determinant of the Euclidean interpolation can exceed the determinant of the interpolated matrices). In the spirit of [231], the determinant of a covariance matrix can be considered as the volume of the polytope described by the column of the matrix. Thus, a distance that is immune to the swelling effect will respect the shape of the polytope along geodesics.

As Eq. 1 indicates, computing the Riemannian distance between two SPD matrices involves adding squared logarithms, which may cause numerical problems; the smallest eigenvalues of matrix $(\mathbf{C}_1^{-1}\mathbf{C}_2)$ tend towards zero as the number of electrodes increases and/or the window size for estimating $\mathbf{C}_1$ and $\mathbf{C}_2$ decreases, making the logarithm operation ill-conditioned and numerically unstable. Further, note that the larger the dimensions, the more the distance is prone to noise. Moreover, Riemannian approaches usually have high computational complexities (e.g. growing cubically with the number of electrodes for computing both the geometric mean and the Riemannian distance). For these reasons, when the number of electrodes is large with respect to the window size, it is advocated to reduce the dimensions of the input matrices. Classical unsupervised methods such as PCA or supervised methods such as CSP can be used for this purpose. Recently, Riemannian-inspired dimensionality reduction methods have been investigated as well [94, 95, 189].

Interestingly, some approaches have tried to bridge the gap between Riemannian approaches and more classical paradigms by incorporating some Riemannian geometry in approaches such as CSP [233, 12]. CSP was the previous golden standard and is based on a different paradigm than Riemannian geometry. Taking the best of those two paradigms is expected to gain better robustness while compressing the information.

Table 3. Summary of Riemannian Geometry classifiers for EEG-based BCI

EEG Pattern	**Features**	**Classifier**	**References**
Motor Imagery	Band-Pass Covariance	RMDM	[46, 13]
Motor Imagery	Band-Pass Covariance	Tangent Space + LDA	[13, 231]
Motor Imagery	Band-Pass Covariance	SVM Riemannian Kernel	[14]
P300	Special Covariance	RMDM	[46]
P300	Special Covariance	RMDM	[15]
P300	Special Covariance	RMDM	[158]
SSVEP	Band-Pass Covariance	RMDM	[100, 34]

4.2.2. Other matrix classifiers

Principles:

As mentioned previously, the classification pipeline in BCI typically involves spatial filtering of the EEG signals followed by classification of the filtered data. This results in the independent optimization of several sets of parameters, namely for the spatial filters and for the final classifier. For instance, the typical linear classifier decision function for an oscillatory activity BCI would be the following:

$$f(\mathbf{X}, \mathbf{w}, \mathbf{S}) = \sum_i w_i log(var(\mathbf{s}_i^T \mathbf{X})) + w_0 \tag{3}$$

where $\mathbf{X}$ is the EEG signals matrix, $\mathbf{w} = [w_0, w_1, \ldots, w_N]$ is the linear classifier weight vector, and $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N]$ is a matrix of spatial filter $\mathbf{s}_i$. Optimizing w and s_i separately may thus lead to suboptimal solutions, as the spatial filters do not consider the objective function of the classifier. Therefore, in addition to RGC, several authors have shown that it is possible to formulate this dual optimization problem as a single one, where the parameters of the spatial filters and the linear classifier are optimized simultaneously, with the potential to obtain improved performance. The key principle of these approaches is to learn classifiers (either linear vector classifiers or matrix classifiers) that directly use covariance matrices as input, or their vectorised version. We briefly present these approaches below.

State-of-the-art:

In [214], the EEG data were represented as an augmented covariance matrix $\mathbf{A}$, containing as block diagonal terms both the first order term $\mathbf{X}$, i.e. the signal time course, and as second order terms the covariance matrices of EEG trials band-pass filtered in various frequency bands. The learned classifier is thus a matrix of weights $\mathbf{W}$ (rather than a vector), with the decision function $f(\mathbf{A}, \mathbf{W}) =< \mathbf{A}, \mathbf{W} > +b$. Due to the large dimensionality of the augmented covariance matrix, a matrix regularization term is necessary with such classifiers, e.g., to obtain sparse temporal or spatial weights. Note that this approach can be applied to both ERP and oscillatory-based BCI, as the first order terms capture the temporal variation, and the covariance matrices capture the EEG signals band power variations.

Following similar ideas in parallel, [65] represented this learning problem in tensor space by constructing tensors of frequency-band specific covariance matrices, which can then be classified using a linear classifier as well, provided appropriate regularization is used.

Finally, [190] demonstrated that equation 3 can be rewritten as follows, if we drop the log-transform:

$$f(\Sigma, \mathbf{w}_\Sigma) = vec(\mathbf{\Sigma})^T \mathbf{w}_\Sigma + w_0 \tag{4}$$

with $\mathbf{w}_\Sigma = \sum_i w_i vec(\mathbf{s}_i \mathbf{s}_i^T)$, $\mathbf{\Sigma} = \mathbf{X}^T \mathbf{X}$ being the EEG covariance matrix, and $vec(\mathbf{M})$ being the vectorisation of matrix $\mathbf{M}$. Thus, equation 3 can be optimized directly in

the space of vectorised covariance matrices by optimizing the weights $\mathbf{w}_\Sigma$. Here as well, owing to the usually large dimensionality of $vec(\sigma)$, appropriate regularization is necessary, and [190] explored different approaches to do so.

These different approaches all demonstrated higher performance than the basic CSP+LDA methods on motor imagery data sets [65, 214, 190]. This suggests that such formulations can be worthy alternatives to the standard CSP+LDA pipelines.

Pros and cons:

By simultaneously optimizing spatial filters and classifiers, such formulations usually achieve better solutions than the independent optimization of individual sets of components. Their main advantage is thus increased classification performance. This formulation nonetheless comes at the expense of a larger number of classifier weights due to the high increase in dimensionality of the input features (covariance matrix with $(N_c * (N_c + 1))/2$ unique values versus N_c values when using only the channels band power). Appropriate regularization is thus necessary. It remains to be evaluated how such methods perform for various amounts of training data, as they are bound to suffer more severely from the curse of dimensionality than simpler methods with fewer parameters. These methods have also not been used online to date. From a computational complexity point of view, such methods are more demanding than traditionnal methods given their increased number of parameters, as mentioned above. They also generally require heavy regularization, which can make their calibration longer. However, their decision functions being linear, they should be easily applicable in online scenarios. However, it remains to be seen whether they can be calibrated quickly enough for online use, and what their performance will be for online data.

4.2.3. Feature extraction and classification using tensors

Principles:

Tensors (i.e., multi-way arrays) provide a natural representation for EEG data, and higher order tensor decompositions and factorizations are emerging as promising (but not yet very well established and not yet fully explored) tools for analysis of EEG data; particularly for feature extraction, clustering and classification tasks in BCI [42, 43, 38, 40, 39].

The concept of tensorization refers to the generation of higher-order structured tensors (multiway arrays) from lower-order data formats, especially time series EEG data represented as vectors or organized as matrices. This is an essential step prior to tensor (multiway) feature extraction and classification [42, 41, 182].

The order of a tensor is the number of modes, also known as ways or dimensions (*e.g.* for EEG BCI data: space (channels), time, frequency, subjects, trials, groups, conditions, wavelets, dictionaries). In the simplest scenario, multichannel EEG signals

can be represented as a 3rd-order tensor that has three physical modes: space (channel) x time x frequency. In other words, S channels of EEG which are recorded over T time samples, can produce S matrices of $F \times T$ dimensional time-frequency spectrograms stacked together into an $F \times T \times S$ dimensional third-order tensor. For multiple trials and multiple subjects, the EEG data sets can be naturally represented by higher-order tensors: e.g., for a 5th-order tensor: space x time x frequency x trial x subject.

It should be noted that almost all basic vector- and matrix-based machine learning algorithms for feature extraction and classification have been or can be extended or generalized to tensors. For example, the SVM for classification has been naturally generalized to the Tensor Support Machine (TSM), Kernel TSM and Higher Rank TSM. Furthermore, the standard LDA method has been generalized to Tensor Fisher Discriminant Analysis (TFDA) and/or Higher Order Discriminant Analysis (HODA) [183, 41]. Moreover Tensor representations of BCI data are often very useful in mitigating the small sample size problem in discriminative subspace selection, because the information about the structure of data is often inherent in tensors and is a natural constraint which helps reduce the number of unknown feature parameters in the description of a learning model. In other words, when the number of EEG training measurements is limited, tensor-based learning machines are expected often to perform better than the corresponding vector- or matrix-based learning machines, as vector representations are associated with problems such as loss of information for structured data and over-fitting for high-dimensional data.

State-of-the-art:

To ensure that the reduced data sets contain maximum information about input EEG data, we may apply constrained tensor decomposition methods. For example, this could be achieved on the basis of orthogonal or non-negative tensor (multi-array) decompositions, or Higher Order (multilinear) Discriminant Analysis (HODA), whereby input data are considered as tensors instead of more conventional vector or matrix representations. In fact, tensor decomposition models, especially PARAFAC (also called CP decomposition), TUCKER, Hierarchical Tucker (HT) and Tensor Train (TT) are alternative sophisticated tools for feature extraction problems by capturing multi-linear and multi-aspect structures in large-scale higher-order data-sets [183, 39]. Using this type of approach, we first decompose multi-way data using TUCKER or CP decompositions, usually by imposing specific constraints (smoothness, sparseness, non-negativity), in order to retrieve basis factors and significant features from factor (component) matrices. For example, wavelets/dictionaries allow us to represent the data often in a more efficient way, i.e. a sparse manner with different sparsity profiles [43, 183].

Moreover, in order to increase performance of BCI classification, we can apply two or more time-frequency representations or the same frequency transform but with two or more different parameter settings. Different frequency transforms (or different mother

wavelets) allow us to obtain different sparse tensor representations with various sparsity profiles and some complimentary information. For multichannel EEG signals we can generate a block of at least two tensors, which can be concatenated as a single data tensor: space x time x frequency x trial [183, 43, 182].

The key problem in tensor representation is the choice of a suitable Time-Frequency Representation (TFR) or frequency transform and the selection of optimal, or close to optimal, corresponding transformation parameters. By exploiting various TFRs, possibly with suitably selected different parameter settings for the same data, we may potentially improve the classification accuracy of BCI due to additional (partially redundant) information. Such approaches have been implemented e.g. for motor imagery (MI) BCI by employing different complex Morlet (Gabor) wavelets for EEG data sets with 62 channels [183]. For such data sets, the authors selected different complex Morlet wavelets with two different bandwidth frequency parameters fb = 1 Hz and fb = 6 Hz for the same centre frequency fc = 1 Hz. For each mother wavelet the authors constructed a 4th-order tensor: 62-channels x 23-frequency bins x 50-time frames x 120-trials for both training and test EEG data. The block of training tensor data can be concatenated as the 5th-order tensor: 62-channels x 23-frequency bins x 50-time frames x 2-wavelets x 120-trials.

The HODA algorithm was used to estimate discriminant bases. The four most significant features were selected to classify the data, and led to an improved accuracy higher than 95%. Thus, it appears that by applying tensor decomposition for suitably constructed data tensors, considerable performance improvement in comparison to the standard approaches can be achieved for both motor-imagery BCI [183, 223] and P300 paradigms [175].

In this approach, transformation of data with a dictionary aims to un-correlate the raw data and express them in a sparse domain. Different dictionaries (transformations) contribute to obtaining different sparse representations with various sparsity profiles. Moreover, augmentation of dimensionality to create samples with additional modes improved the performance.

To summarize, tensor decompositions with nonnegative, orthonormal or discriminant bases improved the classification accuracy for the BCI dataset by almost 10%. A comparison of all methods mentioned is provided in Table 4.

From the time-frequency analysis perspective, tensor decompositions are very attractive, even for a single channel, because they simultaneously take into account temporal and spectral information and variability and/or consistency of Time Frequency Representations (TFRs) for trials and/or subjects. Furthermore, they provide links among various latent (hidden) variables (e.g., temporal, spectral and spatial components) often with physical or physiological meanings and interpretations [40, 183].

Furthermore, standard Canonical Correlation analysis (CCA) was generalized to tensor CCA and multiset CCA and was successfully applied to the classification of SSVEP for BCI [246, 242, 243, 245]. Tensor Canonical correlation analysis (TCCA) and its modification multiset canonical correlation analysis (MsetCCA) have been

one of the most efficient methods for frequency recognition in SSVEP-BCIs. The MsetCCA method learns multiple linear transforms that implement joint spatial filtering to maximize the overall correlation amongst canonical variates, and hence extracts SSVEP common features from multiple sets of EEG data recorded at the same stimulus frequency. The optimized reference signals are formed by combination of the common features and are completely based on training data. Extensive experimental study with EEG data demonstrated that the tensor and MsetCCA method improve the recognition accuracy of SSVEP frequency in comparison with the standard CCA method and other existing methods, especially for a small number of channels and a short time window length. The superior results indicate that the tensor MsetCCA method is a very promising candidate for frequency recognition in SSVEP-based BCIs [243].

Pros and cons:

In summary, the recent advances in BCI technologies have generated massive amounts of brain data exhibiting high dimensionality, multiple modality (e.g., physical modes such as frequency or time, multiple brain imaging techniques or conditions), and multiple couplings as functional connectivity data. By virtue of their multi-way nature, tensors provide powerful and promising tools for BCI analysis and fusion of massive data combined with a mathematical backbone for the discovery of underlying hidden complex (space-time-frequency) data structures [42, 183].

Another of their advantages is that, using tensorization and low-rank tensor decomposition, they can efficiently compress large multidimensional data into low-order factor matrices and/or core tensors which usually represent reduced features. Tensor methods can also analyze linked (coupled) blocks of trials represented as large-scale matrices into the form of tensors in order to separate common/correlated from independent/uncorrelated components in the observed raw EEG data.

Finally, it is worth mentioning that tensor decompositions are emerging techniques not only for feature extraction/selection and BCI classification, but also for pattern recognition, multiway clustering, sparse representation, data fusion, dimensionality reduction, coding, and multilinear blind brain source separation (MBSS). They can potentially provide convenient multi-channel and multi-subject space-time-frequency sparse representations, artefact rejection, feature extraction, multi-way clustering and coherence tracking [40, 39].

On the cons side, the complexity of tensor methods is usually much higher than standard matrix and vector machine learning methods. Moreover, since tensor methods are just emerging as potential tools for feature extraction and classification, existing algorithms are not always mature and are still not fully optimized. Thus, some efforts are still needed to optimize and test them for real-life large scale data sets.

Table 4. Summary of Tensor Classifiers for EEG-based BCI

EEG Pattern	Features/Methods	Classifier	References
Motor Imagery	Topographic map, TFR, Connect.	LDA/HODA	[183]
P300	Multilinear PCA	SVM/TSM	[223]
P300	Tine-Space-Freq.	HODA	[175]
SSVEP	TCCA, MsetCCA, Bayesian	LDA	[246, 243, 245, 244]

4.3. Transfer learning

4.3.1. Principles

One of the major hypotheses in machine learning is that training data, on which the classifier is trained, and test data, on which the classifier is evaluated, belong to the same feature space and follow the same probability distribution. In many applications such as computer vision, biomedical engineering or brain-computer interfaces, this hypothesis is often violated. For BCI, a change in data distribution typically occurs when data are acquired from different subjects and across various time sessions.

Transfer learning aims at coping with data that violates this hypothesis by exploiting knowledge acquired while learning a given task for solving a different but related task. In other words, transfer learning is a set of methodologies considered for enhancing performance of a learned classifier trained on one task (also denoted as a domain) based on information gained while learning another task. Naturally, the effectiveness of transfer learning strongly depends on how well-related the two tasks are. For instance, it is more relevant to perform transfer learning between two P300 speller tasks performed by two different subjects than between one P300 speller task and a motor-imagery task performed by the same subject.

Transfer learning is of importance especially in situations where there exists abundant labelled data for one given task, denoted as a source domain, whilst data are scarce or expensive to acquire for the second task, denoted as a target domain. Indeed, in such cases, transferring knowledge from the source domain to the target domain acts as a bias or as a regularizer for solving the target task. We provide a more formal description of transfer learning based on the survey of Pan *et al.* [177]

More formally, a domain is defined by a feature space $\mathcal{X}$ and a marginal probability distribution $P(\mathbf{X})$ where the random variable $\mathbf{X}$ takes value $\mathcal{X}$. The feature space is associated with a label space $\mathcal{Y}$ and they are linked through a joint probability distribution $P(\mathbf{X}, \mathbf{Y})$ with $\mathbf{Y} = y \in \mathcal{Y}$. A task is defined by a label space $\mathcal{Y}$ and a predictive function $f(\cdot)$ which depends on the unknown probability distribution $P(\mathbf{X}, \mathbf{Y})$. For a given task, the objective is to learn the function $f(\cdot)$ based on pairs of examples $\{x_i, y_i\}_{i=1}^{\ell}$ where $x_i \in \mathcal{X}$ and $y_i \in \mathcal{Y}$.

Define the source and target domains as respectively $\mathcal{D}_S = \{\mathcal{X}_S, P_S(\mathbf{X})\}$ and $\mathcal{D}_T = \{\mathcal{X}_T, P_T(\mathbf{X})\}$ and the source and target tasks as $T_S = \{\mathcal{Y}_S, f_S(\cdot)\}$ $T_T = \{\mathcal{Y}_T, f_T(\cdot)\}$,

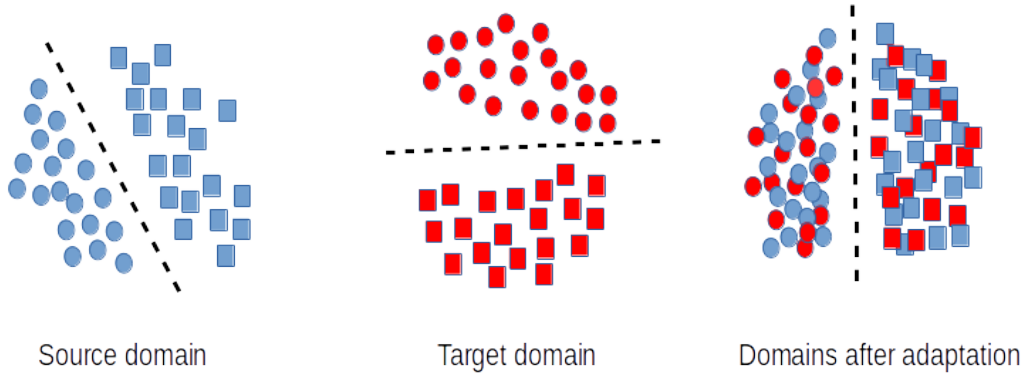

Figure 4. Illustrating the objective of domain adaptation. (left) source domain with labelled samples. (middle) target domain (with labels and decision function for the sake of clarity). A classifier trained on the source domain will perform poorly. (right) a domain adaptation technique will seek a common representation transformation or a mapping of domains so as to match the source and target domain distributions.

respectively. Hence, given the estimation of $f_T(\cdot)$ trained based solely on information from the target task, the goal of transfer learning is to improve on this estimation by exploiting knowledge obtained from $\mathcal{D}_S$ and $\mathcal{T}_S$ with $\mathcal{D}_S \neq \mathcal{D}_T$ or $\mathcal{T}_S \neq \mathcal{T}_T$. Note that $\mathcal{D}_S \neq \mathcal{D}_T$ occurs when either the feature spaces $\mathcal{X}_S$ and $\mathcal{X}_T$ are different or when the marginal distributions $P_S(\mathbf{X})$ and $P_T(\mathbf{X})$ are not equal. Similarly, $\mathcal{T}_S \neq \mathcal{T}_T$ indicates that either the label spaces are different or the predictive functions are different. For the latter situation, this reduces to situations where the two conditional probabilities differ: $P_S(y_S|\mathbf{X}_S) \neq P_T(y_T|\mathbf{X}_T)$.

Based on the learning setting and domains and tasks, there exist several situations applicable to transfer learning. For instance, homogeneous transfer learning refers to cases where $\mathcal{X}_S = \mathcal{X}_T$, and domain adaptation refers to situations where the marginal probability distributions or the conditional probability distributions do not match in the source and target domain. Settings in which labelled data are available in both source and target domains, and $\mathcal{T}_S \neq \mathcal{T}_T$, are referred to as inductive transfer learning. In BCI, this may be the case when the source domain and task are related to visual P300 evoked potentials whilst the target domain and task involve auditory P300-evoked potentials. In contrast, transductive transfer learning refers to situations in which tasks are similar but domains are different. A particular case is the domain adaptation problem when mismatch in domains is caused by mismatch in the marginal or conditional probability distributions. In BCI, transductive transfer learning is the most frequent situation, as inter-subject variability or session-to-session variability usually occurs. For more categorizations in transfer learning, we refer the reader to the survey of Pan *et al.* [177].

There exists a flurry of methods and implementations for solving a transfer learning problem, which depend on specific situations and the application of a domain. For homogeneous transfer learning, which is the most frequent situation encountered in brain-computer interfaces, there exist essentially three main strategies. If domain

distributions do not match, one possible strategy is to learn the transformation of source or target domain data so as to correct the distribution mismatch [203, 134]. If the type of mismatch occurs on the marginal distribution, then a possible method for compensating the change in distribution is to consider a reweighting scheme [208]. Many transfer learning approaches are also based on finding a common feature representation for the two (or more) domains. As the representation, or the retrieved latent space, is common to all domains, labelled samples from the source and target domain can be used to train a general classifier [53, 177]. A classic strategy is to consider approaches whose goal is to locate representations in which domains match. Another trend for transfer learning is to consider methods that learn a transformation of the data so that their distributions match. These transformations can either be linear, based for instance on kernel methods [241, 76] or non-linear, through the use of an optimal transport strategy [51].

Note that transfer learning may not always yield enhanced performance on a specific task $\mathcal{T}_T$. Theoretical results [55] in domain adaptation and transfer learning show that gain in performance on $\mathcal{T}_T$ may be achieved only if the source and target tasks are not too dissimilar. Hence, a careful analysis of how well tasks relate has to be carried out before considering transfer learning methods.

4.3.2. State-of-the-art

In recent years, transfer learning has gained much attention for improving Brain-Computer Interface classification. BCI research has focused on transductive transfer learning, in which tasks are identical between source and target. Motor Imagery has been the most-used paradigm to test transfer learning methods, probably owing to the availability of datasets from BCI Competitions [103, 67, 4, 143, 10, 102, 35, 97]. A few studies considered other paradigms such as the P300-speller [151, 74, 218], and Visual and Spatial attention paradigms [165]. A transfer learning challenge was also recently organized on an Error Potential dataset [1].

Instead of considering source and target domains one-to-one, a widespread strategy is to perform ensemble analyses, in which many pre-recorded sessions, from possibly different subjects, are jointly analysed. This addresses a well-known problem in data scarcity, especially involving labelled data, prone to overfitting.

There are many methods for combining the features and classifiers within ensembles [205]. A first concern when considering ensembles is to guarantee the quality of the features and classifiers from the source domain. Feature selection is also relevant in this context (see Section 2.2) to eliminate outliers. Many methods have been used to select relevant features from the ensemble, for instance Mutual Information [186], classification accuracy [143] or sparsity-inducing methods.

A second major challenge is to cope with the variability of data across subjects or sessions. Methods from adaptive classification are sometimes applicable in the context of transfer learning. Although the goal of adaptive classification, as explained

in section 4.1, is to update classifiers and not to transfer data, transfer learning can benefit from adaptive classification to update classifiers whose initialization is subject-independent. This approach has been proposed for P300 classification by Lu et al [151]. Riemannian geometry can also increase robustness with respect to inter-subject and inter-session variability, as demonstrated in several studies [46, 238].

A particularly fruitful strand of research has focused on building spatial filters based on ensemble data. Common Spatial Patterns (CSP) and spatial filters in general are able to learn quickly on appropriate training data, but do not perform well with a large quantity of heterogeneous data recorded from other subjects or other sessions [46]. A regularization strategy in this case is effective [103]. A more relevant approach is to directly regularize the CSP objective function rather than the covariance matrices [143]. In this vein, Blankertz *et al.* [21] have proposed an invariant CSP (iCSP), which regularizes the CSP objective function in a manner that diminishes the influence of noise and artefacts. Fazli *et al.* [67] built a subject-independent classifier for movement imagination detection. They first extracted an ensemble of features (spatial and frequency filters) and then applied LDA classifiers across all subjects. They compared various ways of combining these classifiers to classify a new subject's data: simply averaging their outcomes (bagging) performs adequately, but is outperformed by a sparse selection of relevant features.

Sparse representations are indeed relevant when applied to ensemble datasets coming from multiple sessions or subjects. The dictionary of waveforms / topographies / time-frequency representations, from which the sparse representations are derived, can be built in a manner to span a space that naturally handles the session- or subject-variability. Sparsity-inducing methods fall in the category of "invariant feature representation". Dictionaries can be predefined, but to better represent the data under study, they can be computed using data-driven methods. Dictionary Learning is a data-driven method which alternatively adapts the dictionary of representative functions and the coefficients of the data representation with the dictionary. Dictionary Learning has been used to reveal inter-trial variability in neurophysiological signals [91]. Morioka *et al.* [165] proposed to learn a dictionary of spatial filters which is then adapted to the target subject. This method has the benefit of taking into account the target subject's specificities, through their resting state EEG. Cho *et al.* [35] also exploit target session data by constructing spatiotemporal filters which minimally overlap with noise patterns, an extension of Blankertz's iCSP [21].

An even more sophisticated method to address the domain adaptation of features is to model their variability across sessions of subjects. Bayesian models capture variability through their model parameters. These models are generally implemented in a multitask learning context, where an ensemble of tasks $T_S = \{\mathcal{Y}_S, f_S(\cdot)\}$ is jointly learned from the source (labelled) domain. For BCIs, "a task" is typically a distinct recording session, either for a single or multiple subjects. Bayesian models have hence been built for features in spectral [4], spatial [102], and recently in combined spatial and spectral domains [97]. Combining a Bayesian model and learning from label proportion (LLP)

has recently been proposed in [218].

Another interesting domain adaptation method is to actually transport the features of the target data onto the source domain. Once transported to the source domain, the target data can then be classified with the existing classifier trained on the source data. Arvaneh et al [10] apply this approach to session-to-session transfer for Motor Imagery BCI, by estimating a linear transformation of the target data which minimizes the Kullback-Leibler distance between source and transformed target distributions. Recently, session-to-session transfer of P300 data has been accomplished using a nonlinear transform obtained by solving an Optimal Transport problem [74]. Optimal transport is well-suited for domain adaptation as its algorithms can be used for transporting probability distributions from one domain onto another [51].

4.3.3. Pros and cons

As reported in the above cited studies, transfer learning is instrumental in session-to-session and subject-to-subject decoding performance. This is essential to be able to achieve a true calibration-free BCI mode of operation in the future, which in turn would improve BCI usability and acceptance. In fact, it is well recognized in the community that the calibration session may be unduly tiring for clinical users, whose cognitive resources are limited, and annoying in general for healthy users. As discussed by Sanelli *et al.* [195], receiving feedback from the very beginning of their BCI experience is highly motivating and engaging for novice users. Transfer learning can then provide users with an adequately-performing BCI, before applying co-adaptive strategies. In this spirit, transfer learning may be used to initialize a BCI using data from other subjects for a naive user and data from other sessions for a known user. In any case such an initialization is suboptimal, thus such an approach entails adapting the classifier during the session, a topic that we have discussed in Section 4.1. Therefore, transfer learning and adaptivity must come arm in arm to achieve the final goal of a calibration-free mode of operation [46].

Although suboptimal in general, transfer learning is robust by definition. For instance, subject-to-subject transfer learning can produce better results as compared to subject-specific calibration if the latter is of low quality [15]. This is particularly useful in clinical settings, where obtaining a good calibration is sometimes prohibitive [158].

As we have seen, the approach of seeking invariant spaces for performing classification in transfer learning settings is appealing theoretically and has shown promising results by exploiting Riemannian geometry; however it comes at the risk of throwing away some of the information that is relevant for decoding. In fact, instead of coping with the variability of data across sessions, as we formulated above, it may be wiser to strive to benefit from the variability in the ensemble to better classify the target session. The idea would be to design classifiers able to represent multiple sessions or subjects.

The combination of transfer learning and adaptive classifiers represents a topic at the forefront of current research in BCI. It is expected to receive increasing attention in the upcoming years, leading to a much-sought new generation of calibration-free brain-computer interfaces.

Very few of the transfer learning presented methods have yet been used online, but computational power is not a limitation, because these methods do not require extensive computational ressources, and can be run on simple desktop computers. For methods whose learning phases may take a long time (such as sparsity-inducing methods, or dictionary learning), this learning should be performed in advance so that the adaptation to a new subject or session is time-efficient [165].

Table 5. Summary of transfer learning methods for BCI

EEG Pattern	**Features / Method**	**Classifier / Transfer**	**References**
Motor Imagery	CSP + band power	linear SVM subject-to-subject	[103, 143]
Motor Imagery	sparse feature set	LDA	[67]
Motor Imagery	CSP	Fisher LDA session-to-session	[35]
Motor Imagery	Surface Laplacian	LDA, Bayesian multitask subject-to-subject	[4, 97]
Motor Imagery	PCSP	LDA, Bayesian model multisubject	[102]
Motor Imagery	CSP + band power	LDA session-to-session	[10]
Visual, Spatial attention	Dictionary learning of spatial filters	linear SVM subject-to-subject	[165]
P300	time points	mixture of bayesian classifiers	[218]
P300	time points	Fisher LDA multisubject	[151]
P300	xDAWN	LDA, optimal transport session-to-session	[74]

4.4. Deep Learning

Deep learning is a specific machine learning algorithm in which features and the classifier are jointly learned directly from data. The term deep learning is coined by the architecture of the model, which is based on a cascade of trainable feature extractor modules and nonlinearities. Owing to such a cascade, learned features are usually related to increasing levels of concepts. We discuss in this section the two most popular deep learning approaches for BCI: convolutional neural networks and restricted Boltzmann

machines.

4.4.1. Principles

A short introduction on Restricted Boltzmann machines:

A restricted Boltzmann machine (RBM) is a Markov Random Field (MRF) [120] associated with a bipartite undirected graph. It is composed of two sets of units: m visible ones $V = (V_1, \cdots, V_m)$ and n hidden ones $H = (H_1, \cdots, H_n)$. The visible units are used for representing observable data whereas the hidden ones capture some dependencies between observed variables. For the usual type of RBM such as those discussed in this paper, units are considered as random variables that take binary values $(\mathbf{v}, \mathbf{h})$ and $\mathbf{W}$ is a matrix whose entries $w_{i,j}$ are the weights associated with the connection between unit v_i and h_j. The joint probability of a given configuration $(\mathbf{v}, \mathbf{h})$ can be modelled according to the probability $p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} e^{-E(\mathbf{v},\mathbf{h})}$ with the energy function $E(\mathbf{v}, \mathbf{h})$ being

$$E(\mathbf{v}, \mathbf{h}) = \mathbf{a}^\top \mathbf{v} - b^\top \mathbf{h} - \mathbf{v}^\top \mathbf{W} \mathbf{h}$$

where $\mathbf{a}$ and $\mathbf{b}$ are bias weight vectors. Note Z is a normalizing factor in order that $p(\mathbf{v}, \mathbf{h})$ sums to one for all possible configurations. Owing to the undirected bipartite graph property, hidden (respective to the visible) variables are independent given the visible (hidden) ones leading to:

$$p(\mathbf{v}|\mathbf{h}) = \Pi_{i=1}^{m} p(v_i|\mathbf{h}) \qquad p(\mathbf{h}|\mathbf{v}) = \Pi_{j=1}^{n} p(h_j|\mathbf{v})$$

and marginal distributions over the visible variables can be easily obtained as [69]:

$$p(\mathbf{v}) = \frac{1}{Z} \sum_{h} e^{-E(\mathbf{v},\mathbf{h})}$$

Hence, by optimizing all model parameters $(\mathbf{W}, \mathbf{b}, \mathbf{a})$, it is possible to model the probability distribution of the observable variables. Other properties of RBMS as well as connections of RBMs with stochastic neural networks are detailed in [90, 69]

To learn the probability distribution of the input data, RBMs are usually trained according to a procedure denoted as contrastive divergence learning [89]. This learning procedure is based on a gradient ascent of the log-likelihood of the training data. The derivative of the log-likelihood of an input v can be easily derived [69] and the mean of this derivative over the training set leads to the rule:

$$\sum_{v} \frac{\partial L(\mathbf{W}|\mathbf{v})}{\partial w_{i,j}} \propto \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model}$$

with the two brackets respectively denoting expectation over $p(\mathbf{h}|\mathbf{v})q(\mathbf{v})$ and over the model $(p(\mathbf{v}, \mathbf{h}))$ with q being the empirical distribution of the inputs. While the first term of this gradient is tractable, the second one has exponential complexity. Contrastive

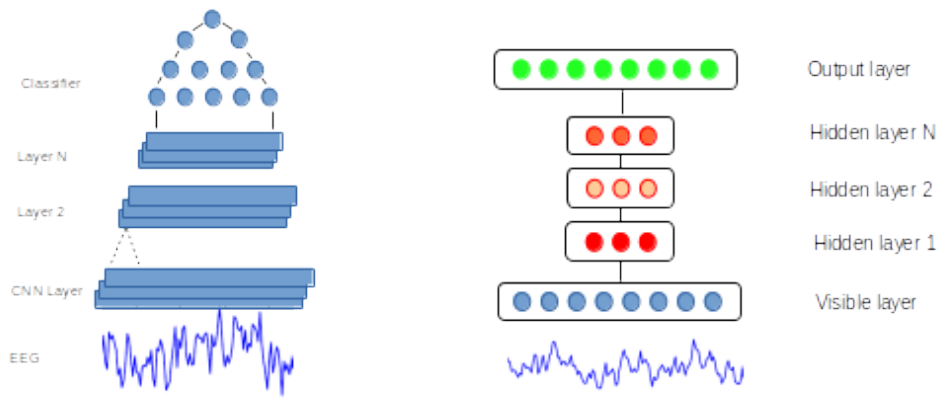

Figure 5. Example architectures of two deep learning frameworks. (left) convolutional neural networks. The blue blocks refer to results of convolving input signal with several different filters. (right) stacked restricted Boltzmann machines. Hidden layers are trained layer-wise and the full network can be fine-tuned according to the task at hand.

divergence aims at approximating this gradient using a Gibbs chain procedure that computes the binary state of $\mathbf{h}$ using $p(\mathbf{h}|\mathbf{v})$ and then obtaining an estimation of $\mathbf{v}$ using $p(\mathbf{v}|\mathbf{h})$ [89]. There exist other methods for approximating the gradient of RBMs log-likelihood that may lead to better solutions as well as methods for learning with continuous variables [213, 17].

The above procedure allows one to learn a generative model of the inputs using a simple layer of RBMs. A deep learning strategy can be obtained by stacking several RBMs with the hidden units of one layer used as inputs of the subsequent layers. Each layer is usually trained in a greedy fashion [90] and fine-tuning can be performed depending on the final objective of the model.

Short introduction on convolutional neural networks:

A Convolutional Neural Network (ConvNet or CNN) is a feedforward neural network (a network in which information flows uni-directionally from the input to the hidden layers to the output) which has at least one convolutional layer [71, 117, 117]. Such a convolutional layer maps its input to an output through a convolution operator. Suppose that the input is a 1D signal $\{x_n\}$ with N samples, its convolution through a 1D filter $\{h_m\}$ of size M is given by:

$$y(n) = \sum_{i=0}^{M-1} h_i x_{n-i} \quad \forall n = 0, \cdots, N-1$$

This equation can be extended to higher dimensions by augmenting the number of summations in accordance with the dimensions. Several filters can also be independently used in convolution operations leading to an increased number of channels in the output. This convolutional layer is usually followed by nonlinearities [75] and possibly by a pooling layer that aggregate the local information of the output into a single value, typically through an average or a max operator [25]. Standard ConvNet architectures usually stack several of these layers (convolution + non-linearity (+ pooling)) followed by other layers, typically fully connected, that act as a classification layer. Note however that some architectures use all convolutional layers as classification layers. Given some architectures, the parameters of the models are the weights of all the filters used for convolution and the weights of the fully connected layers.

ConvNets are usually trained in a supervised fashion by solving an empirical risk minimization problem of the form:

$$\hat{\mathbf{w}} = \arg\min_{w} \frac{1}{\ell} \sum_{i} L(y_i, f_{\mathbf{w}}(\mathbf{x}_i)) + \Omega(\mathbf{w})$$

where $\{\mathbf{x}_i, y_i\}_{i=1}^{\ell}$ are the training data, $f_{\mathbf{w}}$ is the prediction function related to the ConvNet, $L(\cdot,\cdot)$ is a loss function that measures any discrepancy between the true labels of $\mathbf{x}_i$ and $f_{\mathbf{w}}(\mathbf{x})_i$, and Ω is a regularization function for the parameters of the ConvNet. Owing to the specific form of the global loss (average loss over the individual samples), stochastic gradient descent and its variants are the most popular means for optimizing deep ConvNets. Furthermore, the feedforward architecture of $f_{\mathbf{w}}(\cdot)$ allows the computation of the gradient at any given layer using the chain rule. This can be performed efficiently using the back-propagation algorithm [193].

In several domain applications, ConvNets have been very successful because they are able to learn the most relevant features for the task at hand. However, their performances strongly depend on their architectures and their learning hyper-parameters.

4.4.2. State-of-the-art

Deep Neural Networks (DNNs) have been explored for all major types of EEG-based BCI systems; that is P300, SSVEP, Motor Imagery and passive BCI (for emotions and workload detection). They have also been studied for less commonly used EEG patterns such as Slow Cortical Potentials (SCP) or Motion-onset Visual Evoked Potential (MVEP). It is worth mentioning that all these studies were performed offline.

Regarding P300-based BCI, Cecotti *et al.* published the very first paper which explored CNN for BCI [32]. Their network comprised two convolutional layers, one to learn spatial filters and the other to learn temporal filters, followed by a fully connected layer. They also explored ensembles of such CNNs. This network outperformed the BCI competition winners on the P300-speller data set used for evaluation. However, an ensemble of SVMs obtained slightly better performances than

the CNN approach. Remaining with P300 classification, but this time in the context of the Rapid Serial Visualization Paradigm (RSVP), [156] explored another CNN with one spatial convolution layer, two temporal convolution layers and two dense fully-connected layers. They also used Rectifying Linear Units, Dropout and spatio-temporal regularization on the convolution layers. This network was reported as more accurate than a spatially Weighted LDA-PCA classifier, by 2%. It was not compared to any other classifier though. It should be mentioned that in this paper, as in most BCI papers on Deep Learning, the architecture is not justified and not compared to different architectures, apart from the fact that the architecture was reported to perform well.

For SSVEP, [113] also explored a CNN with a spatial convolutional layer and a temporal one that used band power features from two EEG channels. This CNN obtained performance similar to that of a 3-layer MLP or that of a classifier based on Canonical Correlation Analysis (CCA) with kNN data recorded from static users. However, it outperformed both on noisy EEG data recorded from a moving user. However, the classifiers that were compared to the CNN were not the state-of-the-art for SSVEP classification (e.g. CCA was not used with any harmonics of the SSVEP stimulus known to improve performance nor with more channels).

For SCP classification, [59] explored a Deep Extreme Learning Machine (DELM), which is a multilayer ELM with the last layer being a Kernel ELM. The structure of the network, its number of units, the input features and hyper-parameters were not justified. Such network obtained lower performance than the BCI competition winners for the data set used, and was not significantly better than a standard ELM or multilayer ELM.

For MVEP, [153] used a Deep Belief Network (DBN) composed of three RBMs. The dimensionality of the input features, EEG time points, was reduced using compressed sensing (CS). This DBN+CS approach outperformed a SVM approach which used neither DBN nor CS.

Regarding passive BCIs, Yin *et al.* explored DNNs for both workload and emotions classifications [235, 234]. In [234], they used adaptive DBN, composed of several stacked Auto-Encoders (AE), for workload classification. Adaptation was performed by retraining the first layer of the network using incoming data labelled with their estimated class. Compared to kNN, MLP or SVM, the proposed network outperformed all without channel selection, but obtained similar performance with feature selection. As is too often the case in DNN papers for BCI, the proposed approach was not compared to the state-of-the-art, e.g. to methods based on FBCSP. In [235], another DBN composed of stacked AE was studied. This DNN was however a multimodal one, with separate AEs for EEG signals and other physiological signals. Additional layers merged the two feature types. This approach appeared to outperform competing classifiers and published results using the same database. However, the data used to perform model selection of the proposed DNN and determine its structure was all data, that is, it included the test data, which biased the results.

Several studies have explored DNN for motor imagery classification with both

DBN and CNN [150, 198, 207, 210]. A DBN was explored in [150] to classify BP features from two EEG channels. The network outperformed FBCSP and the BCI competition winner but only when using an arbitrary structure whose selection was not justified. When removing or adding a single neuron, this network exhibited lower performance than FBCSP or the competition winner, hence casting doubts on its reliability and its initial structure choice. Another DBN was used in [207] for motor imagery classification, but was outperformed by a simple CSP+LDA classifier. However, the authors proposed a method to interpret what the network has learned and its decisions, which provided useful insights on the possible neurophysiological causes of misclassifications. A combination of CNN and DBN was explored in [210]. They used a CNN whose output was used as input to a 6-layer SAE. Compared to only a CNN, a DBN or a SVM, the CNN+DBN approach appeared to be the most effective. It was not compared to the BCI competition winners on this data set, or to other state-of-the-art methods such as Riemannian geometry and FBCSP. The last study to explore DNN for motor imagery is that of Schirrmeister *et al.* [198]. This study should be particularly commended as, contrary to most previously mentioned papers, various DNN structures are explored and presented, all carefully justified and not arbitrary, and the networks are rigorously compared to state-of-the-art methods. They explored Shallow CNN (one temporal convolution, one spatial convolution, squaring and mean pooling, a softmax layer), Deep CNN (temporal convolution, spatial convolution, then three layers of standard convolution and a softmax layer), an hybrid Shallow+Deep CNN (i.e., their concatenation), and Residual NN (temporal convolution, spatial convolution, 34 residual layers, and softmax layer). Both the Deep and Shallow CNN significantly outperformed FBCSP, whereas the Hybrid CNN and the residual NN did not. The shallow CNN was the most effective with +3.3% of classification accuracy over FBCSP. The authors also proposed methods to interpret what the network has learned, which can provide useful neurophysiological insights.

Finally, a study explored a generic CNN, a compact one with few layers and parameters, for the classification of multiple EEG patterns, namely P300, Movement Related Cortical Potentials (MRCP), ErrP and Motor Imagery. This network outperformed another CNN (that of [156] mentioned above), and XDAWN+BDA as well as RCSP+LDA for subject-to-subject classification. The parameters (number of filters and the pass-band used) for xDAWN and RCSP are not specified though but would be suboptimal if they used the same band as for the CNN. The method is also not compared to the state-of-the-art (FBCSP or Riemannian) methods. Comparison to existing methods is thus again unconvincing.

A summary of the methods using Deep Learning for EEG classification in BCI are listed in Table 6.

4.4.3. Pros and cons

DNNs have the potential to learn both effective features and classifiers

simultaneously from raw EEG data. Given their effectiveness in other fields, DNNs certainly seem promising to lead to better features and classifiers, and thus to much more robust EEG classification. However, so far, the vast majority of published studies on DNNs for EEG-based BCIs have been rather unconvincing in demonstrating their actual relevance and superiority to state-of-the-art BCI methods in practice. Indeed, many studies did not compare the studied DNN to state-of-the-art BCI methods or performed biased comparisons, with either suboptimal parameters for the state-of-the-art competitors or with unjustified choices of parameters for the DNN, which prevents us from ruling out manual tuning of these parameters with knowledge of the test set. There is thus a need to ensure such issues be solved in future publications around DNN for BCI. An interesting exception is the work in [198], who rigorously and convincingly showed that a shallow CNN could outperform FBCSP. This suggests that the major limitation of DNN for EEG-based BCI is that such networks have a very large number of parameters, which thus requires a very large number of training examples to calibrate them. Unfortunately, typical BCI data sets and experiments have very small numbers of training examples, as BCI users cannot be asked to perform millions or even thousands of mental commands before actually using the BCI. As a matter of fact, it has been demonstrated outside the BCI field that DNNs are actually suboptimal and among the worst classifiers with relatively small training sets [36]. Unfortunately, only small training sets are typically available to design BCIs. This may explain why shallow networks, which have much fewer parameters, are the only ones which have proved useful for BCI. In the future, it is thus necessary to either design NNs with few parameters or to obtain BCI applications with very large training data bases, e.g., for multi-subject classification.

It is also worth noting that DNNs so far were only explored offline for BCI. This is owing to their very long training times. Indeed, the computational complexity of DNN is generally very high, both for training and testing. Calibration can take hours or days on standard current computers, and testing, depending on the number of layers and neurons, can also be very demanding. As a result, high-performing computing tools, e.g., multiple powerful graphic cards, may be needed to use them in practice. For practical online BCI applications, the classifier has to be trained in at most a few minutes to enable practical use (BCI users cannot wait for half an hour or more every time they want to use the BCI). Fast training of a DNN would thus be required for BCI. Designing DNNs that do not require any subject-specific training, i.e., a universal DNN, would be another alternative.

4.5. Other new classifiers

4.5.1. Multilabel classifiers

Principles:

Table 6. Summary of works using Deep Learning for EEG-based BCI

EEG Pattern	Features	Classifier	References
SCP	not specified	Deep ELM	[59]
Motion-onset VEP	EEG time points	DBN	[153]
SSVEP	Band Power	CNN	[113]
P300	EEG time points	CNN	[32]
P300	EEG time points	CNN	[156]
Motor Imagery	Band Power	DBN	[150]
Motor Imagery/Execution	raw EEG	CNN	[198]
Motor Imagery	Band power	DBN	[207]
Motor Imagery	Band power	CNN+DBN	[210]
Workload	Band power	adaptive DBN	[234]
Emotions	Band power + zero crossing + entropy	DBN	[235]
ErrP, P300, MRCP, Motor Imagery	EEG time points	CNN	[116]

In order to classify more than two mental tasks, two main approaches can be used to obtain a multiclass classification function [215]. The first approach consists in directly estimating the class using multiclass techniques such as decision trees, multilayer perceptrons, naive Bayes classifiers or k-nearest neighbours. The second approach consists of decomposing the problem into several binary classification problems [5]. This decomposition can be accomplished in different ways using i) one-against-one pairwise classifiers [20, 84], ii) one-against-the-rest (or one-against-all) classifiers [20, 84], iii) hierarchical classifiers similar to a binary decision tree and iv) multi-label classifiers [215, 154]. In the latter case, a distinct subset of L labels (or properties) is associated to each class [58]. The predicted class is identified according to the closest distance between the predicted labels and each subset of labels defining a class.

State-of-the-art:

The number of commands provided by motor imagery-based BCIs depends on the number of mental imagery states that the system is able to detect. This, in turn, is limited by the number of body parts that users can imagine moving in a manner that generate clear and distinct EEG patterns. Multi-label approaches can thus prove useful for detecting combined motor imagery tasks, i.e. imagination of two or more body parts at the same time [226, 192, 125], with each body part corresponding to a single label (indicating whether that body part was used). Indeed, in comparison with the standard approach, this approach has the advantage of considerably increasing the number of different mental states while using the same number of body parts: 2^P

compared to P, where P is the number of body parts. Thus, EEG patterns during simple and combined motor imagery tasks were investigated to confirm the separability of seven different classes of motor imagery for BCI [249, 226, 126]. For the purpose of achieving continuous 3D control, both hands motor imagery was adopted to complement the set of instructions in a simple limb motor imagery based-BCI to go up (and rest to go down) [192, 114]. The up/down control signal was the inverted addition of left and right autoregressive spectral amplitudes calculated for each of the electrodes and 3Hz-frequency bins. Another method converted circular ordinal regression to a multi-label classification approach to control a simulated wheelchair, using data set IIIa of the third BCI competition, with as motor tasks imagination of left hand, right hand, foot and tongue movements [57]. Multiclass and multi-label approaches have been compared to discriminate height commands from the combination of three motor imagery tasks (left hand, right hand and feet) to control a robotic arm [125]. A first method used a single classifier applied to the concatenated features related to each activity source (C3, Cz, C4), with one source for each limb involved. A second approach consisted of a hierarchical tree of three binary classifiers to infer the final decision. The third approach was a combination of the first two approaches. All methods used the CSP algorithm for feature extraction and Linear Discriminant Analysis (LDA) for classification. All methods were validated and compared to the classical One-Versus-One (OVO) and One-versus-Rest (OVR) methods. Results obtained with the hierarchical method were similar to the ones obtained with the OVO and OVR approaches. The performance obtained with the first approach (single classifier) and the last (combined hierarchical classifier) were the best for all subjects. The various multi-label approaches explored are mentioned in Table 7.

Pros and cons:

Multiclass and multi-label approaches therefore aim to recognize more than two commands. In both cases, the resulting increase in the number of recognized classes potentially provides the user with a greater number of commands to interact more quickly with the system, without the need for a drop-down menu, for example. The multi-label approach can make learning shorter and less tiring, as it requires learning only a small number of labels. The many possible combinations of these labels leads to a large number of classes and therefore to more commands. In addition, the multi-label approach allows redundancy in the labels describing a class, which can lead to better class separation. Usually, the number of labels to produce is less than the number of classes. Finally, as compared to standard methods, multiclass and multilabel approaches usually have a lower computational complexity since they can share parameters, e.g., using multilayer perceptron, or class descriptors (especially if no redundancy is introduced).

However, there might be a lack of relationship between the meaning of a label and the corresponding mental command, e.g., two hand imagery to go up. This may

generate a greater mental workload and therefore fatigue. It is therefore necessary to choose carefully the mapping between mental commands and corresponding labels. Finally, classification errors remain of course possible. In particular, the set of estimated labels may sometimes not correspond to any class, and several classes may be at equal distances, thus causing class confusion.

Table 7. Summary of multi-label (and related multiclass) approaches for EEG-based BCI

EEG Pattern	**Features**	**Classifier**	**References**
Motor Imagery (8 classes)	band power	LDA	[127]
Motor Imagery (8 classes)	band power	Riemannian Geometry	[128]
Mental tasks (4 classes)	band power	cSVM	[57]
Motor imagery (4 classes)	band power	(ratio between band powers)	[192, 114]
Motor imagery (7 classes)	band power	SVM	[226]
Motor imagery (4 classes)	band power	(mapping to velocity)	[162]

4.5.2. Classifiers that can be trained from little data

Principles:

As previously discussed, most EEG-based BCIs are currently optimized for each subject. Indeed, this has been shown to lead in general to substantially higher classification performances than subject-independent classifiers. Typical BCI systems can be optimized by using only a few training data, typically 20 to 100 trials per class, as subjects cannot be asked to produce the same mental commands thousands of times before being provided with a functional BCI. Moreover, collecting such training data takes time, which is inconvenient for the subjects, and an ideal BCI would thus require a calibration time as short as possible. This calls for classifiers that can be calibrated using as little training data as possible. In the following we present those classifiers that were shown to be effective for this purpose. They rely on using statistical estimators dedicated to small sample size or on dividing the input features between several classifiers to reduce the dimensionality, thus reducing the amount of training data needed by each classifier.

State-of-the-art:

The three main classifiers that have been shown to be effective with little training data, and thus effective for EEG-based BCI design, are the shrinkage LDA classifier [22, 142, 137], random forest [3, 54] and Riemannian Classifiers [47, 232].

The shrinkage LDA (sLDA), is a standard LDA classifier in which the class-related covariance matrices used in its optimization were regularized using shrinkage

[22]. Indeed, covariance matrices estimated from little data tend to have larger extreme eigenvalues than the real data distribution, leading to poor covariance estimates. This can be solved by shrinking covariance matrices $\mathbf{\Sigma}$ as $\hat{\mathbf{\Sigma}} = \mathbf{\Sigma} - \lambda\mathbf{I}$, with $\mathbf{I}$ the identity matrix, and λ the regularization parameter. Interestingly enough, there are analytical solutions to automatically determine the best λ value (see [118]. The resulting sLDA classifier has been shown to be superior to the standard LDA classifier for BCI, both for ERP-based BCI [22] and for oscillatory activity BCI [137]. It has also been shown that such classifier can be calibrated with much fewer data than LDA to achieve the same performance [142, 137]. For instance, for mental imagery BCI, an sLDA has been shown to obtain similar performance with 10 training trials per class than a standard LDA with 30 training trials per class, effectively reducing the calibration time three-fold [137].

Random Forest (RF) classifiers are ensembles of several decision tree classifiers [26]. The idea behind this classifier is to randomly select a subset of the available features, and to train a decision tree classifier on them, then to repeat the process with many random feature subsets to generate many decision trees, hence the name random forest. The final decision is taken by combining the outputs of all decision trees. Because each tree only uses a subset of the features, it is less sensitive to the curse-of-dimensionality, and thus requires fewer training data to be effective. Outside of BCI research, among various classifiers and across various classification problems and domains, random forest algorithms were actually often found to be among the most accurate classifiers, including problems with small training data sets [26, 36]. RFs were used successfully even online both for ERP-based BCI [3] and for motor imagery BCI [54]. They outperformed designs based on LDA classifiers for motor imagery BCI [54].

Riemannian Classifiers have been discussed in section 4.2.1. Typically, a simple Riemannian classifier such as the RMDM requires less training data as compared to optimal filtering approaches such as the CSP for motor imagery [46] and xDAWN for P300 [15]. This is due to the robustness of the Riemannian distance, which the geometric mean inherits directly, as discussed in [47]. Even more robust mean estimations can be obtained computing Riemannian medians or trimmed Riemannian means. Shrinkage and other regularization strategies can also be applied to a Riemannian framework to improve the estimation of covariance matrices when a small number of data points is considered [100].

Pros and cons:

sLDA, RF and the RMDM are simple classifiers that are easy to use in practice and provide good results in general, including online. We thus recommend their use. sLDA and RMDM do not have any hyper-parameters, which makes them very convenient to use. sLDA have been shown to be superior to LDA both for ERP and oscillatory activity-based BCI across a number of data sets [22, 137]. There is thus no reason to use classic LDA; instead sLDA should be preferred. RMDM performs as well as CSP+LDA

for oscillatory activity-based BCI [13, 46], as well as xDAWN+LDA but better than a step-wise LDA on time samples for ERP-based BCI [15, 46] and better than CCA for SSVEP [100]. Note that because LDA is a linear classifier, it may be suboptimal in the hypothetic future case when vast amounts of training data will be available. RF on the other hand is a non-linear classifier that can be effective both with small and large training sets [36]. RMDM is also non-linear and performs well with small as well as large training sets [46]. In terms of computational complexity, while RF can be more demanding than RMDM or sLDA since it uses many classifiers, all of them are fairly simple and fast methods, and all have been used online successfully on standard computers.

Table 8. Summary of classifiers that can be trained with limited amount of data

EEG Pattern	Features	Classifier	References
P300	Time points	sLDA	[142]
P300	Time points	sLDA	[22]
P300	Time points	RF	[3]
P300	Special Covariance	RMDM	[46]
P300	Special Covariance	RMDM	[15]
Motor Imagery	CSP + band power	RF	[54]
Motor Imagery	CSP + band power	sLDA	[137]
Motor Imagery	Band-Pass Covariance	RMDM	[46, 14]
SSVEP	Band-Pass Covariance	RMDM	[100]

5. Discussion and guidelines

Based on the many papers surveyed in this manuscript, we identify some guidelines on whether to use various types of classification methods, and if so, when and how it seems relevant to do so. We also identify a number of open research questions that deserve to be answered in order to design better classification methods to make BCI more reliable and usable. These guidelines and open research questions are presented in the two following sections.

5.1. Summary and guidelines

According to the various studies surveyed in this paper, we extract the following guidelines for choosing appropriate classification methods for BCI design:

- In terms of classification performance, adaptive classification approaches, both for classifiers and spatial filters, should be preferred to static ones. This should be the case even if only unsupervised adaptation is possible for the targeted application.

- Deep learning networks do not appear to be effective to date for EEG signals classification in BCI, given the limited training data available. Shallow convolutional neural networks are more promising.
- Shrinkage Linear Discriminant Analysis (sLDA) should always be used instead of classic LDA, as it is more effective and more robust for limited training data.
- When very little training data is available, transfer learning, sLDA, Riemannian Minimum Distance to the Mean (RMDM) classifiers or Random Forest should be used.
- When tasks are similar between subjects, domain adaptation can be considered for enhancing classifier performance. However, care should be taken regarding the effectiveness of the transfer learning, as it may sometimes decrease performance.
- Riemannian Geometry Classifiers (RGC) are very promising, and are considered the current state-of-the-art for multiple BCI problems, notably Motor Imagery, P300 and SSVEP classification. They should be further applied and further explored to increase their effectiveness.
- Tensor approaches are emerging and as such may also be promising but currently require more research to be applicable in practice, online, and to assess their performance as compared to other state-of-the-art methods.

5.2. Open research questions and challenges

In addition to guidelines, our survey also enabled us to identify a number of unresolved challenges or research questions and points that must be addressed. These challenges and questions are presented below.

- Many of the classification methods surveyed in this paper have been evaluated offline only. However, an actual BCI application is fundamentally online. There is thus a need to study and validate these classification methods online as well, to ensure they are sufficiently computationally efficient to be used in real time, can be calibrated quickly enough to be convenient to use and to ensure that they can withstand real-life noise in EEG signals. In fact, online evaluation of classifiers should be the norm rather than the exception, as there is relatively little value in studying classifiers for BCI if they cannot be used online.
- Transfer learning and domain adaptation may be key components for calibration-free BCI. However, at this stage, several efforts must be taken before they can be routinely used. Among the efforts, coupling advanced features such as covariance matrices and domain adaptation algorithms can further improve on the invariance ability of BCI systems.
- There are also several open challenges that, once solved, could make Riemannian geometry classifiers even more efficient. One would be to design a stable estimator of the Riemannian median to make RMDM classifiers more robust to outliers than when using the Riemannian mean. Another would be to work on multimodal

RMDM, with multiple modes per class, not just one, which could potentially improve their effectiveness. Finally, there is a need for methods to avoid poorly conditioned covariance matrices or low rank matrices, as these could cause RGC to fail.

- While deep learning approaches are lagging in performance for BCI, mostly due to lack of large training datasets, they can be strongly relevant for end-to-end domain adaptation [73] or for augmenting datasets through the use of generative adversarial networks [77].
- Classifiers, and the entire machine learning/signal processing pipeline are not the only considerations in a BCI system design. In particular, the user should be considered as well and catered to so as to ensure efficient brain-computer communications [144, 112, 33]. As such, future BCI classifiers should be designed to ensure that users can make sense of the feedback from the classifier, and can learn effective BCI control from it [146].

6. Conclusion

In this manuscript, we have surveyed the EEG classification approaches that have been developed and evaluated between 2007 and 2017 in order to design BCI systems. The numerous approaches that were explored can be divided into four main categories: adaptive classifiers, matrix and tensor classifiers, transfer learning methods, and deep learning. In addition, a few miscellaneous methods were identified outside these categories, notably the promising shrinkage LDA and Random Forest classifiers.

Overall, our review revealed that adaptive classifiers, both supervised and unsupervised, outperform static ones in general. Matrix and tensor classifiers are also very promising to improve BCI reliability; in particular, Riemannian geometry classifiers are the current state-of-the-art for many BCI designs. Transfer learning seems useful as well, particularly when little training data is available, but its performance is highly variable. More research should be invested to evaluate it as part of standard BCI design. Shrinkage LDA and Random Forest are also worthy tools for BCI, particularly for small training data sets. Finally, contrary to their success in other fields, deep learning methods have not demonstrated convincing and consistent improvements over state-of-the-art BCI methods to date.

Future work related to EEG-based BCI classification should focus on developing more robust and consistently efficient algorithms that can be used easily and online and are able to work with small training samples, noisy signals, high-dimensional and non-stationary data. This could be addressed by further developing transfer learning methods, Riemannian geometry and tensor classifiers, and by identifying where and how deep networks could be useful for BCI. Altogether, improving those methods or defining new ones should consider invariance. Indeed, an ideal classification method would use features and/or classifiers that are invariant over time, over users and contexts, to be effective in any situation. Additionally, there is also need for a new generation of BCI

classification methods that consider the human user in the loop, i.e., that can adapt to user states, traits and skills, and provide feedback that the user can make sense of and learn from.

To conclude, this review suggests that it is time to change the gold standard classification methods used in EEG-based BCI so far, and apply a second generation of BCI classifiers. We could for instance move from the classical CSP+LDA design, mostly unchanged for years in many online studies, to adaptive Riemannian geometry classifiers. Finally, and even more importantly, the next generation of classification approaches for EEG-based BCI will have to take the user into account.

Acknowledgments

F. Lotte received support from the French National Research Agency with the REBEL project and grant ANR-15-CE23-0013-01, the European Research Council with the BrainConquest project (grant ERC-2016-STG-714567) and the Japan Society for the Promotion of Science (JSPS). A. Cichocki was supported by the Ministry of Education and Science of the Russian Federation (grant 14.756.31.0001) and the Polish National Science Center (grant 2016/20/W/N24/00354). The authors would also like to acknowledge the support of the BCI-LIFT Inria Project Lab.

References

[1] BCI challenge @ NER2015. https://www.kaggle.com/c/inria-bci-challenge, 2015. Online transfer learning competition, at the IEEE EMBS Neural Engineering conference, 2015.

[2] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2009.

[3] F. Akram, S. M. Han, and T.-S. Kim. An efficient word typing P300-BCI system using a modified T9 interface and random forest classifier. *Computers in Biology and Medicine*, 56:30–36, 2015.

[4] M. Alamgir, M. Grosse-Wentrup, and Y. Altun. Multitask learning for brain-computer interfaces. In *International Conference on Artificial Intelligence and Statistics*, pages 17–24, 2010.

[5] E. L. Allwein, R. E. Schapire, and Y. Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. *J. Mach. Learn. Res.*, 1:113–141, Sept. 2001.

[6] A. Andreev, A. Barachant, F. Lotte, and M. Congedo. Recreational applications in OpenViBE: Brain invaders and use-the-force. In *In "Brain-Computer Interfaces 2: Technology and Application", M. Clerc, L. Bougrain, F. Lotte (Eds), Wiley-iSTE*, pages 241–258, 2016.

[7] K. Ang, Z. Chin, C. Wang, C. Guan, and H. Zhang. Filter bank common spatial pattern algorithm on BCI competition IV datasets 2a and 2b. *Frontiers in Neuroscience*, 6, 2012.

[8] K. K. Ang and C. Guan. Brain-computer interface in stroke rehabilitation. *Journal of Computing Science and Engineering*, 7(2):139–146, 2013.

[9] X. Artusi, I. K. Niazi, M.-F. Lucas, and D. Farina. Performance of a simulated adaptive BCI based on experimental classification of movement-related and error potentials. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 1(4):480–488, 2011.

[10] M. Arvaneh, C. Guan, K. K. Ang, and C. Quek. EEG data space adaptation to reduce intersession nonstationarity in brain-computer interface. *Neural computation*, 25(8), 2014.

[11] T. Balli and R. Palaniappan. Classification of biological signals using linear and nonlinear features. *Physiological Measurement*, 31(7):903, 2010.

[12] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten. Common spatial pattern revisited by Riemannian geometry. In *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pages 472–476. IEEE, 2010.

[13] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten. Multi-class brain computer interface classification by riemannian geometry. *IEEE Transactions on Biomedical Engineering*, 59(4):920–928, 2012.

[14] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten. Classification of covariance matrices using a Riemannian-based kernel for BCI applications. *Neurocomputing*, 112:172–178, 2013.

[15] A. Barachant and M. Congedo. A plug&play P300 BCI using information geometry. *arXiv preprint arXiv:1409.0107*, 2014.

[16] A. Bashashati, M. Fatourechi, R. K. Ward, and G. E. Birch. A survey of signal processing algorithms in brain-computer interfaces based on electrical brain signals. *Journal of Neural Engineering*, 4(2):R35–57, 2007.

[17] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle. Greedy layer-wise training of deep networks. In *Advances in Neural Information Processing Systems*, pages 153–160, 2007.

[18] M. Besserve, J. Martinerie, and L. Garnero. Improving quantification of functional networks with EEG inverse problem: Evidence from a decoding point of view. *Neuroimage*, 2011.

[19] R. Bhatia. *Positive Definite Matrices*. Princeton University Press, 2009.

[20] M. C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[21] B. Blankertz, M. Kawanabe, R. Tomioka, F. Hohlefeld, V. Nikulin, and K.-R. Müller. Invariant common spatial patterns: Alleviating nonstationarities in brain-computer interfacing. In *Advances in Neural Information Processing Systems 20, In . MIT Press, Cambridge, MA*, 2008.

[22] B. Blankertz, S. Lemm, M. Treder, S. Haufe, and K.-R. Müller. Single-trial analysis and classification of ERP components? A tutorial. *Neuroimage*, 2010.

[23] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K.-R. Müller. Optimizing spatial filters for robust EEG single-trial analysis. *IEEE Signal Proc Magazine*, 25(1):41–56, 2008.

[24] J. Blumberg, J. Rickert, S. Waldert, A. Schulze-Bonhage, A. Aertsen, and C. Mehring. Adaptive classification for brain computer interfaces. In *29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 2536–2539, 2007.

[25] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2559–2566. IEEE, 2010.

[26] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[27] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984.

[28] N. Brodu, F. Lotte, and A. Lécuyer. Comparative study of band-power extraction techniques for motor imagery classification. In *Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB), 2011 IEEE Symposium on*, pages 1–6. IEEE, 2011.

[29] N. Brodu, F. Lotte, and A. Lécuyer. Exploring two novel features for EEG-based brain-computer interfaces: Multifractal cumulants and predictive complexity. *Neurocomputing*, 79(1):87–94, 2012.

[30] A. Buttfield, P. Ferrez, and J. Millan. Towards a robust BCI: Error potentials and online learning. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2):164–168, 2006.

[31] N. Caramia, F. Lotte, and S. Ramat. Optimizing spatial filter pairs for EEG classification based on phase synchronization. In *International Conference on Audio, Speech and Signal Processing (ICASSP'2014)*, 2014.

[32] H. Cecotti and A. Graser. Convolutional neural networks for P300 detection with application to brain-computer interfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):433–445, 2011.

[33] R. Chavarriaga, M. Fried-Oken, S. Kleih, F. Lotte, and R. Scherer. Heading for new shores!

overcoming pitfalls in BCI design. *Brain-Computer Interfaces*, pages 1–14, 2017.

[34] S. Chevallier, E. Kalunga, Q. B. elemy, and F. Yger. Riemannian classification for SSVEP-based BCI. In C. Nam, A. Nijholt, and F. Lotte, editors, *BrainComputer Interfaces Handbook: Technological and Theoretical Advances*. Taylor & Francis, 2018, in press.

[35] H. Cho, M. Ahn, K. Kim, and S. Chan Jun. Increasing session-to-session transfer in a braincomputer interface with on-site background noise acquisition. *Journal of Neural Engineering*, 12(6):066009, Dec. 2015.

[36] Z. Chongsheng, L. Changchang, Z. Xiangliang, and A. George. An up-to-date comparison of state-of-the-art classification algorithms. *Expert Systems with Applications*, 82:128–150, 2017.

[37] T. Cibas, F. F. Soulié, P. Gallinari, and S. Raudys. *Variable Selection with Optimal Cell Damage*, pages 727–730. Springer London, London, 1994.

[38] A. Cichocki. Tensor decompositions: A new concept in brain data analysis? *Journal of the Society of Instrument and Control Engineers (SICE)*, 58 (7):507–516, 2011.

[39] A. Cichocki, N. Lee, I. Oseledets, A.-H. Phan, Q.Zhao, and D. Mandic. Tensor networks for dimensionality reduction and large-scale optimization: Part 1 low-rank tensor decompositions. *Foundations and Trends in Machine Learning*, 9(4-5):249–429, 2016.

[40] A. Cichocki, D. Mandic, L. D. Lathauwer, G. Zhou, Q. Zhao, C. Caiafa, and A.-H. Phan. Tensor decompositions for signal processing applications: From two-way to multiway component analysis. *IEEE Signal Processing Magazine*, 32(2):145–163, 2015.

[41] A. Cichocki, A.-H. Phan, Q. Zhao, N. Lee, I. Oseledets, M. Sugiyama, and D. Mandic. Tensor networks for dimensionality reduction and large-scale optimizations. Part 2 applications and future perspectives. *Foundations and Trends in Machine Learning*, 9(6):431–673, 2017.

[42] A. Cichocki, Y. Washizawa, T. Rutkowski, H. Bakardjian, A.-H. Phan, S. Choi, H. Lee, Q. Zhao, L. Zhang, and Y. Li. Noninvasive BCIs: Multiway signal-processing array decompositions. *Computer*, 41(10), 2008.

[43] A. Cichocki, R. Zdunek, A. Phan, and S. Amari. *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*. John Wiley & Sons, 2009.

[44] M. Clerc, L. Bougrain, and F. Lotte. *Brain-Computer Interfaces 1: Foundations and Methods*. ISTE-Wiley, 2016.

[45] M. Clerc, L. Bougrain, and F. Lotte. *Brain-Computer Interfaces 2: Technology and Applications*. ISTE-Wiley, 2016.

[46] M. Congedo. *EEG Source Analysis*. Habilitation à diriger des recherches (HDR), Univ. Grenoble Alpes, 38000 Grenoble, France, TEL, 2013.

[47] M. Congedo, A. Barachant, and R. Bhatia. Riemannian geometry for EEG-based brain-computer interfaces; a primer and a review. *Brain-Computer Interfaces*, 4(3):155–174, 2017.

[48] M. Congedo, A. Barachant, and K. Kharati. Classification of covariance matrices using a riemannian-based kernel for BCI applications. *IEEE Transactions on Signal Processing*, 65(9):2211–2220, 2016.

[49] M. Congedo, F. Lotte, and A. Lécuyer. Classification of movement intention by spatially filtered electromagnetic inverse solutions. *Physics in Medicine and Biology*, 51(8):1971–1989, 2006.

[50] R. Corralejo, R. Hornero, and D. lvarez. Feature selection using a genetic algorithm in a motor imagery-based brain computer interface. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 7703–7706, Aug 2011.

[51] N. Courty, R. Flamary, D. Tuia, and A. Rakotomamonjy. Optimal transport for domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.

[52] D. Coyle, J. Principe, F. Lotte, and A. Nijholt. Guest editorial: Brain/neuronal computer games interfaces and interaction. *IEEE Transactions on Computational Intelligence and AI in Games*, 5(2):77–81, 2013.

[53] H. Daumé III. Frustratingly easy domain adaptation. In *Proceedings of the Association Computational linguistics*, 2007.

[54] S. David, S. Reinhold, F. Josef, and G. R. Müller-Putz. Random forests in non-invasive sensorimotor rhythm brain-computer interfaces: a practical and convenient non-linear classifier. *Biomedical Engineering/Biomedizinische Technik*, 61(1):77–86, 2016.

[55] S. B. David, T. Lu, T. Luu, and D. Pál. Impossibility theorems for domain adaptation. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 129–136, 2010.

[56] J. del Millán. On the need for on-line learning in brain-computer interfaces. In *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, volume 4, pages 2877–2882. IEEE, 2004.

[57] D. Devlaminck, W. Waegeman, B. Bauwens, B. Wyns, P. Santens, and G. Otte. From circular ordinal regression to multilabel classification. In *Preference learning : ECML/PKDD-10 tutorial and workshop, Papers*, page 15, 2010.

[58] T. G. Dietterich and G. Bakiri. Solving multiclass learning problems via error-correcting output codes. *J. Artif. Int. Res.*, 2(1):263–286, Jan. 1995.

[59] S. Ding, N. Zhang, X. Xu, L. Guo, and J. Zhang. Deep extreme learning machine and its application in EEG classification. *Mathematical Problems in Engineering*, 2015, 2015.

[60] G. Dornhege, B. Blankertz, G. Curio, and K. Müller. Boosting bit rates in non-invasive EEG single-trial classifications by feature combination and multi-class paradigms. *IEEE Transactions on Biomedical Engineering*, 51(6):993–1002, 2004.

[61] G. Dornhege, B. Blankertz, M. Krauledat, F. Losch, G. Curio, and K.-R. Müller. Combined optimization of spatial and temporal filters for improving brain-computer interfacing. *IEEE Trans. Biomed. Eng.*, 53(11):2274–2281, 2006.

[62] A. Edelman, A. Tomás, and S. T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2):303–353, 1998.

[63] J. Faller, R. Scherer, U. Costa, E. Opisso, J. Medina, and G. R. Müller-Putz. A co-adaptive brain-computer interface for end users with severe motor impairment. *PloS one*, 9(7):e101168, 2014.

[64] J. Faller, C. Vidaurre, T. Solis-Escalante, C. Neuper, and R. Scherer. Autocalibration and recurrent adaptation: Towards a plug and play online ERD-BCI. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 20(3):313–319, 2012.

[65] J. Farquhar. A linear feature space for simultaneous learning of spatio-spectral filters in BCI. *Neural Networks*, 22(9):1278–1285, 2009.

[66] M. Fatourechi, R. Ward, S. Mason, J. Huggins, A. Schlogl, and G. Birch. Comparison of evaluation metrics in classification applications with imbalanced datasets. In *International Conference on Machine Learning and Applications (ICMLA)*, page 777782. IEEE, 2008.

[67] S. Fazli, F. Popescu, M. Danóczy, B. Blankertz, K.-R. Müller, and C. Grozea. Subject-independent mental state classification in single trials. *Neural Networks*, 22(9):1305–1312, 2009.

[68] P. Ferrez and J. Millán. Error-related EEG potentials generated during simulated brain-computer interaction. *IEEE Transactions on Biomedical Engineering*, 55(3):923–929, 2008.

[69] A. Fischer and C. Igel. An introduction to restricted boltzmann machines. *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 14–36, 2012.

[70] J. Frey, A. Appriou, F. Lotte, and M. Hachet. Classifying EEG signals during stereoscopic visualization to estimate visual comfort. *Computational Intelligence & Neuroscience*, 2015.

[71] K. Fukushima and S. Miyake. Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and Cooperation in Neural Nets*, pages 267–285. Springer, 1982.

[72] J. Gan. Self-adapting BCI based on unsupervised learning. In *3rd International Brain-Computer Interface Workshop*, 2006.

[73] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning*

Research, 17(59):1–35, 2016.

[74] N. T. Gayraud, A. Rakotomamonjy, and M. Clerc. Optimal Transport Applied to Transfer Learning For P300 Detection. In *7th Graz Brain-Computer Interface Conference 2017*, 2017.

[75] X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 315–323, 2011.

[76] B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2066–2073. IEEE, 2012.

[77] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[78] J. Grizou, I. Iturrate, L. Montesano, P.-Y. Oudeyer, and M. Lopes. Calibration-free BCI based control. In *AAAI*, pages 1213–1220, 2014.

[79] M. Grosse-Wentrup. Understanding brain connectivity patterns during motor imagery for brain-computer interfacing. In *Advances in Neural Information Processing Systems (NIPS) 21*, 2009.

[80] M. Grosse-Wentrup. What are the causes of performance variation in brain-computer interfacing? *International Journal of Bioelectromagnetism*, 2011.

[81] Z. Gu, Z. Yu, Z. Shen, and Y. Li. An online semi-supervised brain–computer interface. *IEEE Transactions on Biomedical Engineering*, 60(9):2614–2623, 2013.

[82] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

[83] B. A. S. Hasan and J. Q. Gan. Hangman BCI: An unsupervised adaptive self-paced brain–computer interface for playing games. *Computers in Biology and Medicine*, 42(5):598–606, 2012.

[84] T. Hastie and R. Tibshirani. Classification by pairwise coupling. In *Advances in Neural Information Processing Systems 10, [NIPS Conference, Denver, Colorado, USA, 1997]*, pages 507–513, 1997.

[85] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, 2001.

[86] M. K. Hazrati and A. Erfanian. An online EEG-based brain–computer interface for controlling hand grasp using an adaptive probabilistic neural network. *Medical Engineering & Physics*, 32(7):730–739, 2010.

[87] P. Herman, G. Prasad, T. McGinnity, and D. Coyle. Comparative analysis of spectral approaches to feature extraction for EEG-based motor imagery classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(4):317–326, 2008.

[88] H. Higashi and T. Tanaka. Simultaneous design of FIR filter banks and spatial patterns for EEG signal classification. *IEEE transactions on biomedical engineering*, 60(4):1100–1110, 2013.

[89] G. E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002.

[90] G. E. Hinton, S. Osindero, and Y.-W. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, 2006.

[91] S. Hitziger, M. Clerc, S. Saillet, C. Benar, and T. Papadopoulo. Adaptive Waveform Learning: A Framework for Modeling Variability in Neurophysiological Signals. *IEEE Transactions on Signal Processing*, 2017.

[92] U. Hoffmann, J. Vesin, and T. Ebrahimi. Spatial filters for the classification of event-related potentials. In *European Symposium on Artificial Neural Networks (ESANN 2006)*, 2006.

[93] J. Höhne, E. Holz, P. Staiger-Sälzer, K.-R. Müller, A. Kübler, and M. Tangermann. Motor imagery for severely motor-impaired patients: evidence for brain-computer interfacing as superior control solution. *PLOS One*, 9(8):e104854, 2014.

[94] I. Horev, F. Yger, and M. Sugiyama. Geometry-aware principal component analysis for symmetric positive definite matrices. *ACML*, 2015.

[95] I. Horev, F. Yger, and M. Sugiyama. Geometry-aware stationary subspace analysis. In *ACML*, 2016.

[96] W.-Y. Hsu. EEG-based motor imagery classification using enhanced active segment selection and adaptive classifier. *Computers in Biology and Medicine*, 41(8):633–639, 2011.

[97] V. Jayaram, M. Alamgir, Y. Altun, B. Scholkopf, and M. Grosse-Wentrup. Transfer Learning in Brain-Computer Interfaces. *IEEE Computational Intelligence Magazine*, 11(1):20–31, Feb. 2016.

[98] A. Kachenoura, L. Albera, L. Senhadji, and P. Comon. ICA: A potential tool for BCI systems. *IEEE Signal Processing Magazine*, 25(1):57–68, 2008.

[99] E. Kalunga, S. Chevallier, and Q. Barthélemy. Data augmentation in Riemannian space for brain-computer interfaces. In *ICML Workshop on Statistics, Machine Learning and Neuroscience (Stamlins 2015)*, 2015.

[100] E. Kalunga, S. Chevallier, Q. Barthlemy, K. Djouani, and E. Monacelli. Online SSVEP-based BCI using riemannian geometry. *Neurocomputing*, 191:55–68, 2016.

[101] B. Kamousi, Z. Liu, and B. He. Classification of motor imagery tasks for brain-computer interface applications by means of two equivalent dipoles analysis. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 13(2):166–171, 2005.

[102] H. Kang and S. Choi. Bayesian common spatial patterns for multi-subject EEG classification. *Neural Networks*, 57:39–50, Sept. 2014.

[103] H. Kang, Y. Nam, and S. Choi. Composite common spatial pattern for subject-to-subject transfer. *IEEE Signal Processing Letters*, 16(8):683 – 686, 2009.

[104] P.-J. Kindermans, M. Schreuder, B. Schrauwen, K.-R. Mller, and M. Tangermann. Improving zero-training brain-computer interfaces by mixing model estimators. *PLoS ONE*, 9:e102504, 2014.

[105] P.-J. Kindermans, M. Tangermann, K.-R. Müller, and B. Schrauwen. Integrating dynamic stopping, transfer learning and language models in an adaptive zero-training erp speller. *Journal of Neural Engineering*, 11(3):035005, 2014.

[106] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1):273 – 324, 1997. Relevance.

[107] I. Koprinska. *Feature Selection for Brain-Computer Interfaces*, pages 106–117. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[108] M. Krauledat, M. Schröder, B. Blankertz, and K.-R. Müller. Reducing calibration time for brain-computer interfaces: A clustering approach. In *In B. Scholkopf, J. Platt, and T. Hoffman, editors, Advances in Neural Information Processing Systems (NIPS 07) 19, Cambridge, MA, 2007. MIT Press.*, pages 753–760, 2007.

[109] D. Krusienski, M. Grosse-Wentrup, F. Galán, D. Coyle, K. Miller, E. Forney, and C. Anderson. Critical issues in state-of-the-art brain-computer interface signal processing. *Journal of Neural Engineering*, 8(2):025002, 2011.

[110] D. Krusienski, D. McFarland, and J. Wolpaw. Value of amplitude, phase, and coherence features for a sensorimotor rhythm-based brain–computer interface. *Brain Research Bulletin*, 87(1):130–134, 2012.

[111] D. Krusienski, E. Sellers, F. Cabestaing, S. Bayoudh, D. McFarland, T. Vaughan, and J. Wolpaw. A comparison of classification techniques for the P300 speller. *Journal of Neural Engineering*, 3:299–305, 2006.

[112] A. Kübler, E. M. Holz, A. Riccio, C. Zickler, T. Kaufmann, S. C. Kleih, P. Staiger-Sälzer, L. Desideri, E.-J. Hoogerwerf, and D. Mattia. The user-centered design as novel perspective for evaluating the usability of BCI-controlled applications. *PLoS One*, 9(12):e112392, 2014.

[113] N.-S. Kwak, K.-R. Müller, and S.-W. Lee. A convolutional neural network for steady state visual evoked potential classification under ambulatory environment. *PloS One*, 12(2):e0172578, 2017.

[114] K. LaFleur, K. Cassady, A. Doud, K. Shades, E. Rogin, and B. He. Quadcopter control in three-dimensional space using a non-invasive motor imagery-based braincomputer interface. *Journal*

of Neural Engineering, 10(4):046003, 2013.

[115] T. Lal, M. Schröder, T. Hinterberger, J. Weston, M. Bogdan, N. Birbaumer, and B. Schölkopf. Support vector channel selection in BCI. *IEEE TBME*, 51(6):1003?1010, 2004.

[116] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance. EEGNet: A compact convolutional network for EEG-based brain-computer interfaces. *arXiv preprint arXiv:1611.08024*, 2016.

[117] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, 1989.

[118] O. Ledoit and M. Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411, 2004.

[119] J. Li and L. Zhang. Bilateral adaptation and neurofeedback for brain computer interface system. *Journal of Neuroscience Methods*, 193(2):373–379, 2010.

[120] S. Z. Li. *Markov random field modeling in image analysis*. Springer Science & Business Media, 2009.

[121] Y. Li and C. Guan. Joint feature re-extraction and classification using an iterative semi-supervised support vector machine algorithm. *Machine Learning*, 71(1):33–53, 2008.

[122] Y. Li, C. Guan, H. Li, and Z. Chin. A self-training semi-supervised SVM algorithm and its application in an EEG-based brain computer interface speller system. *Pattern Recognition Letters*, 29(9):1285–1294, 2008.

[123] N. Liang and L. Bougrain. Decoding finger flexion from band-specific ECoG signals in humans. *Frontiers in Neuroscience*, 6:91, 2012.

[124] J. T. Lindgren. As above, so below? Towards understanding inverse models in BCI. *Journal of Neural Engineering*, 2017.

[125] C. Lindig-León. *Multilabel classification of EEG-based combined motor imageries implemented for the 3D control of a robotic arm. (Classification multilabels à partir de signaux EEG d'imaginations motrices combinées : application au contrôle 3D d'un bras robotique)*. PhD thesis, University of Lorraine, Nancy, France, 2017.

[126] C. Lindig-León and Bougrain. Comparison of sensorimotor rhythms in EEG signals during simple and combined motor imageries over the contra and ipsilateral hemispheres. In *37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Milan Italy (EMBC2015)*, 2015.

[127] C. Lindig-León and L. Bougrain. A multilabel classification method for detection of combined motor imageries. In *2015 IEEE international conference on systems, man, and cybernetics (SMC2015)*, 2015.

[128] C. Lindig-León, N. Gayraud, L. Bougrain, and M. Clerc. Hierarchical classification using Riemannian geometry for motor imagery based BCI systems. In *BCI meeting 2016, Asilomar, California, USA.*, 2016.

[129] G. Liu, G. Huang, J. Meng, D. Zhang, and X. Zhu. Improved GMM with parameter initialization for unsupervised adaptation of brain–computer interface. *International Journal for Numerical Methods in Biomedical Engineering*, 26(6):681–691, 2010.

[130] G. Liu, D. Zhang, J. Meng, G. Huang, and X. Zhu. Unsupervised adaptation of electroencephalogram signal processing based on fuzzy C-means algorithm. *International Journal of Adaptive Control and Signal Processing*, 26(6):482–495, 2012.

[131] A. Llera, V. Gómez, and H. J. Kappen. Adaptive classification on brain-computer interfaces using reinforcement signals. *Neural Computation*, 24(11):2900–2923, 2012.

[132] A. Llera, V. Gómez, and H. J. Kappen. Adaptive multiclass classification for brain computer interfaces. *Neural Computation*, 26(6):1108–1127, 2014.

[133] A. Llera, M. A. van Gerven, V. Gómez, O. Jensen, and H. J. Kappen. On the use of interaction error potentials for adaptive brain computer interfaces. *Neural Networks*, 24(10):1120–1127, 2011.

[134] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu. Transfer feature learning with joint distribution adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2200–2207, 2013.

[135] F. Lotte. A new feature and associated optimal spatial filter for EEG signal classification: Waveform length. In *International Conference on Pattern Recognition (ICPR)*, pages 1302–1305, 2012.

[136] F. Lotte. A tutorial on EEG signal-processing techniques for mental-state recognition in brain–computer interfaces. In *Guide to Brain-Computer Music Interfacing*, pages 133–161. Springer, 2014.

[137] F. Lotte. Signal processing approaches to minimize or suppress calibration time in oscillatory activity-based brain–computer interfaces. *Proceedings of the IEEE*, 2015.

[138] F. Lotte. *Towards Usable Electroencephalography-based Brain-Computer Interfaces*. Habilitation thesis / habilitation à diriger des recherches (HDR), Univ. Bordeaux, 2016.

[139] F. Lotte, L. Bougrain, and M. Clerc. Electroencephalography (EEG)-based brain-computer interfaces. In *Wiley Encyclopedia on Electrical and Electronices Engineering*. Wiley, 2015.

[140] F. Lotte and M. Congedo. *EEG Feature Extraction*, pages 127–143. Wiley Online Library, 2016.

[141] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi. A review of classification algorithms for EEG-based brain-computer interfaces. *Journal of Neural Engineering*, 4:R1–R13, 2007.

[142] F. Lotte and C. Guan. An efficient P300-based brain-computer interface with minimal calibration time. In *Assistive Machine Learning for People with Disabilities Symposium (NIPS'09 Symposium)*, 2009.

[143] F. Lotte and C. Guan. Regularizing common spatial patterns to improve BCI designs: Unified theory and new algorithms. *IEEE Transactions on Biomedical Engineering*, 58(2):355–362, 2011.

[144] F. Lotte and C. Jeunet. Towards improved BCI based on human learning principles. In *3rd International Brain-Computer Interfaces Winter Conference*, 2015.

[145] F. Lotte and C. Jeunet. Online classification accuracy is a poor metric to study mental imagery-based BCI user learning: an experimental demonstration and new metrics. In *International Brain-Computer Interface Conference*, 2017.

[146] F. Lotte, C. Jeunet, J. Mladenovic, B. N'Kaoua, and L. Pillette. *A BCI challenge for the signal processing community: Considering the human in the loop*. IET, 2018.

[147] F. Lotte, F. Larrue, and C. Mühl. Flaws in current human training protocols for spontaneous brain-computer interfaces: lessons learned from instructional design. *Frontiers in Human Neuroscience*, 7(568), 2013.

[148] F. Lotte, A. Lécuyer, and B. Arnaldi. FuRIA: An inverse solution based feature extraction algorithm using fuzzy set theory for brain-computer interfaces. *IEEE transactions on Signal Processing*, 57(8):3253–3263, 2009.

[149] D. Lowne, S. J. Roberts, and R. Garnett. Sequential non-stationary dynamic classification with sparse feedback. *Pattern Recognition*, 43(3):897–905, 2010.

[150] N. Lu, T. Li, X. Ren, and H. Miao. A deep learning scheme for motor imagery classification based on restricted Boltzmann machines. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(6):566–576, 2017.

[151] S. Lu, C. Guan, and H. Zhang. Unsupervised brain computer interface based on inter-subject information and online adaptation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 17(2):135–145, 2009.

[152] S. Luke. *Essentials of Metaheuristics*. Lulu, 2013.

[153] T. Ma, H. Li, H. Yang, X. Lv, P. Li, T. Liu, D. Yao, and P. Xu. The extraction of motion-onset VEP BCI features based on deep learning and compressed sensing. *Journal of Neuroscience Methods*, 275:80–92, 2017.

[154] G. Madjarov, D. Kocev, D. Gjorgjevikj, and S. Deroski. An extensive experimental comparison

of methods for multi-label learning. *Pattern Recognition*, 45(9):3084 – 3104, 2012. Best Papers of Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA'2011).

[155] S. Makeig, C. Kothe, T. Mullen, N. Bigdely-Shamlo, Z. Zhang, and K. Kreutz-Delgado. Evolving signal processing for brain–computer interfaces. *Proceedings of the IEEE*, 100:1567–1584, 2012.

[156] R. Manor and A. B. Geva. Convolutional neural network for multi-category rapid serial visual presentation BCI. *Frontiers in Computational Neuroscience*, 9, 2015.

[157] P. Margaux, M. Emmanuel, D. Sébastien, B. Olivier, and M. Jérémie. Objective and subjective evaluation of online error correction during P300-based spelling. *Advances in Human-Computer Interaction*, 2012:4, 2012.

[158] L. Mayaud, S. Cabanilles, A. V. Langhenhove, M. Congedo, A. Barachant, S. Pouplin, S. Filipe, L. Ptgnief, O. Rochecouste, E. Azabou, C. Hugeron, M. Lejaille, D. Orlikowski, and D. Annane. Brain-computer interface for the communication of acute patients: A feasibility study and a randomized controlled trial comparing performance with healthy participants and a traditional assistive device. *Brain-Computer Interfaces*, 3(4):197–215, 2016.

[159] D. McFarland, W. Sarnacki, and J. Wolpaw. Should the parameters of a BCI translation algorithm be continually adapted? *Journal of Neuroscience Methods*, 199(1):103 – 107, 2011.

[160] D. J. McFarland, L. M. McCane, S. V. David, and J. R. Wolpaw. Spatial filter selection for EEG-based communication. *Electroencephalographic Clinical Neurophysiology*, 103(3):386–394, 1997.

[161] J. Meng, L. Yao, X. Sheng, D. Zhang, and X. Zhu. Simultaneously optimizing spatial spectral features based on mutual information for EEG classification. *IEEE Transactions on Biomedical Engineering*, 62(1):227–240, 2015.

[162] J. Meng, S. Zhang, A. Bekyo, J. Olsoe, B. Baxter, and B. He. Noninvasive electroencephalogram based control of a robotic arm for reach and grasp tasks, 2016.

[163] J. Millán, F. Renkens, J. Mouriño, and W. Gerstner. Noninvasive brain-actuated control of a mobile robot by human EEG. *IEEE Transactions on Biomedical Engineering*, 51(6):1026–1033, 2004.

[164] J. Mladenovic, J. Mattout, and F. Lotte. A generic framework for adaptive EEG-based BCI training and operation. In C. Nam, A. Nijholt, and F. Lotte, editors, *Handbook of Brain-Computer Interfaces*. Taylor & Francis, 2017, in press.

[165] H. Morioka, A. Kanemura, J.-i. Hirayama, M. Shikauchi, T. Ogawa, S. Ikeda, M. Kawanabe, and S. Ishii. Learning a common dictionary for subject-transfer decoding with resting calibration. *NeuroImage*, 111:167–178, May 2015.

[166] C. Mühl, C. Jeunet, and F. Lotte. EEG-based workload estimation across affective contexts. *Frontiers in Neuroscience section Neuroprosthetics*, 8:114, 2014.

[167] T. Mullen, C. Kothe, Y. M. Chi, A. Ojeda, T. Kerth, S. Makeig, G. Cauwenberghs, and T.-P. Jung. Real-time modeling and 3D visualization of source dynamics and connectivity using wearable EEG. In *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, volume 2013, pages 2184–2187, 2013.

[168] J. S. Müller, C. Vidaurre, M. Schreuder, F. C. Meinecke, P. von Bünau, and K.-R. Müller. A mathematical model for the two-learners problem. *Journal of Neural Engineering*, 14(3):036005, 2017.

[169] K. R. Müller, M. Krauledat, G. Dornhege, G. Curio, and B. Blankertz. Machine learning techniques for brain-computer interfaces. *Biomedical Technologies*, 49:11–22, 2004.

[170] C. Neuper and G. Pfurtscheller. *Brain-Computer Interfaces*, chapter Neurofeedback Training for BCI Control, pages 65–78. The Frontiers Collection, 2010.

[171] L. F. Nicolas-Alonso, R. Corralejo, J. Gomez-Pilar, D. Álvarez, and R. Hornero. Adaptive semi-supervised classification to reduce intersession non-stationarity in multiclass motor imagery-based brain–computer interfaces. *Neurocomputing*, 159:186–196, 2015.

[172] E. Niedermeyer and F. L. da Silva. *Electroencephalography: basic principles, clinical applications, and related fields*. Lippincott Williams & Wilkins, ISBN 0781751268, 5th edition, 2005.

[173] Q. Noirhomme, R. Kitney, and B. Macq. Single trial EEG source reconstruction for brain-

computer interface. *IEEE Transactions on Biomedical Engineering*, 55(5):1592–1601, 2008.

[174] E. S. Nurse, P. J. Karoly, D. B. Grayden, and D. R. Freestone. A generalizable brain-computer interface (BCI) using machine learning for feature discovery. *PLOS ONE*, 10(6):1–22, 06 2015.

[175] A. Onishi, A. Phan, K. Matsuoka, and A. Cichocki. Tensor classification for P300-based brain computer interface. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 581–584. IEEE, 2012.

[176] J. Ortega, J. Asensio-Cubero, J. Q. Gan, and A. Ortiz. Classification of motor imagery tasks for BCI with multiresolution analysis and multiobjective feature selection. *Biomed Eng Online*, 15(15), 2016.

[177] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[178] R. C. Panicker, S. Puthusserypady, and Y. Sun. Adaptation in P300 brain–computer interfaces: A two-classifier cotraining approach. *IEEE Transactions on Biomedical Engineering*, 57(12):2927–2935, 2010.

[179] E. Parzen. On estimation of a probability density function and mode. *Ann. Math. Stat.*, 33:1065 1076, 1962.

[180] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.

[181] G. Pfurtscheller, G. Müller-Putz, R. Scherer, and C. Neuper. Rehabilitation with brain-computer interface systems. *IEEE Computer*, 41(10):58–65, 2008.

[182] A. Phan, A. Cichocki, P. Tichavskỳ, and S. Zdunek, R.and Lehky. From basis components to complex structural patterns. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3228–3232. IEEE, 2013.

[183] A.-H. Phan and A. Cichocki. Tensor decompositions for feature extraction and classification of high dimensional datasets. *Nonlinear Theory and its Applications (NOLTA), IEICE*, 1(1):37–68, 2010.

[184] J. R. Quinlan. Induction of decision trees. *Mach. Learn.*, 1(1):81–106, Mar. 1986.

[185] H. Ramoser, J. Muller-Gerking, and G. Pfurtscheller. Optimal spatial filtering of single trial EEG during imagined hand movement. *IEEE Transactions on Rehabilitation Engineering*, 8(4):441–446, 2000.

[186] A. M. Ray, R. Sitaram, M. Rana, E. Pasqualotto, K. Buyukturkoglu, C. Guan, K.-K. Ang, C. Tejos, F. Zamorano, F. Aboitiz, N. Birbaumer, and S. Ruiz. A subject-independent pattern-based Brain-Computer Interface. *Frontiers in Behavioral Neuroscience*, 9, Oct. 2015.

[187] B. Rivet, H. Cecotti, R. Phlypo, O. Bertrand, E. Maby, and J. Mattout. EEG sensor selection by sparse spatial filtering in P300 speller brain-computer interface. In *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, pages 5379–5382. IEEE, 2010.

[188] B. Rivet, A. Souloumiac, V. Attina, and G. Gibert. xDAWN algorithm to enhance evoked potentials: Application to brain computer interface. *IEEE Transactions on Biomedical Engineering*, 56(8):2035–2043, 2009.

[189] P. Rodrigues, F. Bouchard, M. Congedo, and C. Jutten. Dimensionality reduction for BCI classification using Riemannian geometry. In *7th Graz Brain-Computer Interface Conference, Sep 2017, Graz, Austria*, pages –, 2017.

[190] L. Roijendijk, S. Gielen, and J. Farquhar. Classifying regularized sensor covariance matrices: An alternative to CSP. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(8):893–900, Aug 2016.

[191] R. N. Roy, S. Charbonnier, A. Campagne, and S. Bonnet. Efficient mental workload estimation using task-independent EEG features. *Journal of Neural Engineering*, 13(2):026019, 2016.

[192] A. S. Royer, A. J. Doud, M. L. Rose, and B. He. EEG control of a virtual helicopter in 3-dimensional space using intelligent control strategies. *IEEE Transactions on Neural Systems*

and Rehabilitation Engineering, 18(6):581–589, Dec 2010.

[193] D. E. Rumelhart, G. E. Hinton, R. J. Williams, et al. Learning representations by back-propagating errors. *Cognitive Modeling*, 5(3):1, 1988.

[194] W. Samek, M. Kawanabe, and K.-R. Müller. Divergence-based framework for common spatial patterns algorithms. *IEEE Reviews in Biomedical Engineering*, 2014.

[195] C. Sannelli, C. Vidaurre, K.-R. Müller, and B. Blankertz. CSP patches: an ensemble of optimized spatial filters. an evaluation study. *J. Neural Eng.*, 8, 2011.

[196] C. Sannelli, C. Vidaurre, K.-R. Müller, and B. Blankertz. Ensembles of adaptive spatial filters increase BCI performance: an online evaluation. *Journal of Neural Engineering*, 13(4):046003, 2016.

[197] F. Schettini, F. Aloise, P. Aricò, S. Salinari, D. Mattia, and F. Cincotti. Self-calibration algorithm in an asynchronous P300-based brain–computer interface. *Journal of Neural Engineering*, 11(3):035004, 2014.

[198] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, and T. Ball. Deep learning with convolutional neural networks for EEG decoding and visualization. *Human Brain Mapping*, 2017.

[199] A. Schlögl, J. Kronegg, J. Huggins, and S. G. Mason. *Towards Brain-Computer Interfacing*, chapter Evaluation criteria in BCI research, pages 327–342. MIT Press, 2007.

[200] A. Schlögl, C. Vidaurre, and K.-R. Müller. Adaptive methods in BCI research-an introductory tutorial. In *Brain-Computer Interfaces*, pages 331–355. Springer, 2010.

[201] B. D. Seno, M. Matteucci, and L. Mainardi. A genetic algorithm for automatic feature extraction in P300 detection. In *IEEE International Joint Conference on Neural Networks*, pages 3145–3152, June 2008.

[202] P. Shenoy, M. Krauledat, B. Blankertz, R. Rao, and K.-R. Müller. Towards adaptive classification for BCI. *Journal of Neural Engineering*, 3(1):R13, 2006.

[203] S. Si, D. Tao, and B. Geng. Bregman divergence-based regularization for transfer subspace learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(7):929–942, 2010.

[204] X. Song, S.-C. Yoon, and V. Perera. Adaptive common spatial pattern for single-trial EEG classification in multisubject BCI. In *International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 411–414. IEEE, 2013.

[205] A. Soria-Frisch. A Critical Review on the Usage of Ensembles for BCI. In B. Z. Allison, S. Dunne, R. Leeb, J. Del R. Millán, and A. Nijholt, editors, *Towards Practical Brain-Computer Interfaces*, Biological and Medical Physics, Biomedical Engineering, pages 41–65. Springer Berlin Heidelberg, 2012.

[206] M. Spüler, W. Rosenstiel, and M. Bogdan. Online adaptation of a c-VEP brain-computer interface (BCI) based on error-related potentials and unsupervised learning. *PloS one*, 7(12):e51077, 2012.

[207] I. Sturm, S. Lapuschkin, W. Samek, and K.-R. Müller. Interpretable deep neural networks for single-trial EEG classification. *Journal of Neuroscience Methods*, 274:141–145, 2016.

[208] M. Sugiyama, S. Nakajima, H. Kashima, P. V. Buenau, and M. Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Advances in Neural Information Processing Systems*, pages 1433–1440, 2008.

[209] P. Sykacek, S. J. Roberts, and M. Stokes. Adaptive BCI based on variational bayesian kalman filtering: an empirical evaluation. *IEEE Transactions on Biomedical Engineering*, 51:719–729, 2004.

[210] Y. R. Tabar and U. Halici. A novel deep learning approach for classification of EEG motor imagery signals. *Journal of Neural Engineering*, 14(1):016003, 2016.

[211] D. B. Thiyam, S. Cruces, J. Olias, and A. Cichocki. Optimization of alpha-beta log-det divergences and their application in the spatial filtering of two class motor imagery movements. *Entropy*, 19(3):89, 2017.

[212] E. Thomas, M. Dyson, and M. Clerc. An analysis of performance evaluation for motor-imagery

based bci. *Journal of neural engineering*, 10(3):031001, 2013.

[213] T. Tieleman. Training restricted boltzmann machines using approximations to the likelihood gradient. In *Proceedings of the 25th International Conference on Machine Learning*, pages 1064–1071. ACM, 2008.

[214] R. Tomioka and K.-R. Müller. A regularized discriminative framework for EEG analysis with application to brain-computer interface. *Neuroimage*, 49(1):415–432, 2010.

[215] G. Tsoumakas and I. Katakis. Multilabel classification: An overview. *International Journal of Data Warehousing & Mining*, 3(3):1–13, 2007.

[216] J. van Erp, F. Lotte, and M. Tangermann. Brain-computer interfaces: Beyond medical applications. *IEEE Computer*, 45(4):26–34, 2012.

[217] T. Vaughan, D. McFarland, G. Schalk, W. Sarnacki, D. Krusienski, E. Sellers, and J. Wolpaw. The wadsworth BCI research and development program: at home with BCI. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2):229–233, 2006.

[218] T. Verhoeven, D. Hbner, M. Tangermann, K.-R. Mller, J. Dambre, and P.-J. Kindermans. True zero-training brain-computer interfacing - an online study. *Journal of Neural Engineering*, 14(3):036021, 2017.

[219] C. Vidaurre, M. Kawanabe, P. Von Bunau, B. Blankertz, and K. Muller. Toward unsupervised adaptation of LDA for brain–computer interfaces. *IEEE Transactions on Biomedical Engineering*, 58(3):587–597, 2011.

[220] C. Vidaurre, C. Sannelli, K.-R. Müller, and B. Blankertz. Co-adaptive calibration to improve BCI efficiency. *Journal of Neural Engineering*, 8(2):025009, 2011.

[221] C. Vidaurre, C. Sannelli, K.-R. Müller, and B. Blankertz. Machine-learning-based coadaptive calibration for brain-computer interfaces. *Neural Computation*, 23(3):791–816, 2011.

[222] C. Vidaurre, A. Schlögl, R. Cabeza, R. Scherer, and G. Pfurtscheller. Study of on-line adaptive discriminant analysis for EEG-based brain computer interfaces. *IEEE Transactions on Biomedical Engineering*, 54(3):550–556, 2007.

[223] Y. Washizawa, H. Higashi, T. Rutkowski, T. Tanaka, and A. Cichocki. Tensor based simultaneous feature extraction and sample weighting for EEG classification. *Neural Information Processing. Models and Applications*, pages 26–33, 2010.

[224] N. Waytowich, V. Lawhern, A. Bohannon, K. Ball, and B. Lance. Spectral transfer learning using information geometry for a user-independent brain-computer interface. *Frontiers in Neuroscience*, 10:430, 2016.

[225] Q. Wei, Y. Wang, X. Gao, and S. Gao. Amplitude and phase coupling measures for feature extraction in an EEG–based brain–computer interface. *Journal of Neural Engineering*, 4(2):120, 2007.

[226] Y. Weibo, Q. Shuang, Q. Hongzhi, Z. Lixin, W. Baikun, and M. Dong. EEG feature comparison and classification of simple and compound limb motor imagery. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2010.

[227] H. Woehrle, M. M. Krell, S. Straube, S. K. Kim, E. A. Kirchner, and F. Kirchner. An adaptive spatial filter for user-independent single trial detection of event-related potentials. *IEEE Transactions on Biomedical Engineering*, 62(7):1696–1705, 2015.

[228] J. Wolpaw, N. Birbaumer, D. McFarland, G. Pfurtscheller, and T. Vaughan. Brain-computer interfaces for communication and control. *Clinical Neurophysiology*, 113(6):767–791, 2002.

[229] J. Wolpaw and E. Wolpaw. *Brain-Computer Interfaces: Principles and Practice*. Oxford University Press, 2012.

[230] J. R. Wolpaw, D. J. McFarland, G. W. Neat, and C. A. Forneris. An EEG-based brain-computer interface for cursor control. *Electroencephalography and Clinical Neurophysiology*, 78:252–259, 1991.

[231] F. Yger. A review of kernels on covariance matrices for BCI applications. In *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE, 2013.

[232] F. Yger, M. Berar, and F. Lotte. Riemannian approaches in brain-computer interfaces: A review.

IEEE Transactions on Neural System and Rehabilitation Engineering, 2017.

[233] F. Yger, F. Lotte, and M. Sugiyama. Averaging covariance matrices for EEG signal classification based on the CSP: An empirical study. In *23rd European Signal Processing Conference (EUSIPCO)*, pages 2721–2725. IEEE, 2015.

[234] Z. Yin and J. Zhang. Cross-session classification of mental workload levels using EEG and an adaptive deep learning model. *Biomedical Signal Processing and Control*, 33:30–47, 2017.

[235] Z. Yin, M. Zhao, Y. Wang, J. Yang, and J. Zhang. Recognition of emotions using multimodal physiological signals and an ensemble deep learning model. *Computer Methods and Programs in Biomedicine*, 140:93–110, 2017.

[236] J. W. Yoon, S. J. Roberts, M. Dyson, and J. Q. Gan. Adaptive classification for brain computer interface systems using sequential monte carlo sampling. *Neural Networks*, 22(9):1286–1294, 2009.

[237] T. Zander and C. Kothe. Towards passive brain-computer interfaces: applying brain-computer interface technology to human-machine systems in general. *Journal of Neural Engineering*, 8, 2011.

[238] P. Zanini, M. Congedo, C. Jutten, S. Said, and Y. Berthoumieu. Transfer learning: a Riemannian geometry framework with applications to brain-computer interfaces. *IEEE Transactions on Biomedical Engineering*, 2017.

[239] T. Zeyl, E. Yin, M. Keightley, and T. Chau. Partially supervised P300 speller adaptation for eventual stimulus timing optimization: target confidence is superior to error-related potential score as an uncertain label. *Journal of Neural Engineering*, 13(2):026008, 2016.

[240] H. Zhang, R. Chavarriaga, and J. d. R. Millán. Discriminant brain connectivity patterns of performance monitoring at average and single-trial levels. *NeuroImage*, 120:64–74, 2015.

[241] K. Zhang, V. Zheng, Q. Wang, J. Kwok, Q. Yang, and I. Marsic. Covariate shift in hilbert space: A solution via sorrogate kernels. In *International Conference on Machine Learning*, pages 388–395, 2013.

[242] Y. Zhang, G. Zhou, J. Jin, M. Wang, X. Wang, and A. Cichocki. L1-regularized multiway canonical correlation analysis for SSVEP-based BCI. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 21(6):887–896, 2013.

[243] Y. Zhang, G. Zhou, J. Jin, X. Wang, and A. Cichocki. Frequency recognition in SSVEP-based BCI using multiset canonical correlation analysis. *International Journal of Neural Systems*, 24(04):1450013, 2014.

[244] Y. Zhang, G. Zhou, J. Jin, X. Wang, and A. Cichocki. Optimizing spatial patterns with sparse filter bands for motor-imagery based brain–computer interface. *Journal of Neuroscience Methods*, 255:85–91, 2015.

[245] Y. Zhang, G. Zhou, J. Jin, Y. Zhang, X. Wang, and A. Cichocki. Sparse Bayesian multiway canonical correlation analysis for EEG pattern recognition. *Neurocomputing*, 225:103–110, 2017.

[246] Y. Zhang, G. Zhou, Q. Zhao, A. Onishi, J. Jin, X. Wang, and A. Cichocki. Multiway canonical correlation analysis for frequency components recognition in SSVEP-based BCIs. In *Neural Information Processing*, pages 287–295. Springer, 2011.

[247] Q. Zhao, L. Zhang, A. Cichocki, and J. Li. Incremental common spatial pattern algorithm for BCI. In *IEEE International Joint Conference on Neural Networks (IJCNN)*, pages 2656 –2659, june 2008.

[248] S.-M. Zhou, J. Q. Gan, and F. Sepulveda. Classifying mental tasks based on features of higher-order statistics from EEG signals in brain–computer interface. *Information Sciences*, 178(6):1629–1640, 2008.

[249] Z. Zhou, B. Wan, D. Ming, and H. Qi. A novel technique for phase synchrony measurement from the complex motor imaginary potential of combined body and limb action. *J Neural Eng*, 7, 2010.