The largest EEG-based BCI reproducibility study for open science: the MOABB benchmark

Sylvain Chevallier, Igor Carrara, Bruno Aristimunha, Pierre Guetschel, Sara Sedlar, Bruna Lopes, Sebastien Velut, Salim Khazem, Thomas Moreau

Inria TAU, LISN-CNRS, Université Paris-Saclay, 91405, Orsay, France
Université Côte d'Azur, Inria Cronos Team, Sophia Antipolis, France
Donders Institute, Radboud University, Nijmegen, Netherlands
University of São Paulo, Sao Paulo, Brazil
Federal University of ABC, Santo Andre, Brazil
GeorgiaTech-CNRS IRL 2958, Centralesupelec, Metz, France
Inria Mind team, Université Paris-Saclay, CEA, Palaiseau, 91120, France

E-mail: sylvain.chevallier@universite-paris-saclay.fr

Abstract.

Objective. This study conduct an extensive Brain-computer interfaces (BCI) reproducibility analysis on open electroencephalography datasets, aiming to assess existing solutions and establish open and reproducible benchmarks for effective comparison within the field. The need for such benchmark lies in the rapid industrial progress that has given rise to undisclosed proprietary solutions. Furthermore, the scientific literature is dense, often featuring challenging-to-reproduce evaluations, making comparisons between existing approaches arduous. *Approach.* Within an open framework, 30 machine learning pipelines (separated into raw signal: 11, Riemannian: 13, deep learning: 6) are meticulously re-implemented and evaluated across 36 publicly available datasets, including motor imagery (14), P300 (15), and SSVEP (7). The analysis incorporates statistical meta-analysis techniques for results assessment, encompassing execution time and environmental impact considerations. *Main results.* The study yields principled and robust results applicable to various BCI paradigms, emphasizing motor imagery, P300, and SSVEP. Notably, Riemannian approaches utilizing spatial covariance matrices exhibit superior performance, underscoring the necessity for significant data volumes to achieve competitive outcomes with deep learning techniques. The comprehensive results are openly accessible, paving the way for future research to further enhance reproducibility in the BCI domain. *Significance.* The significance of this study lies in its contribution to establishing a rigorous and transparent benchmark for BCI research, offering insights into optimal methodologies and highlighting the importance of reproducibility in driving advancements within the field.

Keywords: Brain-computer interface, EEG, Reproducibility, Riemannian classifier, Deep learning

Submitted to: *J. Neural Eng.*

1. Introduction

The field of Brain-Computer Interface (BCI) aims at developing methodologies to allow interactions with devices, like prostheses or computer environments, from decoding brain signals. It is a very good candidate technology to assist people with motor disability, as it requires very limited motor capabilities of the subject. To go from brain signals to decision-making, BCI systems are defined by several choices: a paradigm, that specifies the cognitive tasks to perform to control the interface, an acquisition device, to record the brain activity, and an algorithmic pipeline, that processes the acquired data and predicts which action the subject intends to perform. As such, BCI is at the forefront of interdisciplinary research, integrating expertise from diverse fields such as electronics, neuroscience, human-machine interaction (HMI), signal processing, and machine learning.

1.1. Open data

To fasten the emergence of BCI system, the field has organized to decouple the lengthy paradigm design and data acquisition processes from the development of algorithmic pipelines to process them. The creation and publication of many accessible and openly available datasets allow for offline development and evaluation of novel processing pipelines. They also improve experiment replication and ensure the replicability of published results.

Due to the constraints on real-time acquisitions in open environments, Electroencephalography (EEG) has become the leading device to develop BCI systems with its high-frequency acquisition and limited constraints on its deployment. In EEG-based BCI, many real-world competitions and open datasets exist on a world scale to design and evaluate the best BCI systems. BCI datasets come with various paradigms, depending on the decoding task to perform from brain signals. Motor Imagery (MI) is a common paradigm choice to control a BCI, with different imagined movements, but other paradigms are also very efficient like Event Related Potential (ERP) generated by oddball stimulus or Steady State Visually Evoked Potential (SSVEP) produced by repetitive stimulations. Many open datasets are thus openly available for EEG-based BCI.

Yet, open data is not solely about availability. It also requires the use of open formats that enable easy reading and exploitation of metadata. Indeed, there exists a wide variety of EEG acquisition devices with various hardware specifications, as well as a plethora of paradigm specifications, with their experimental design, shared annotations, and code for interpreting the cued events. Retrieving this information is critical to developing valid systems that can correctly process the retrieved data. Drawing inspiration from brain imaging techniques like fMRI or MEG that rely on the Brain Imaging Data Structure (BIDS), an extension of this format has been proposed for EEG-based research [91]. Despite this proposed format, most EEG open data available online are stored in diverse structures and formats, which hinders the automation of data collection, as each dataset requires specific processing scripts.

Moreover, while numerous books on EEG data acquisition and BCI exist, it remains challenging to find well-founded guidelines for hardware requirements and BCI design. These guidelines are needed to allow practitioners to make informed decisions regarding experimental design and hardware selection. In the worst-case scenario, a poorly designed BCI or an inappropriate hardware choice can result

in unusable data. The unique characteristics of EEG-based BCIs could be guided by meta-analyses of existing datasets, leading to recommendations on the number and positioning of electrodes, sampling frequency, the required number of subject trials, and the influence of the number of classes. However, to the best of our knowledge, there is no comprehensive and systematic evaluation of these parameters available in the existing literature.

1.2. BCI pipelines

Based on these open datasets, the development of BCI classification pipelines for EEG signals has a long-standing research history and a very active community [75]. Pipelines initially focused on methods based on feature extraction from raw signals, such as channel variance or local variation [74], combined with complex classifiers. The first breakthrough came with the introduction of *spatial filtering* methods based on covariance estimation, such as XDAWN [95], CSP [2], or CCA [72]. These methods significantly improved classification accuracy when paired with simple classifiers. By using supervised filters, the separability of different BCI classes was improved while reducing the dimensionality of the input signal. As a result, these raw-signal-based pipelines with spatial filtering became the top-ranked approaches in BCI competitions. The second advancement in BCI classification involved a reformulation of spatial filtering based on covariance matrices using *Riemannian geometry* [12, 124]. Leveraging the inherent manifold structure of symmetric covariance matrices, this approach provides robust classification by exploiting the invariance properties of the covariance manifold. This method has demonstrated remarkable performance across various BCI tasks, even with noisy EEG signals, making it the preferred choice and quickly outperforming other pipelines in BCI competitions [33]. Pipelines based on Riemannian geometry are considered state-of-the-art methods in the current literature.

Recently, *deep learning methods* have been explored for EEG-based BCI classification [99]. Although deep learning has enjoyed success in computer vision tasks, it is not straightforward to adapt these techniques to handle strongly correlated time series data, such as EEG signals. As a result, the deep learning approaches performances on EEG signals have not outperformed existing approaches [103]. Indeed, the main workhorse for the success of deep learning lies in the availability of vast amounts of data. However, in the context of BCI and EEG data, there is a scarcity of data at subject-level, which could potentially explain the limited performance of deep neural networks in this domain. To address this issue, incorporating auxiliary tasks such as self-supervised learning [7] or data augmentation [97] are promising approaches. However, further investigation is still required to explore the efficacy of these methods.

1.3. Evaluation and Reproducibility in BCI

While the literature on EEG-based BCI pipelines is very dense, their evaluation and the interpretation of the results is a major issue for BCI. Indeed, for many studies, it is very difficult, if not impossible to compare the produced results. This stems from the fact that various factors hinder the experimental result comparison: specific preprocessing, cherry-picking datasets, subjects, or classes on a dataset, missing pipelines in the comparison, or lack of statistical analysis.

On the one hand, this issue is compounded

by the need for shared resources and comprehensive evaluations. Thorough evaluations often exceed the scope of a single research work, and it is thus crucial to find rigorous methods to assess the performance of BCI pipelines and ensure the reproducibility of their results. On the other hand, the scientific community has been grappling with a reproducibility crisis across various domains [6], and the field of BCI is not exempt. Addressing this crisis in BCI research is particularly pertinent due to the domain's unique requirements and complexities. BCI studies involve complex methodologies, intricate data acquisition techniques, and multifaceted analysis pipelines, making it challenging to replicate and validate experiments. The transdisciplinary nature of BCI research necessitates specialized knowledge from multiple disciplines, further complicating the ability to reproduce results.

Meta-analysis, a statistical technique for combining the results of multiple studies, is a powerful tool for synthesizing findings and deriving insights from a large body of research [48]. However, due to non-standardized evaluation protocols, conducting meta-analyses in the field of BCI has proven to be challenging. The variability in datasets, experimental tasks, and analysis pipelines inhibits the aggregation of results, despite efforts to establish standardization protocols. Consequently, it is difficult to obtain a comprehensive understanding of the performance of BCIs pipelines across different paradigms through the existing literature.

In response to these challenges, the field has seen the emergence of the Mother of All BCI Benchmarks (MOABB). MOABB [4] was developed as an open-source platform to facilitate the benchmarking and assessment of new datasets and classifiers in major BCI paradigms. The first version of MOABB initiated a community-driven effort to define rigorous and expert methods for conducting proper benchmarking assessments. By providing standardized evaluation procedures, MOABB enables researchers to compare and evaluate the performance of BCI methods in a transparent and reproducible manner. While its adoption by the community is broadening, the systematic evaluation of existing pipelines proposed in [54] is now outdated because it is focused on a specific BCI modality, and new pipelines have emerged in the literature.

1.4. Environmental impact

Addressing climate change requires comprehensive actions across all facets of human societies to adhere to the Paris Climate Agreement. Research, including the field of AI, plays a significant role in achieving this objective. Assessing the environmental impact of machine learning techniques is a challenging task [71], which has been pioneered by the natural language processing community with the prominence of conversational agents [76]. While machine learning models used in BCI are smaller in scale, assessing the performance of various models in link with their environmental impact is critical to promoting virtuous and sustainable models.

The costs associated with deploying computers or computer clusters for training machine learning models are dependent on infrastructure requirements. The environmental impact resulting from energy consumption during model training varies based on geographical criteria, as electricity production methods differ across countries. Several libraries have been developed to provide estimates of this environmental impact, measured in terms of grams of CO2 equivalent emissions [53]. Although these libraries have limitations, they offer a valuable

measurement to enhance our understanding of the training requirements for models. This facilitates a more comprehensive comparison of pipelines within a benchmark setting.

1.5. Contributions

This paper aims to address the lack of reproducibility studies in the field of EEG-based BCI and to go beyond a simple benchmark of existing machine learning pipelines. Indeed, it is important to provide an updated comparison of the previous benchmark [54], including deep learning pipelines. Also, new BCI paradigms for controlling systems could now be included in the evaluation to propose a global take on the current BCI state.

It is an impossible task to evaluate all openly available datasets and all published pipelines in the literature, as there are exotic or unreadable data formats, BCI protocols that are used for a unique dataset, and unavailable code. For this paper, we chose to restrict to three common BCI paradigms, namely MI, ERP, and SSVEP, that are well documented in the literature. The reason for this choice is to ensure reproducible evaluation with enough datasets and subjects to achieve decent statistical meta-analysis.

For the machine learning pipelines, we follow similar guidelines to consider only approaches that could be reimplemented with Python open-source libraries. Indeed, we could only reimplement part of the published pipelines, but we try to include to the best of our effort all pipelines that have been reused in several publications or that are often cited as reference. The MOABB framework is designed to allow seamless integration of novel pipelines for new contributors and facilitate as much as possible the reproducibility of a benchmark to compare a novel pipeline with the one presented here.

We thought of the MOABB open science initiative as a long-term project. The objective is to endow the community with the ability to easily compare results on new datasets or new pipelines, that will be added after the paper publication. A website reproduces the results presented here and will be updated with novel additions. The goal is also to limit the environmental impact of research works by providing a simple means for scientists to copy/paste up-to-date results in their publication to compare with a reference benchmark.

Beyond the purely quantitative benchmark, it is difficult to have a good overview of the open EEG datasets available online. They have been recorded with different hardware, using similar experimental protocols with some variability in experiment design. As data sharing is becoming a common requirement in the scientific community, future works will often result in sharing new open data. It is important to be able to quickly identify existing datasets, and common design choices, to correctly position new experiments with well-informed knowledge. This task is difficult, as open data is scattered in the literature with no common format.

To summarize, in this paper we aim to address the following open questions:

- What is the most effective approach for classifying EEG? How do their computation time and energy consumption compare for training a model?
- What is the best deep learning method? The best Riemannian-BCI pipeline?
- How many trials or channels are required in a dataset to achieve correct performances?
- For MI, which motor imagery tasks give

the best accuracy?

- What are the open datasets in different paradigms and how do they compare?

To investigate those research questions, we conducted many experiments built on open-source tools and with the help of a large community driven by open science guidelines. This paper describes the results obtained from a wide experimental campaign and their in-depth analysis, resulting in the following contributions:

- the largest benchmarking in EEG-based BCI in open science relying on open source libraries
- a fair and replicable evaluation with expert knowledge in BCI
- a deep analysis of the benchmark results, with guidelines for proposing new machine learning pipelines and new datasets or experiments.

2. Benchmark methodology

In this section, we describe the methodology for our benchmark, including which methods are considered, how they are trained and evaluated, and how we analyze the results.

2.1. Analysis pipelines inclusion

A major difficulty in the BCI literature, apart from the data and code availability, is the importance of signal processing in EEG. Many toolboxes are available, in different software platforms, like Matlab, Python, C/C++, Java, C#, Julia, Delphi, and many more. Some toolboxes are sold as commercial products, with undisclosed code or signal processing techniques that are hidden or obfuscated for intellectual property reason. The choices made in those toolboxes, let apart all single projects maintained by only one person, are very different regarding signal filtering or electrode referencing and yield very different outcomes. Despite the fact that complex preprocessing pipelines, and often arguably overcomplicated ones, are detrimental for EEG interpretation and classification [37], most of the toolboxes include them and promote their application in tutorials and guidelines.

In this paper, our analysis pipeline relies on the Python language for its large adoption in the scientific computing, neuroscience and machine learning communities, supported by robust and extensively validated libraries such as numpy [47], scipy [118], MNE [43], scikit-learn [89] and pyriemann [14]. Manipulation of EEG recordings is facilitated through MNE, enabling the extraction of hardware events and the conversion of recordings into numpy arrays. As a light preprocessing, the signal is processed with a 4th-order Butterworth bandpass IIR filter, applied with a forward-backward pass, using standard MNE parametrization. The specific bandpass frequencies are subsequently provided as they depend on the chosen BCI paradigm. Machine learning pipelines are based on scikit-learn and pyriemann estimators.

2.2. Evaluation method

Another difficulty for anyone who wants to reproduce literature results is that reporting classification performances on EEG-based BCI tasks is not standardized. A crucial methodological consideration pertains to the selection of the evaluation metric (whether it be Area Under the Curve (AUC), precision/recall, F1-score, or accuracy), a decision that holds notable significance in the assessment of outcomes. Moreover, in many studies, the authors focus on specific subsets of subjects

within established datasets, or selectively use a restricted part of the cognitive tasks conducted during experiments. These choices make comparing findings across papers becomes inherently cumbersome.

To maintain consistent terminology concerning EEG signals, we will establish the following definitions. A *session* refers to a series of runs where EEG electrodes remain attached to a subject's head, and the overarching experimental parameters remain constant. The term *run* denotes the period during which an experiment is conducted without interruption or pause. Within a session, several runs are performed with potential intervals between them. An *epoch* or a *trial* signifies a segment within a run during which an atomic event occurs, triggered by an external stimulus for event-related potential or steady-state evoked potential or internal volition for motor imagery. These epochs are positioned in time relative to an onset, which corresponds to the start of a stimulus or a task.

In the BCI framework, different evaluation methodologies exist for partitioning the data into training and test datasets, each tailored to address specific challenges. We differentiate between *within-session* as shown in Figure 1, *cross-session*, and *cross-subject* evaluations, respectively illustrated in the annexes in Figure A1 and Figure A2.

In the context of within-session evaluation, the primary concern lies in identifying algorithms that can effectively mitigate overfitting within a single session. In line with common practices in machine learning for cross-validation, all trials from a session are shuffled before being split in k-folds, to evaluate the generalization performance on unseen epochs. Consequently, the pipelines are trained on trials sampled throughout the session duration, which helps mitigate the impact of intra-session variability in an individual's EEG. While this allows a more statistically accurate benchmark of a pipeline, it provides an upper bound for the evaluation metrics when compared to online evaluation. Despite the difference with experimental applications of BCI system, this training methodology is commonly employed in the existing literature [13, 111, 84, 75], influencing our decision to adopt it for this reproducibility study.

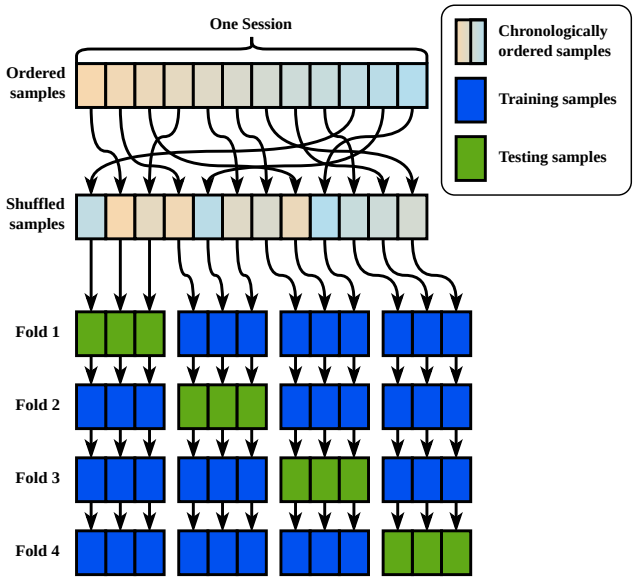


Figure 1: Within-session evaluation, small rectangles indicate a sample or EEG trial, pastel colors on the two top lines shows the chronological order, bright color on the last three lines indicates training and testing samples/trials.

In contrast, the cross-session (resp. cross-subject) evaluation employs a leave-one-out cross-validation technique, where only one session (resp. subject) is designated as the testing dataset. The results from cross-session or cross-subject evaluation methods put more emphasis on transfer learning to cope with subjects' variability. However, the questions raised by transfer learning in a BCI context [119, 121] are manyfold – dealing with

subject alignment, training models for each subject or a single one, at a session, subject or dataset level – and are outside the scope of this article. For this reason, this paper focuses on the most common evaluation strategy in the literature, which is within-session evaluation.

Nonetheless, one should acknowledge that within-session evaluation has its limitations. It addresses only partly the complications stemming from variations in sessions and subjects. Such variations may arise from internal factors, like minor electrode displacements between sessions, different calibrations of EEG hardware, or external factors, like the dynamic nature of EEG measurements in an individual based on the cognitive states [98], such as alertness, drowsiness or fatigue.

Within-session evaluation of a pipeline is conducted for each subject and each session by splitting the epochs into training and test epoch sets using five stratified K-fold splits. In each fold, the testing set contains 20% of the session epochs while maintaining the class distribution. The final evaluation score corresponds to the average over the five splits. Depending on whether the classification problem is binary or involves more than two classes, evaluation scores correspond to the Area Under the Receiver Operating Characteristic Curve (ROC) or classification accuracy, respectively.

2.3. Grid search

Another important aspect in BCIs is how the hyperparameters are selected. They can significantly affect the performance of the algorithm and, consequently, the accuracy of the system's predictions. It is therefore crucial to employ a method that facilitates the search for optimal parameter values. Grid search stands out as a popular method for hyperparameter tuning in machine learning algorithms. In particular, it is essential to search for the correct hyper-parameters for each scenario, considering variations in evaluation procedures, datasets, subjects, and sessions.

Using nested cross-validation for hyperparameters selection, we mitigate the risk of overfitting, as recommended in existing literature [25]. This approach is implemented by using an inner 3-fold cross-validation. Additionally, we have devised tailored grid search functions for each evaluation procedure, as detailed in subsection Appendix A.1. This standardized framework streamlines hyperparameter tuning, enabling seamless performance comparison across diverse machine-learning models and promoting experiment reproducibility across varied datasets.

2.4. Statistical analysis

The statistical comparison of two different pipelines can be conducted either at a *dataset-wise* level or across *multiple datasets*.

The *dataset-wise* comparison involves assessing the statistical differences between two pipelines within the same dataset. This is done using effect sizes and p-values. To optimize computational efficiency, the number of subjects N in the dataset determines the method of estimating p-values. For datasets with $N < 13$, one-tailed permutation-based paired t-tests are used with all possible permutations. For datasets with $13 \leqslant N \leqslant 20$, 10000 random permutations are employed. For datasets with $N > 20$, the Wilcoxon signed-rank test [122] is used. The effect size between two pipelines is measured via Standardized Mean Difference (SMD) [48] estimated over the subjects.

For *multiple datasets*, the comparison of

statistical differences between two pipelines over D is done by combining effect sizes $\{s_i\}_{i=1}^{D}$ and p-values $\{p_i\}_{i=1}^{D}$ with Stouffer's Z-score method [96] that take in account the different sizes of the datasets. Combined p-values are obtained by estimating $Z = \sum_{i=1}^{D} w_i Z_i$, where $Z_i = \Phi^{-1}(1 - p_i)$ with Φ the standard normal cumulative distribution function. This weighted Z-score relies on weights $w_i = \sqrt{\frac{N_i}{\sum_{i=1}^{D} w_i^2}}$ that are proportional to N_i the number of subjects in the dataset i. Combined measures of the effect sizes are obtained by $S = \sum_{i=1}^{D} w_i s_i$ weighted average of SMD.

2.5. Code Carbon

The assessment of the environmental impact of research is gathering momentum, yet the methodology remains a topic of heavy debate, particularly in computer science and machine learning [71].

Within these fields, a significant environmental impact stems from the production processes of devices (such as acquisition devices, computers, GPUs, and clusters) and the energy consumption during their usage. Evaluating the impact of device production poses challenges, given the limited information shared by manufacturers, potential inaccuracies in estimations, and the frequent sharing and reusing of research equipment across multiple projects. This issue is notably prevalent for CPU and GPU clusters, where numerous experiments run in parallel [76].

Energy consumption during usage is commonly acknowledged as a key indicator of the environmental impact, although it represents only a fraction of the overall impact. This measurement is typically expressed as grams of CO2 equivalent (gCO2) emissions released into the atmosphere and heavily relies on factors such as the power grid setup and energy production localization. National power companies typically offer estimates of the carbon footprint of consumed energy, articulated in gCO2 per Watt-hour. This value can significantly vary between countries based on energy production sources; for instance, countries reliant on coal or fuel power plants tend to have a higher carbon footprint compared to those utilizing hydroelectric power or solar panels.

For optimal measurement of energy consumption and carbon footprint, power meters are ideal, albeit encompassing the entire computer's energy consumption. To gain more precise insights, especially for evaluating specific programs or machine learning pipelines, software-based power meters prove useful [53]. Various tools exist for this purpose, many of which are open source. In our study, we opted to employ Code Carbon [35], a Python-based tool that seamlessly integrates into experiments conducted on individual computers or clusters.

When estimating the carbon footprint of a machine learning pipeline, it is critical to analyze both the training and inference phases. During k-fold validation, inference constitutes a substantial portion of energy consumption. Consequently, the estimated carbon footprint offers valuable insights into the trade-offs between different algorithms, with considerations for CPU consumption, execution time, and the parallelization of algorithms. Notably, well-parallelized algorithms can achieve lower execution times when distributed across a large number of CPUs. The carbon footprint serves as an insightful metric for assessing the computational complexity of a pipeline, particularly when considering its adaptability to embedded or dedicated architectures, common in commercial neurotechnology products.

3. Datasets

There are different types of BCI applications, referred to as *paradigms* throughout this article, each relying on distinctive neurological patterns to facilitate communication between the brain and the BCI system. Depending on the selected paradigm, the raw EEG recordings are transformed into trials for machine learning pipelines. These transformations include bandpass filtering, signal cropping based on stimulus events, and potential resampling to adjust the sampling frequency if needed by the machine learning pipelines. Employing standardized preprocessing procedures enables a fair comparison among different algorithms.

3.1. Motor Imagery

The MI paradigm involves a cognitive process where an individual mentally simulates the execution of a motor action without actually performing it. This paradigm is widely used in neuroscience to delve into the neural mechanisms governing motor control and learning. Moreover, it finds applications in neurorehabilitation to assist in restoring motor functions for individuals with neurological disorders or injuries. Various tasks are associated with this paradigm in Mother Of All BCI Benchmark (BCI), with common examples including left-hand, right-hand, and feet imagery. The choice of evaluation metrics for classification performance varies based on the number of tasks involved in the classification process: ROC-AUC is used for two-task classifications, whereas accuracy metrics are employed for multiclass scenarios. With the MI paradigm, signal processing includes bandpass filtering within the $[8 - 32]$ Hz frequency range [86].

The Table 1 provides a comprehensive overview of MI datasets, with class name abbreviations including RH (Right Hand), LH (Left Hand), F (Feet), H (Hands), T (Tongue), R (Resting State), LHRF (Left Hand Right Foot), and RHLF (Right Hand Left Foot). The column 'No. trials' denotes the number of trials per class, session, and run. For instance, BNCI2014_001 comprises 12 trials per class across 4 classes, 6 runs, and 2 sessions, resulting in a total of $12 \times 4 \times 6 \times 2 = 576$ trials in the dataset. The only exception is for the PhysionetMI dataset, where in the first 3 runs, there are RH, LH, and R events; in the last 3 runs, there are H, F, and R events.

3.2. P300/ERP

The P300 paradigm, which is a specific ERP paradigm, serves as a framework for categorizing psychophysical experiments [77]. It provides a visual representation of a distinct component characterized by a prominent positive deflection occurring approximately 300 ms after stimulus onset. Typically, this component is elicited in cognitive tasks involving unpredictable and infrequent changes in stimuli. This is improperly called P300 ERP in the BCI community (see P300 BCI or P300 speller), whereas the cognitive components are more diverse than just the P300 wave. See [77] for a detailed discussion about this point.

In the present study, our primary focus is on P3b, a specific subtype that examines stimulus changes relevant to the task at hand. This particular brain wave response manifests when the cognitive tasks involve stimuli that are predictable to some extent but are still imbued with an element of unpredictability. As is commonly the case in the literature, we classify events as targets or non-targets, *i.e.*, at the epoch level, resulting in a binary classification task. We evaluate

Table 1: Overview of the Motor Imagery EEG datasets available in MOABB.

Motor imagery									
Dataset	No. subj.	No. ch.	No. classes	No. trials /session/class	Trial len.(s)	S.freq. (Hz)	No. sessions	No. runs	Classes
AlexMI [8]	8	16	2(3)	20 ± 0	3	512	1	1	RH, F, (R)
BNCI2014_001 [111]	9	22	3(4)	72 ± 0	4	250	2	6	RH, LH, F, (T)
BNCI2014_002 [109]	14	15	2	80 ± 0	5	512	1	8	RH, F
BNCI2014_004 [70]	9	3	2	72.4 ± 9.5	4.5	250	5	1	RH, LH
BNCI2015_001 [38]	12	13	2	100 ± 0	5	512	3 subj. 8-11 2 others	1	RH, F
BNCI2015_004 [102]	9	30	2(5)	39.4 ± 1.6	7	256	2	1	RH, F
Cho2017 [30]	52	64	2	101.2 ± 4.7	3	512	1	1	RH, LH
Lee2019_MI [69]	54	62	2	50	4	1000	2	1	RH, LH
GrosseWentrup2009 [44]	10	128	2	150 ± 0	7	500	1	1	RH, LH
PhysionetMI [101]	109	64	4(5)	22.6 ± 1.3	3	160	1	6***	RH, LH, H, F, (R)
Schirrmeister2017 [103]	14	128	3(4)	240.8 ± 37.7	4	500	1	2	RH, LH, F, (R)
Shin2017A [105]	29	30	2	10 ± 0	10	200	3	1	RH, LH
Weibo2014 [125]	10	60	4(7)	79 ± 3	4	200	1	1	RH, LH, H, F, (LHRF), (RHLF), (R)
Zhou2016 [128]	4	14	3	50 ± 3.5	5	250	3	2	RH, LF, F

its performance with the ROC-AUC metric, which handles the inherent imbalance in the problem at hand. With the ERP paradigm, the signal is bandpass filtered to the 1-24 Hz frequency band [77].

The Table 2 shows an overview of all the ERP datasets. The column "No. epochs NT/T" indicates the number of *NonTarget* and *Target* epochs per session and run.

3.3. SSVEP

Steady State Visually Evoked Potentials are generated when presenting repetitive sensory stimuli to the subject. While tactile and auditory stimulations are seldom used, visual stimulation is very common, both for control [28] or cognitive probes [123]. The frequency of the stimulus repetition induces a brain oscillation in the associated sensory area. The amplitude of the generated oscillation follows the 1/f law, meaning that stimulation in low frequency (5-7 Hz) induces responses of higher amplitude than higher frequency (20-25 Hz). Stimulation above 40 Hz could be difficult to detect due to the weak generated oscillations. In BCI applications, SSVEPs have been used for building spellers [83] and for button-pressing [56], but those applications are limited by the number of available frequencies of stimulation. It is possible, using systems with precisely synchronized stimulation and recording, to encode information both in frequency and phase, therefore multiplying the choices possible [85]. Similarly to the MI paradigm, we evaluate the classifiers using the ROC-AUC metric if only two classes are used and the accuracy metric if there are more. With the SSVEP paradigm, the signal is bandpass filtered to the 7-45 Hz frequency band [29]. The Table 3 encompasses all the SSVEP datasets considered in this study.

Table 2: Overview of the ERP EEG datasets available in MOABB.

P300 / ERP								
Dataset	No. subj.	No. ch.	No. epochs NT/T /session	Epoch len.(s)	S.freq. (Hz)	No. sessions	No. runs	Keyboard
BI2012 [116]	25	16	638.2 ± 1.9/127.6 ± 0.7	1	128	1	1	36 aliens
BI2013a [115, 10, 32]	24	16	400.3 ± 2.3/80.1 ± 0.5	1	512	8 subj. 1-7 1 subj. 8-24	1	36 aliens
BI2014a [64]	64	16	794.5 ± 276.7/158.9 ± 55.3	1	512	1	1	36 aliens
BI2014b [65]	37	32	201.3 ± 61.5 /40.3 ± 12.3	1	512	1	1	36 aliens
BI2015a [62]	43	32	461.8 ± 220.9/92.3 ± 44.1	1	512	3	1	36 aliens
BI2015b [63]	44	32	2158.7 ± 6.3/479.9 ± 0.3	1	512	1	4	36 aliens
BNCI2014_008 [94, 39]	8	8	3500 ± 0/700 ± 0	1	256	1	1	36 char.
BNCI2014_009 [3]	10	16	480 ± 0/96 ± 0	0.8	256	3	1	36 char.
BNCI2015_003 [46]	10	8	2250 ± 1500 /270 ± 60	0.8	256	1	2	36 char.
EPFLP300 [49]	8	32	685.2 ± 16.9/137.2 ± 3.5	1	2048	4	6	6 images
Huebner2017 [50]	13	31	3275.3 ± 2.1 /1007.8 ± 0.6	0.9	1000	2 subj. 6 3 others	9	42 char.
Huebner2018 [51]	12	31	3638.4 ± 7.7 /1119.6 ± 2.5	0.9	1000	3	10	42 char.
Lee2019_ERP [69]	54	62	3450/690	1	1000	2	1	36 char.
Sosulski2019 [106, 108]	13	31	75 ± 0 /15 ± 0	1.2	1000	4 subj. 1 3 others	20	2 tones
Cattan2019_VR [24]	21	16	600 ± 0/120 ± 0	1	512	2	60	36 crosses

Table 3: Overview of the SSVEP EEG datasets available in MOABB.

SSVEP									
Dataset	No. subj.	No. channels	No. classes	No. trials /session/class	Trial len.(s)	S.freq. (Hz)	No. sess.	No. runs	Classes
Lee2019_SSVEP [69]	54	62	4	25	1	1000	2	1	4 (5.45-12)
MAMEM1 [87]	10	256	5	16.8 ± 3.5 classes 8.57,10.0 21.0 ± 4.4 classes 6.66,7.5,12.0	3	250	1	3 subj. 1,3,8; 4 subj. 4,6 5 others	5 (6.66-12.00)
MAMEM2 [87]	10	256	5	20 class 12.0; 30 class 8.57 25 others	3	250	1	5	5 (6.66-12.00)
MAMEM3 [87]	10	14	4	20.0 ± 0.0 class 6.66; 25.0 ± 0.0 class 8.57 30.0 ± 0.0 class 10.0; 25.0 ± 0.0 class 12.0	3	128	1	10	4 (6.66-12.00)
Nakanishi2015 [84]	9	8	12	15.0 ± 0.0	4.15	256	1	1	12 (9.25-14.75)
Kalunga2016 [56]	12	8	4	20.0 ± 7.7	2	256	1	5 subj. 12; 4 subj.10 3 subj. 7; 2 others	4 (13,17,21,rest)
Wang2016 [120]	34	62	40	6.0 ± 0.0	5	250	1	1	40 (8-15.8)

The Figure 2 displays an embedding of all the datasets in 2 dimensions, using UMAP for dimensionality reduction on all feature information regarding the datasets, as listed in Tables 1, 2 and 3. The color indicates the paradigms (blue for Motor Imagery, green for Event Related Potential, and red for Steady State Visually Evoked Potential), and the name of each dataset is written on top of the circle. The color intensity is related to the number of electrodes, datasets with a low number of electrodes are in lighter color, and the size of the circles is proportional to the number of subjects. The UMAP embedding preserves the local topology. This highlights that, despite different paradigms, datasets with many electrodes and many subjects are in a central position. Datasets with fewer subjects and fewer electrodes are closer to the border of the figure. The

BNCI2014_001 dataset, commonly called BCI Competition IV dataset 2a, is the most widely used in BCI literature. There are closely related datasets with roughly the same number of subjects, electrodes, and trials (BNCI2015_004, BNCI2015_001), with more subjects (Shin2017A) or with fewer subjects (Zhou2016). This group of datasets is useful for the fast evaluation of new pipelines and to see how well results generalize with the same kind of dataset. It is also possible to ensure a good coverage of the dataset features, using only a few datasets. With MI, a selection of BNCI2014_001, BNCI2014_004, Schirrmeister2017, and PhysioNetMI might be sufficient to evaluate an approach on datasets with very different configurations.

4. Pipelines

As discussed in section 1, most classification algorithms in BCI research for EEG signals fall into one of three main categories: those based on raw signals (referred to as "Raw" hereafter), algorithms relying on covariance matrices seen as elements of a Riemannian manifold (denoted "Riemannian") and the Deep Learning (DL) approaches.

The Raw signal methods typically employ supervised spatial filters to simultaneously enhance the component related to the cognitive task while reducing the dimensionality of the EEG data. In contrast, Riemannian pipelines consider the signal through its estimated covariance matrices, leveraging the natural metric acting on the curved geometry of SPD matrices, which remains invariant by congruence transformations [124, 33]. Those approaches are thus mostly invariant to any spatial transformations applied on the signal, making them highly effective in BCI applications.

Lastly, deep neural networks learn spatial and temporal filters directly from raw EEG data. Although there is a wealth of literature on deep learning models, there are few models available or evaluated with reproducible frameworks. Initiatives like Braindecode [103] or, to a lesser extent for BCI, torchEEG [127], offer an open-source implementation of the most efficient models. This study considers a set of raw, Riemannian, and deep learning models, which are detailed in Table 4, along with the hyperparameters used for the grid-search approach, listed in the annexes within Table A1.

4.1. Raw signal

The category of pipelines referred to as *Raw* consists of BCI classifiers employing traditional statistical analysis, as well as temporal and/or spatial filtering tools to extract features. Variance-based pipelines represent one of the initial BCI pipeline concepts proposed for motor imagery. Several approaches have highlighted the value of utilizing intra-channel variance for online decoding tasks. These pipelines calculate the variance of each electrode within an epoch to create a positive definite real-valued vector. To address noise or artifacts, it is common practice to logarithmically transform the observed variance [74]. This approach essentially boils down to only considering the diagonal elements of the covariance matrices. Classifiers such as Linear Discriminant Analysis (LDA) or Support Vector Machine (SVM) can be utilized on the resulting vector to predict the label of the epoch.

Common Spatial Pattern (CSP) approach learns spatial filtering matrices in a supervised manner, minimizing the variance of the band power feature vectors within the same

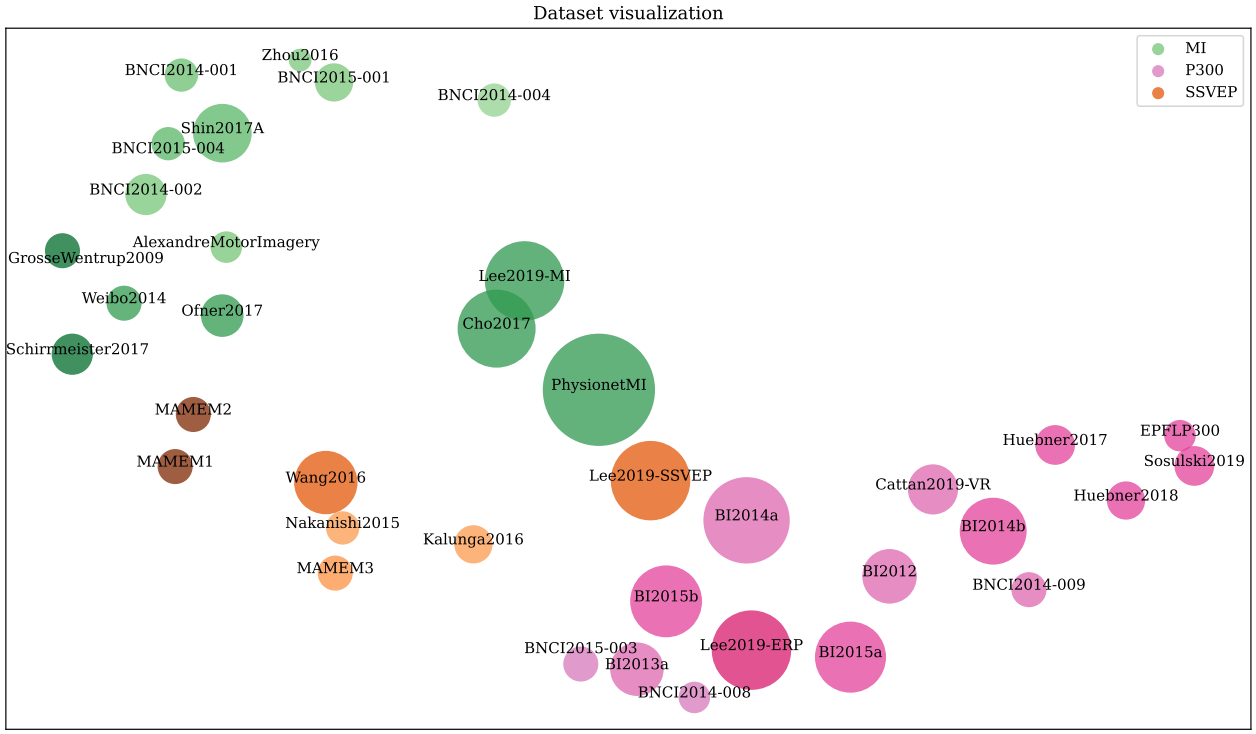


Figure 2: Visualization of the MOABB datasets, with Motor Imagery in green, Event Related Potential in pink/purple and Steady State Visually Evoked Potential in yellow/brown. The size of the circle is proportional to the number of subjects and the contrast depends on the number of electrodes.

Table 4: Pipelines considered in this study, the color indicates the paradigm. Green is for motor imagery, pink for P300 and orange for SSVEP.

Pipeline Name	**Category**	**References**	**Pipeline Name**	**Category**	**References**
LogVar + LDA	Raw		TS + LR	Riemannian	[12]
LogVar + SVM	Raw		TS + SVM	Riemannian	[12]
CSP + LDA	Raw	[61, 19]	ACM + TS + SVM	Riemannian	[22]
CSP + SVM	Raw	[61, 19]	ShallowConvNet	Deep Learning	[103]
TRCSP + LDA	Raw	[73]	DeepConvNet	Deep Learning	[103]
DLCSPauto + shLDA	Raw	[61, 19],	EEGNet 8 2	Deep Learning	[67]
FBCSP+SVM	Raw	[2]	EEGTCNet	Deep Learning	[52]
FgMDM	Riemannian	[12]	EEGITNet	Deep Learning	[100]
MDM	Riemannian	[13]	EEGNeX 8 32	Deep Learning	[27]
TS + EL	Riemannian	[34]			
XDAWN + LDA	Raw	[95]	XDAWNCov + TS + SVM	Riemannian	[28]
XDAWNCov + MDM	Riemannian	[9]	ERPCov + MDM	Riemannian	[10]
ERPCov($svd_n = 4$) + MDM	Riemannian	[10]			
TRCA	Raw	[85]	SSVEP MDM	Riemannian	[29]
CCA	Raw	[72]	SSVEP TS + LR	Riemannian	[29]
MsetCCA	Raw	[126]	SSVEP TS + SVM	Riemannian	[29]

class, while maximizing the between-class variance [81, 19]. To enhance the robustness of the CSP against noise and overfitting, Tikhonov Regularized CSP (CSP) approach has been proposed in [73]. In contrast to the classical CSP, which uses band-pass filtered EEG signals that may vary per subject, Filter-Bank CSP (CSP) addresses this issue by extracting CSP features for each band-pass filter from the filter bank [2]. The subsection Appendix B.1 details the CSP algorithm.

Canonical Correlation Analysis (CCA) has emerged as a prominent approach for classifying SSVEP signals, initially introduced in the work by [72]. Subsequently, CCA has been successfully employed in numerous notable SSVEP-based BCI studies, such as those conducted by [18] and [83]. SSVEP signals exhibit correlation to flickering visual stimuli, with their signal phase and frequency corresponding to stimulus characteristics. Leveraging this relationship, CCA aids in extracting EEG spatial components with the strongest correlation to SSVEP stimuli. Further details on the different pipelines can be found in subsection Appendix B.2.

4.2. Riemannian geometry

The introduction of Riemannian geometry into BCI processing marked a pivotal moment for the BCI community [13]. The fundamental concept underlying this approach is to represent the signal using covariance matrices or their derivatives. As covariance matrices are Symmetric Positive Definite (SPD) matrices, they live in a Riemannian space [124]. We provide here a general description of the framework needed to define algorithms based on the Riemann distance for classification tasks. For a more in-depth description of the Riemannian framework, we refer the reader to [20] which gives a pedagogical introduction to these concepts.

Due to the curvature of the SPD matrix space, traditional Euclidean geometry is ill-suited and introduces a swelling effect. Particularly, the Euclidean distance could result in a wrong characterization of the relationship between SPD matrices. Instead, Riemannian methods rely on a distance that respects the geometry of the SPD matrices space, based on geodesics, i.e., the shortest path that connects 2 elements and stays in the space of SPD matrices. A common choice of distance is the affine-invariant one [79]. Considering the manifold of SPD matrices $\mathcal{M}_n = \{\mathbf{P} \in \mathbb{R}^{n\times n} | \mathbf{P} = \mathbf{P}^\top \text{and} x^\top \mathbf{P} x > 0, \forall x \in \mathbb{R}^n\}$, the affine-invariant distance for $\mathbf{P}_1, \mathbf{P}_2 \in \mathcal{M}$ is defined as

$$\delta_R\left(\mathbf{P}_1, \mathbf{P}_2\right) = \left[\sum_{i=1}^{n} \log^2 \lambda_i \left(\mathbf{P}_1^{-1}\mathbf{P}_2\right)\right]^{1/2} \quad (1)$$

where $\lambda_i(\mathbf{P})$ is the i-th eigenvalues of $\mathbf{P}$.

Similarly, the concept of the mean in Riemannian geometry must be redefined to ensure it belongs to the manifold; it is known as the Frechet mean [79] and is defined as:

$$\hat{\mathbf{G}} = \underset{\mathbf{P}\in P(n)}{\operatorname{argmin}} \sum_{i=1}^{m} \delta_R^2\left(\mathbf{P}, \mathbf{P}_i\right) \quad (2)$$

As we are operating within a Riemannian manifold, traditional machine-learning classification algorithms cannot be directly applied. Instead, there are two options: either create new algorithms to classify SPD matrices on the Riemannian manifold or map the matrices to an associated Euclidean space and then apply standard classification algorithms. In the former scenario, algorithms like Minimum Distance to Mean give robust accuracy. In the latter scenario, it is possible to rely on the tangent

space to a point of the manifold. It is possible to circulate between the manifold and the tangent space using the Log and Exp map functions. The Log (resp. Exp) maps the manifold to the tangent space (resp. the tangent space to the manifold):

$$\mathrm{Exp}_{\mathbf{P}}\left(\mathbf{S}_i\right) = \mathbf{P}^{1/2}\,\mathrm{Exp}\left(\mathbf{P}^{-1/2}\mathbf{S}_i\mathbf{P}^{-1/2}\right)\mathbf{P}^{1/2} \tag{3}$$

$$\mathrm{Log}_{\mathbf{P}}\left(\mathbf{P}_i\right) = \mathbf{P}^{1/2}\,\mathrm{Log}\left(\mathbf{P}^{-1/2}\mathbf{P}_i\mathbf{P}^{-1/2}\right)\mathbf{P}^{1/2} \tag{4}$$

Projected in the tangent space, any machine learning algorithm could be applied. One limitation is the size of the considered space, that is $\frac{n(n+1)}{2}$ for $\mathcal{M}_n$. Machine learning algorithms like Support Vector Machine (SVM), ElasticNet (EL) or Logistic Regression (LR) are among the most popular for classification in the tangent space. Details regarding the pipelines implementation are available in subsection Appendix B.3.

4.3. Deep learning

DL methods have demonstrated in recent years considerable promise in various tasks that involve handling massive volumes of digital data and those in different fields such as computer vision [66] and Natural Language Processing [117, 104].

This also holds for BCI applications. The BCI field has been significantly impacted by the integration of DL techniques [99, 36], which exhibit strong generalization capabilities, with transfer learning emerging as a key focus in BCI research. One notable advantage of DL models is their capacity to leverage vast datasets, a task typically challenging for classical Machine Learning (ML) algorithms. Moreover, by conducting all processing and classification steps within a neural network, DL models enable optimized end-to-end learning. In this paper, we will focus specifically on DL methods that have been applied to MI paradigms.

The predominant Python libraries for DL are Tensorflow [1] and PyTorch [88]. To facilitate broader access to MOABB, we developed integration for both libraries using wrappers from Scikeras [42] (for Tensorflow) and Skorch [114] (for PyTorch). We were also keen on integrating the Braindecode [103] library, which incorporates several DL algorithms for EEG processing using PyTorch.

Most DL architectures for EEG decoding operate on minimally pre-processed (bandpass filters) or raw epochs. To address the spatio-temporal complexity of EEG signals, convolutional layers are commonly employed with separable 1D convolution along the temporal and spatial dimension (i.e., EEG channels). By segregating the spatial and temporal convolutions, these models account for the fact that all EEG channels observe all brain sources.

Although the goal is to minimize preprocessing steps before passing data to DL pipelines, a few steps are usually necessary. Bandpass filtering is applied to ensure a fair comparison between ML and DL methods. Neural network architectures reviewed here are designed for decoding EEG signals at a certain sampling frequency. Resampling datasets to match DL architectures' expected frequencies is performed to avoid interfering with the relative temporal length of their kernels. Standard rescaling of the signal is implemented before DL pipelines in adherence to common deep learning practices [68].

Data augmentation, a technique generating synthetic training samples through transformations, is well-established in computer vision for producing state-of-the-art results [26]. However, in the realm of EEG applications,

data augmentation poses unique challenges. Despite its potential to reduce overfitting and enable complex algorithms, it is still an active area of research. Assessing whether a generated signal accurately captures the physiological attributes of EEG remains an open question. Thus, this study abstained from employing data augmentation procedures in the context of EEG decoding. For a comprehensive examination of various existing techniques for EEG, we refer the reader to [97].

Consistent parameters were used across all DL experiments, training networks for 300 epochs with a batch size of 64, Adam optimizer [60] with a base learning rate of 0.001 and cross-entropy as the loss function. An early stopping strategy with a patience parameter of 75 epochs was crucial to prevent overfitting. While the considered DL architecture hyperparameters align with state-of-the-art standards through signal resampling, further optimization could enhance model performance via a grid-search approach tailored to individual scenarios.

5. Experimental results

The experimental results outlined in this section encapsulate the key insights gathered during this benchmark study. While we provide only the most important findings to the reader in this section, we also report all the raw results to ensure proper reproducibility. Due to space limitations, all the detailed evaluations, including ROC-AUC or accuracy scores, are available in the appendix. Tables D1, D2, D3, D4 and D5 display the average scores of each pipeline across all subjects and sessions within a specific dataset. Additionally, Figures C1, C2, C3, C4 and C5 present the pipeline groups' scores for each subject and session across all datasets to aid in result interpretation.

5.1. Riemannan pipelines outperforms others pipelines

The Riemannian distance-based classification pipeline consistently outperforms results obtained through DL and Raw pipelines across all datasets, on all paradigms. Figure 3 illustrates this superiority in the context of right-hand vs left-hand classification, SSVEP and P300. The ROC-AUC and accuracy results are shown based on each pipeline's performance relative to its category (Raw, Riemannian, DL). It should be emphasized that each dot on the plots represents the average of all pipelines from a category on a dataset, each dataset encompassing 10 to 100 subjects. In the distributions, each dataset has the same weight, regardless of the number of subjects or sessions. The results presented here are thus summarizing the largest BCI study to date.

The plots demonstrate the dominance of the Riemannian approaches, showcasing its superior performance not only in overall averages but also consistently across all datasets examined. This trend is further reinforced by the noticeable peak shift in the distribution of results, providing additional confirmation of the effectiveness of the Riemannian approaches. Similar findings are observed across various classification tasks within the Motor Imagery paradigm, as could be seen in the appendix on Figures C1 and C2.

The suboptimal performance of DL pipelines can be attributed to two main factors. Firstly, the hyperparameters of DL pipelines were not fine-tuned for each dataset; instead, the parameters described in the original articles were employed. This marks a significant divergence from the Riemannian and Raw approaches, which had their hyperpa-

rameters optimized through a nested cross-validation strategy. We could not complete a similar hyperparameter search for DL as the search space is too large and it involves complex changes in the architecture shape, such as the kernel size, or the activation functions that could not be automatized easily. Secondly, we chose not to include any data augmentation steps in our research, as explained in Sect. 4.3. It is noteworthy that such procedures have been demonstrated to exert a substantial impact on performance, as evidenced by previous studies [97]. Still, it required a lengthy and dataset-specific parametrization. It is important to recall that the objective of this study is to assess existing and published algorithms, to evaluate their off-the-shelve performances, and not to investigate how to properly tune them.

5.2. Riemannian pipelines work well with limited number of electrodes

Riemannian pipelines perform best in scenarios involving a reduced number of channels for MI tasks. Employing a limited number of electrodes in EEG recordings offers numerous advantages for practical BCI experiments. This approach simplifies the setup, reducing both complexity and cost. Additionally, it enhances user comfort, providing greater flexibility and ease of use in diverse settings, including home-based or mobile applications. However, using fewer electrodes may compromise signal quality and spatial resolution, potentially resulting in lower classification accuracy and reduced robustness against artifacts and noise.

To address these challenges, it becomes crucial to design classification algorithms capable of delivering high performance with a limited number of electrodes. As depicted in Figure 4-(a), the Riemannian pipelines excel in performance with datasets containing $[0, 25]$ electrodes. Notably, their performances tend to decrease as the number of channels increases. Conversely, DL pipelines require a substantial amount of information to achieve satisfactory performance, while facing the limitations described earlier, emphasizing the importance of balancing electrode count and classification effectiveness.

As observed, there is a decline in performance in settings with a moderate number of channels. This decrease in effectiveness can be attributed to several factors. Primarily, an increased number of electrodes elevates the problem's complexity, as the ML algorithm needs to extract relevant information more efficiently. Furthermore, this category encompasses datasets with varying average scores, impacting overall performance. Conversely, scenarios with a high number of electrodes exhibit less pronounced detrimental effects. This could be a side effect; a large number of electrodes implies higher-grade equipment and specialized technicians. The enhanced data recording procedures and superior data quality associated with a larger number of channels might thus alleviate previous issues.

Classification algorithms based on Riemannian distances can be implemented in two distinct approaches. The first method involves performing classification directly on the Riemannian manifold, using algorithms like the MDM (Minimum Distance to Mean) algorithm. The alternative approach involves projecting data onto the tangent space and conducting classification using conventional ML algorithms (SVM, LR, EL). Comparing the two strategies, it is observed that the Riemannian method based on Tangent space projection consistently outperforms the approach centered on the Riemannian surface as shown in Figure 4-(b).

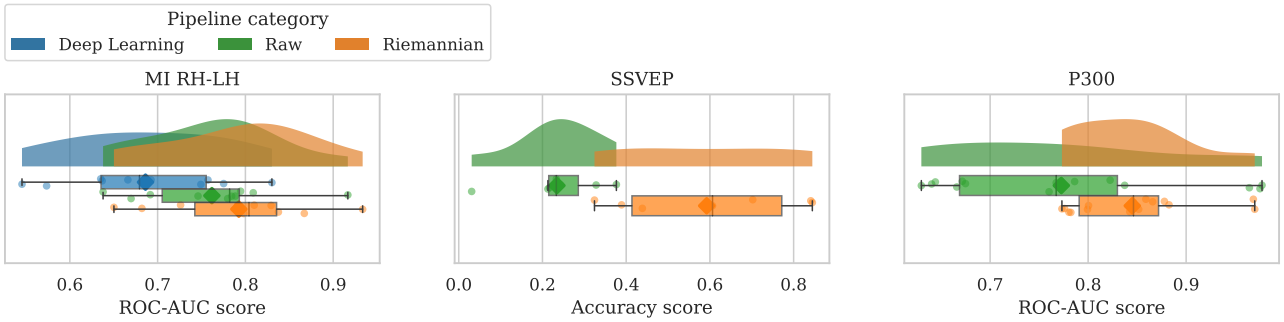


Figure 3: Average performance of pipelines grouped by category (Deep Learning, Riemannian, and Raw) across the MI (right-hand vs left-hand), SSVEP, and ERP paradigms displayed as raincloud plots. Each point in the plot corresponds to the average score of one dataset across all pipelines within a specific category, encompassing all subjects and sessions.

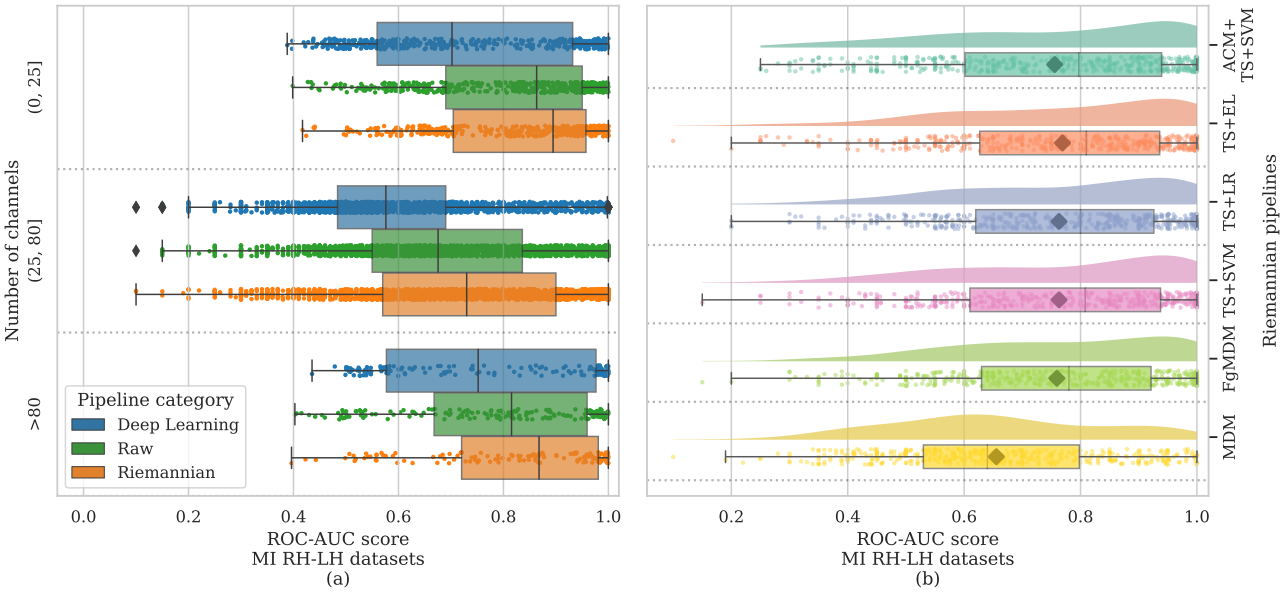


Figure 4: (a) AUC scores are averaged across all sessions, subjects, and datasets within the right-hand vs left-hand MI paradigm for each category (Deep Learning, Riemannian, Raw), segmented by the number of channels on the y-axis. Box plots overlaid with strip plots show individual ROC-AUC scores. (b) Distribution of AUC scores for the Riemannian MI pipelines is depicted for the right-hand vs left-hand classification task. The boxes and horizontal black bars denote quartile ranges.

5.3. Deep learning requires a high number of trials

It is essential to emphasize that the lower performance of DL pipelines, in comparison to the other pipeline categories, is closely linked to the specific DL architecture under consideration. This observation is illustrated in Figure 5-(a). Amongst DL pipelines, ShallowConvNet stands out with the highest AUC. Interestingly, a distinctive dichotomy appears within DL models. The first group, consisting of ShallowConvNet, EEGNet-8.2, and DeepConvNet, exhibits superior performance. In contrast, the second group, gathering EEGTCNet, EEGITNet, and EEGNeX, shows lower performance levels. This divergence in performance is likely attributed to the optimization of hyperparameters in the models. It demonstrates that DL models that have been more extensively tested, and hence correctly parametrized, yield higher classification performance on all datasets.

The number of trials employed in training algorithms carries particular importance, specifically in the context of DL pipelines. A clear trend emerges when evaluating DL algorithm performance, indicating that achieving satisfactory results typically requires more than 150 trials per class, as shown in Figure 5-(b). Again we observe the same two distinct groups described above, the group comprising ShallowConvNet, EEGNet-8.2, and DeepConvNet exhibits a higher resilience to the impact of the number of trials. Notably, this group manages to achieve already satisfactory performance with as few as 50 trials per class, showcasing a relatively robust response to variations in the training dataset size, highlighting the distinct capabilities of certain architectures to yield superior performance with a more limited number of trials.

5.4. Recommended number of trials depends on the task complexity

In scenarios where tasks exhibit a clear distinguishability, such as the MI task involving right-hand/feet movements, achieving impressive AUC performance with a limited number of trials is feasible, as clearly depicted in Figure 6a. Notably, both the Raw and Riemannian pipelines demonstrate exceptional classification scores even with fewer than 50 trials. Increasing the trial count beyond 50 does not yield significant improvements in AUC results for these pipelines. On the other hand, for DL pipelines to reach good AUC, datasets associated with more than 150 trials are imperative to achieve satisfactory AUC scores, highlighting the pivotal role trial quantity plays in DL model performance. This discrepancy highlights the varying requirements and efficiencies of different pipeline approaches based on the task complexity and nature of the data.

When dealing with tasks of higher complexity, such as MI paradigms involving 3 to 7 distinct classes (refer to Section 3.1 for detailed explanation), obtaining optimal accuracy becomes considerably more challenging. In such intricate tasks, a larger number of trials is imperative for pipelines to achieve the highest levels of accuracy. This phenomenon is illustrated in Figure 6b, where the accuracy variability across different datasets becomes apparent. The fluctuation in accuracy levels observed here is intricately tied to the unique characteristics and quality of individual datasets, portraying the nuanced dynamic between dataset quality, trial quantity, and the performance of the classification pipelines. Specifically, the noticeable decline in accuracy for datasets with over 200 trials compared to datasets with 100 to 200 trials underscores the diverse demands and responses of different

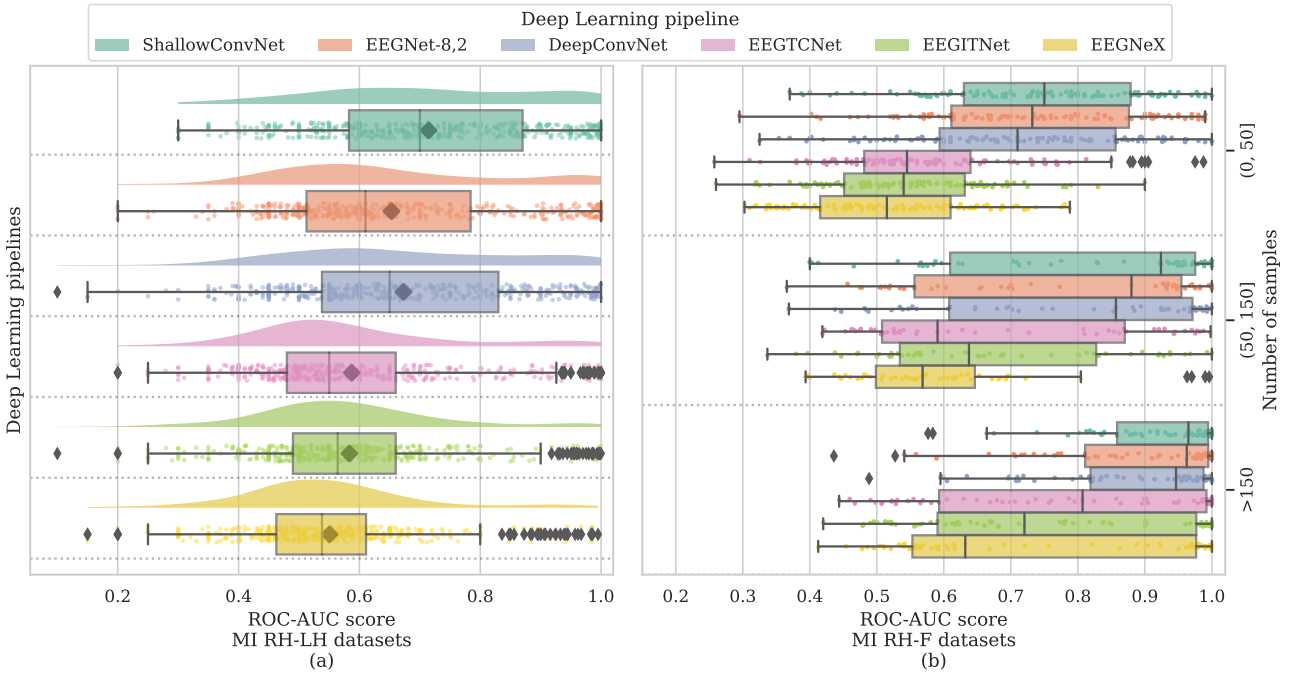


Figure 5: (a) Distributions of AUC scores averaged over all datasets for the right-hand vs left-hand classification task within the DL pipelines. (b) AUC scores averaged across all sessions, subjects, and datasets within the right-hand vs feet MI paradigm for the DL pipelines, segmented based on the number of epochs on the y-axis.

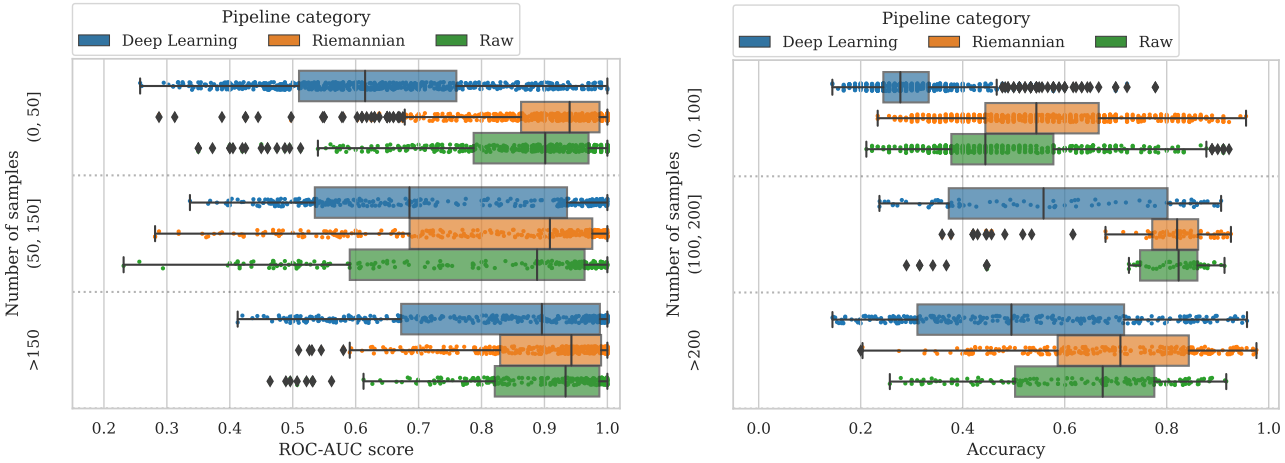
pipelines to varying trial quantities in complex classification tasks.

5.5. Best practices in motor imagery

Motor imagery is a widely utilized paradigm in the BCI community, and a detailed analysis was conducted on the MI results obtained in this benchmark to assist practitioners in experimental design and pipeline selection.

The highest performance levels in MI are achieved in the binary classification distinguishing between right-hand and feet movements. This task notably surpasses the performance of the right-hand/left-hand classification, highlighting a significant disparity in performance between the two tasks. This performance discrepancy is visible when comparing the results of right-hand/left-hand and right-hand/feet tasks in the supplementary Figures C1 and C2, respectively. This trend is consistent across all pipeline categories – Raw, Riemannian and DL – and is prevalent across datasets, with 9 datasets for right-hand/feet tasks and 10 for right-hand/left-hand tasks. Five datasets are common to both tasks, with the AUC notably higher for the right-hand/feet task.

This results holds significant implications for the BCI community, as it provides for the first time valuable insight into selecting the task that yields the highest accuracy in MI. The findings presented here are highgly reliable, drawn from a diverse range of subjects recorded under various protocols and using different hardware setups. Furthermore, since this trend is observable even when analyzing subjects performing both tasks, any confounding effects related to recording



(a) right-hand/feet

(b) all classes

Figure 6: (a) AUC scores for datasets segmented by the number of trials in the right-hand vs feet MI paradigm for each pipeline category (*Deep learning, Riemannian, Raw*). (b) Same results for MI using all available classes and the accuracy metric.

conditions can be effectively eliminated. This insight holds particular importance for BCI practitioners when designing experimental protocols.

To delve deeper into how different pipelines compare in the right-hand/feet classification task, Figure 7-(a) illustrates the AUC scores of the top three pipelines in each category, arranged by their average scores per dataset. The very good performances of Raw and Riemannian pipelines are visible on this plot. The performances of the DL pipelines are below Raw and Riemannian pipelines for the reasons outlined previously.

While the AUC performance measurements offer valuable insights, Figure7-(a) reveals significant subject variability. In practical BCI applications, the choice of the pipeline is often influenced by the algorithm's ranking, with the best-performing algorithm selected for each subject. To evaluate how pipelines fare based on this criterion, pipelines were ranked in each session for all subjects, ranging from 1 (the best) to 16 (the worst) based on their relative scores. Figure 7-(b)visualizes the frequency of sessions (y-axis) where a pipeline achieves a specific rank (x-axis), using distinct colors for each pipeline.

While the results align with the average AUC scores presented in Figure 7-(a), they provide different qualitative insights. Firstly, the ACM+TS+SVM is outperforming other pipelines, consistently securing top positions and rarely falling below the 7th spot. Secondly, the TS+LR and TS+EL Riemannian pipelines exhibit comparable AUC scores, yet TS+EL consistently outperforms other pipelines, whereas TS+LR seldom claims the top spot but frequently ranks among the top 3 pipelines. Lastly, DL pipelines generally rank below the 9th position, excluding the notable exception of the ShallowConvNet pipeline, which attains top 3 rankings in several sessions, despite its lower average AUC score placing it behind CSP-based pipelines.

Accuracy is a primary criterion for selecting a BCI pipeline, but it's also essential to consider calibration time. Pipelines

are often trained immediately following a calibration run or updated after running multiple trials, making execution time a crucial factor in pipeline selection. User interaction is typically paused during pipeline training on the training dataset, emphasizing the significance of efficient execution. This aspect is crucial not only for real-time BCI operation but also for offline evaluation and hyperparameter optimization, as it directly impacts experiment duration.

To address this issue, the average execution times for pipeline categories (Raw, Riemannian, and DL) are presented in Figure 8-(a). These findings, focused on MI right-hand vs feet classification, are applicable across other tasks and paradigms as well. The measurements were conducted using the French Jean Zay CPU/GPU HPC environment, featuring Intel Cascade Lake 6248 CPUs and Nvidia V100 16 GB RAM, encompassing both training and inference phases for a single fold of cross-validation.

Raw pipelines demonstrate the shortest computational time requirements, closely followed by Riemannian pipelines. DL pipelines, on the other hand, exhibit longer training durations on average but remain within an acceptable 30-second range for experimental systems. It's noteworthy that these results were obtained using an early stopping strategy to prevent overfitting, which also contributes to reducing training time.

Environmental impact assessment is crucial in AI-related domains given the exponential growth in these areas. In this study, the direct environmental impact, measured in gCO2 equivalent generated during training and inference phases, is evaluated. The absolute values are significantly influenced by the country's energy production methods, hence the CPU/GPU server localization are important to measure the generated gCO2 equivalent. Precisely measuring algorithm energy consumption is challenging, as existing libraries differ in solutions and measurements. Using Code Carbon [35], the environmental footprint, expressed in gCO2 equivalent emissions, for Riemannian TS+EL, Raw CSP+SVM, and ShallowConvNet pipelines is documented in Figure 8-(b). This provides a unified measure of required computational resources, illustrating that the Riemannian pipeline consumes less energy despite its longer training times as shown in Figure 8-(a).

6. Future Directions for reproducible BCI machine learning pipelines

The path towards open and reproducible approaches in BCI improved with initiatives in open EEG hardware [40, 21], libraries for experimental design [93, 90, 31] and, indeed, machine learning pipelines [4, 14]. For the latter, there is still room for improvements, along two main axes. The first one is to get closer to experimental situations and the second one is to allow fast benchmarking of the most recent BCI decoding techniques. An initiative‡ to tackle the second axes relies on BENCHOPT [80] and aims to provide an easy environment to evaluate novel BCI techniques along with reproducible evaluation conditions.

6.1. Pseudo-online benchmarking

The first limitation is that, in order to maintain compatibility with numerous datasets, the inherent chronology between epochs is disregarded, and the evaluation within each session consists of a simple 5-fold cross-validation over shuffled epochs. However, by leveraging historical knowledge, certain unsupervised

‡ https://github.com/benchopt/benchmark_bci

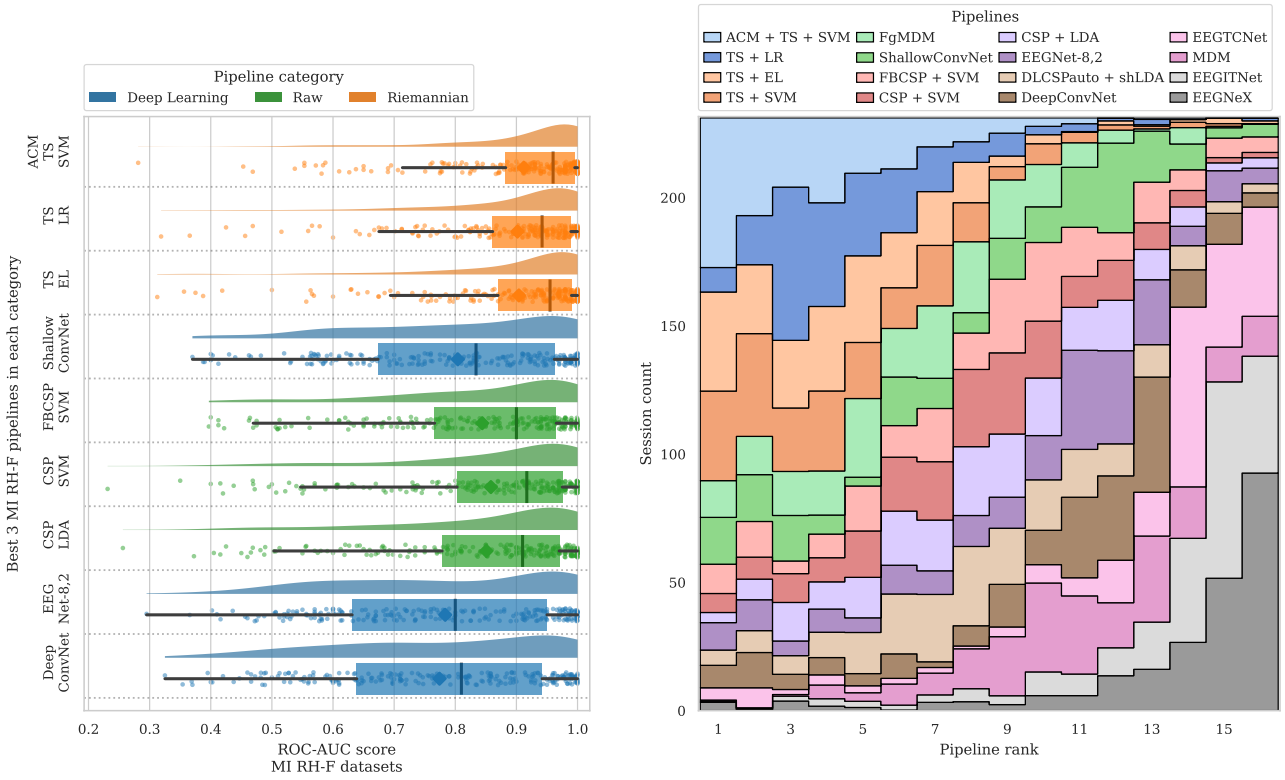


Figure 7: (a) AUC scores are presented for the best three motor imagery pipelines in each category for the right-hand vs feet classification task, ordered by their average score per dataset. (b) Pipeline rankings within individual sessions for the right-hand vs feet task, with each pipeline color-coded. The x-axis displays different ranks achieved by pipelines, while the y-axis indicates the number of sessions each pipeline achieves a specific rank.

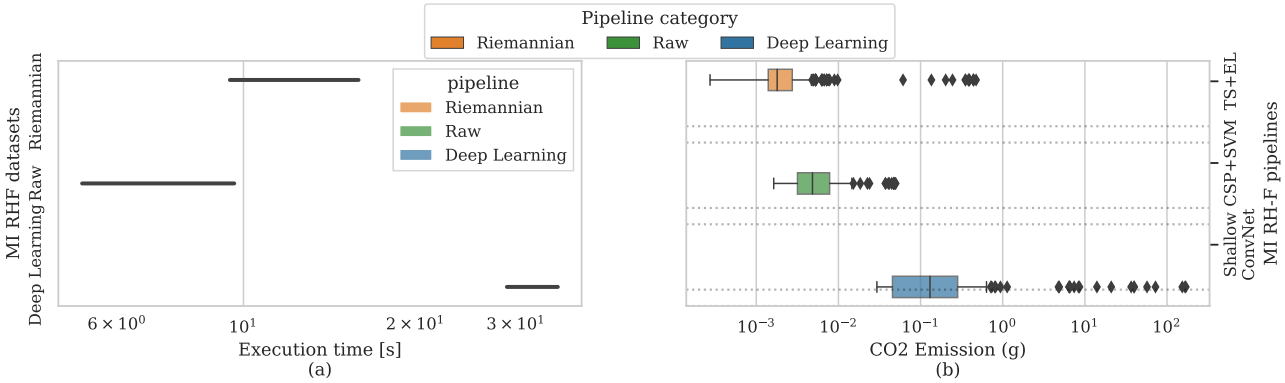


Figure 8: (a) Average execution times in seconds per dataset for the MI right-hand vs feet paradigm, segmented by pipeline category (DL, Riemannian, Raw). (b) Carbon emissions in gCO2 equivalent for the high-ranking pipelines (Raw CSP+SVM, Riemannian TS+EL, DL ShallowConvNet) in the MI right-hand vs feet task.

classification techniques can rival supervised ones [51, 107]. This evaluation presents a significant challenge as it completely overlooks the non-stationarity of the data, causing some classifiers to perform well under these conditions but fail in an online scenario. Consequently, in the near future, it is important to integrate pseudo-online benchmarking of algorithms, like along the lines proposed in [23].

In real online experiments, we anticipate a decrease in accuracy with excessively long calibration periods. User feedback could prove to be highly motivating, a factor that remains undisclosed in offline evaluations.

Regarding pseudo-online evaluation, we hypothesize that a gradual distribution shift occurs during the session. Achieving high accuracy becomes considerably more challenging as training data only capture a limited range of subject variability. Additionally, we anticipate observing learning curves that ascend and level off due to this shift; a plateau phenomenon not observed in offline analyses, where accuracy appears to increase as more data is amassed.

6.2. CVEP paradigm

Most open data in BCI include MI, ERP, and SSVEP paradigms. A recent addition to the landscape of evoked potentials, alongside the well-established ERP and SSVEP scenario, is the emerging Code-modulated Visually Evoked Potentials (c-VEP) paradigm [78]. Drawing an analogy to telecommunications, these three evoked paradigms can be likened to time-domain, frequency-domain, and code-domain multiple access schemes [41]. Notably, research has demonstrated that the c-VEP paradigm exhibits superior performance compared to both ERP and SSVEP paradigms [17], garnering increasing attention and leading to the development of high-speed BCIs measured through Information Transfer Rate (ITR); see, for instance, [82, 113, 110]).

6.3. Character-level benchmarking of ERP and c-VEP

The decoding algorithms for ERP (and soon c-VEP) are mostly benchmarked at the epoch classification level, typically addressing a binary problem where the algorithms need to predict whether an epoch corresponds to a *target* or a *non-target* in the original application. However, this decoding approach overlooks the specific character or target being attended to in the original application, despite the availability of information regarding the stimulus sequence used for each character. Recent advancements have introduced methods that leverage application-level information to reduce the number of target hypotheses [59, 16, 15]. Moreover, unsupervised classifiers have been proposed that capitalize on the structural sequence information, showcasing the potential to surpass established, even supervised algorithms [107, 112]. Unfortunately, the emphasis on binary decoding has led to incomplete availability of the necessary structural information in all existing ERP datasets.

6.4. Cross-dataset transfer learning

Transfer learning has consistently posed a significant challenge in the realm of BCI [55, 57]. While current support includes benchmarks for cross-session and cross-subject scenarios, the absence of benchmarks across datasets remains a gap. Recent advancements in DL models have yielded remarkably high performances in solving cross-dataset transfer learning challenges, particularly in MI paradigms [58, 45, 121, 5]. This surge in interest toward cross-dataset transfer no longer stems solely from

fundamental research but also unfolds as a promising avenue for future BCIs.

7. Conclusion

This contribution represents the largest reproducibility study in EEG-based BCI, leveraging the MOABB library. By utilizing openly available data collected from different hardware sources in varied formats and structures, a systematic benchmarking process was undertaken. Machine learning pipelines from published works were re-implemented in a unified and open framework, aligning with the established standards of the machine learning community. This effort extends to DL pipelines, considering the rapidly evolving processing standards for time series data.

The study's strength lies in the extensive number of subjects analyzed across diverse datasets, enabling robust assessments through meta-analysis statistical techniques. The pipelines undergo evaluation using 5-fold cross-validation, employing the AUC metric for binary classification tasks and accuracy for datasets with multiple classes. Furthermore, the environmental impact of the pipelines is assessed and factored into the reported results.

The primary outcome is a comprehensive and meticulous benchmarking of prominent pipelines from within the BCI literature. Resources are provided to reproduce these results and facilitate comparisons with future works, including result tables in the annexes and on a dedicated online platform to streamline comparisons and avoid unnecessary duplications. The Riemannian pipelines demonstrate the highest accuracy, whereas DL pipelines, while achieving admirable accuracy with extensive trial data, show limitations across most datasets. Although data augmentation techniques and advanced parameterization can enhance their performance, improvements are still required for these off-the-shelf pipelines.

As the benchmark incorporates various pipelines and datasets, recommendations can be formulated regarding the optimal number of trials or channels for designing BCI experiments to achieve peak performance. A detailed analysis of all considered datasets is presented, aiding practitioners in tailoring their experimental designs or selecting specific datasets for evaluating novel pipelines.

Future development avenues are outlined along two key directions. Firstly, benchmarks could progress towards evaluations that mirror real-world experimental conditions, aiming to narrow the disparity between offline assessment and practical online BCI applications. Secondly, the integration of novel BCI paradigms like CVEP and transfer learning approaches across datasets is suggested for further exploration and integration.

8. Acknowledgements

SC, BA and SS were supported by DATAIA Convergence Institute as part of the "Programme d'Investissement d'Avenir", (ANR-17-CONV-0003) operated by LISN-CNRS. This work was granted access to the HPC resources of IDRIS under the allocation 2023-AD011014322 made by GENCI.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens,

B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL https://www.tensorflow.org/.

[2] K. K. Ang, Z. Y. Chin, H. Zhang, and C. Guan. Filter bank common spatial pattern (FBCSP) in brain-computer interface. In *IEEE IJCNN*, pages 2390–2397, 2008.

[3] P. Aricò, F. Aloise, F. Schettini, S. Salinari, D. Mattia, and F. Cincotti. Influence of P300 latency jitter on event related potential-based brain–computer interface performance. *Journal of neural engineering*, 11(3):035008, 2014.

[4] B. Aristimunha, I. Carrara, P. Guetschel, S. Sedlar, P. Rodrigues, J. Sosulski, D. Narayanan, E. Bjareholt, B. Quentin, R. T. Schirrmeister, E. Kalunga, L. Darmet, C. Gregoire, A. Abdul Hussain, R. Gatti, V. Goncharenko, J. Thielen, T. Moreau, Y. Roy, V. Jayaram, A. Barachant, and S. Chevallier. Mother of all BCI Benchmarks, 2023. URL https://github.com/NeuroTechX/moabb.

[5] B. Aristimunha, R. Y. de Camargo, W. H. L. Pinaya, S. Chevallier, A. Gramfort, and C. Rommel. Evaluating the structure of cognitive tasks with transfer learning. *arXiv preprint arXiv:2308.02408*, 2023.

[6] M. Baker. 1,500 scientists lift the lid on reproducibility. *Nature*, 533(7604):452–454, May 2016.

[7] H. Banville, O. Chehab, A. Hyvärinen, D.-A. Engemann, and A. Gramfort. Uncovering the structure of clinical EEG signals with self-supervised learning. *Journal of Neural Engineering*, 18(4): 046020, 2021.

[8] A. Barachant. *Commande robuste d'un effecteur par une interface cerveau machine EEG asynchrone*. PhD thesis, Grenoble, 2012.

[9] A. Barachant. MEG decoding using Riemannian geometry and unsupervised classification. *Grenoble University: Grenoble, France*, 2014.

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

[11] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten. Common spatial pattern revisited by riemannian geometry. In *IEEE International Workshop on Multimedia Signal Processing*, pages 472–476, 2010. doi: 10.1109/MMSP.2010.5662067.

[12] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten. Riemannian geometry applied to BCI classification. *Lva/Ica*, 10:629–636, 2010.

[13] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten. Multiclass brain–computer interface classification by Riemannian geometry. *IEEE Transactions on Biomedical Engineering*, 59(4):920–928, 2011.

[14] A. Barachant, Q. Barthélemy, J.-R. King, A. Gramfort, S. Chevallier, P. L. C. Rodrigues, E. Olivetti, V. Goncharenko, G. W. vom Berg, G. Reguig, A. Lebeurrier, E. Bjäreholt, M. S. Yamamoto, P. Clisson, and M.-C. Corsi. pyriemann, June 2023. URL https://doi.org/10.5281/zenodo.593816.

[15] Q. Barthélemy, S. Chevallier, R. Bertrand-Lalo, and P. Clisson. End-to-end P300 BCI using Bayesian accumulation of riemannian probabilities. *Brain-Computer Interfaces*, 10(1): 50–61, 2023.

[16] L. Bianchi, C. Liti, G. Liuzzi, V. Piccialli, and C. Salvatore. Improving P300 speller performance by means of optimization and machine learning. *Annals of Operations Research*, pages 1–39, 2021.

[17] G. Bin, X. Gao, Y. Wang, B. Hong, and S. Gao. VEP-based brain-computer interfaces: time, frequency, and code modulations [research frontier]. *IEEE Computational Intelligence Magazine*, 4 (4):22–26, 2009.

[18] G. Bin, X. Gao, Z. Yan, B. Hong, and S. Gao. An online multi-channel SSVEP-based brain–computer interface using a canonical correlation analysis method. *Journal of neural engineering*, 6(4):046002, 2009.

[19] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K.-R. Muller. Optimizing spatial filters for robust EEG single-trial analysis. *IEEE Signal processing magazine*, 25(1):41–56, 2007.

[20] N. Boumal. *An introduction to optimization on smooth manifolds*. Cambridge University Press, 2023.

[21] Y. N. Cardona-Álvarez, A. M. Álvarez-Meza, D. A. Cárdenas-Peña, G. A. Castaño-Duque, and G. Castellanos-Dominguez. A novel OpenBCI framework for EEG-based neurophysiological experiments. *Sensors*, 23(7):3763, 2023.

[22] I. Carrara and T. Papadopoulo. Classification of BCI-EEG based on augmented covariance matrix. *arXiv preprint arXiv:2302.04508*, 2023.

[23] I. Carrara and T. Papadopoulo. Pseudo-online framework for BCI evaluation: a MOABB perspective using various MI and SSVEP datasets. *Journal of Neural Engineering*, 21(1):016003, 2024.

[24] G. Cattan, A. Andreev, P. Rodrigues, and M. Congedo. Dataset of an EEG-based BCI experiment in virtual reality and on a personal computer. *arXiv preprint arXiv:1903.11297*, 2019.

[25] G. C. Cawley and N. L. Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *The Journal of Machine Learning Research*, 11:2079–2107, 2010.

[26] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[27] X. Chen, X. Teng, H. Chen, Y. Pan, and P. Geyer. Toward reliable signals decoding for electroencephalogram: A benchmark study to EEGNeX. *arXiv preprint arXiv:2207.12369*, 2022.

[28] S. Chevallier, G. Bao, M. Hammami, F. Marlats, L. Mayaud, D. Annane, F. Lofaso, and E. Azabou. Brain-machine interface for mechanical ventilation using respiratory-related evoked potential. In *Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27*, pages 662–671. Springer, 2018.

[29] S. Chevallier, E. Kalunga, Q. Barthélemy, and F. Yger. Rie-

mannian classification for SSVEP based BCI: offline versus online implementations. In *Brain–Computer Interfaces Handbook: Technological and Theoretical Advances*. Taylor & Francis, 2018.

[30] H. Cho, M. Ahn, S. Ahn, M. Kwon, and S. C. Jun. EEG datasets for motor imagery brain–computer interface. *GigaScience*, 6(7):gix034, 2017.

[31] P. Clisson, R. Bertrand-Lalo, M. Congedo, G. Victor-Thomas, and J. Chatel-Goldman. Timeflux: an open-source framework for the acquisition and near real-time processing of signal streams. In *BCI 2019-8th International Brain-Computer Interface Conference*, 2019.

[32] M. Congedo, M. Goyat, N. Tarrin, G. Ionescu, L. Varnet, B. Rivet, R. Phlypo, N. Jrad, M. Acquadro, and C. Jutten. Brain invaders: a prototype of an open-source P300-based video game working with the OpenViBE platform. In *BCI 2011-5th International Brain-Computer Interface Conference*, pages 280–283, 2011.

[33] 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.

[34] M.-C. Corsi, S. Chevallier, F. D. V. Fallani, and F. Yger. Functional connectivity ensemble method to enhance BCI performance (FUCONE). *IEEE Transactions on Biomedical Engineering*, 69 (9):2826–2838, 2022.

[35] B. Courty, V. Schmidt, Goyal-Kamal, MarionCoutarel, B. Feld, J. Lecourt, LiamConnell, SabAmine, kngoyal, inimaz, M. Léval, L. Blanche, A. Cruveiller, ouminasara, F. Zhao, A. Joshi, A. Bogroff, A. Saboni, H. de Lavoreille, N. Laskaris, E. Abati, D. Blank, Z. Wang, A. Catovic, alencon, M. Stechly, JPW, MinervaBooks, N. Carkaci, and J. Crall. Codecarbon, 2024.

[36] A. Craik, Y. He, and J. L. Contreras-Vidal. Deep learning for electroencephalogram EEG classification tasks: a review. *Journal of neural engineering*, 16 (3):031001, 2019.

[37] A. Delorme. EEG is better left alone. *Scientific reports*, 13(1):2372, 2023.

[38] 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.

[39] L. A. Farwell and E. Donchin. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and clinical Neurophysiology*, 70(6):510–523, 1988.

[40] J. Frey. Comparison of an open-hardware electroencephalography amplifier with medical grade device in brain-computer interface applications. In *PhyCS-International Conference on Physiological Computing Systems*. SCITEPRESS, 2016.

[41] S. Gao, Y. Wang, X. Gao, and B. Hong. Visual and auditory brain–computer interfaces. *IEEE Transactions on Biomedical Engineering*, 61(5):1436–1447, 2014.

[42] A. Garcia Badaracco. *SciKeras*, 2020. URL https://github.com/adriangb/scikeras.

[43] A. Gramfort, M. Luessi, E. Larson, D. A. Engemann, D. Strohmeier, C. Brodbeck, R. Goj, M. Jas, T. Brooks, L. Parkkonen, and M. S. Hämäläinen. MEG and EEG data analysis with MNE-Python. *Frontiers in Neuroscience*, 7(267):1–13, 2013. doi: 10.3389/fnins.2013.00267.

[44] M. Grosse-Wentrup, C. Liefhold, K. Gramann, and M. Buss. Beamforming in noninvasive brain–computer interfaces. *IEEE Transactions on Biomedical Engineering*, 56(4):1209–1219, 2009.

[45] P. Guetschel and M. Tangermann. Transfer learning between motor imagery datasets using deep learning - validation of framework and comparison of datasets, Nov. 2023.

[46] C. Guger, S. Daban, E. Sellers, C. Holzner, G. Krausz, R. Carabalona, F. Gramatica, and G. Edlinger. How many people are able to control a P300-based brain–computer interface (BCI)? *Neuroscience letters*, 462(1):94–98, 2009.

[47] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant. Array programming with NumPy. *Nature*, 585(7825): 357–362, Sept. 2020.

[48] L. V. Hedges and I. Olkin. *Statistical methods for meta-analysis*. Academic press, 2014.

[49] U. Hoffmann, J.-M. Vesin, T. Ebrahimi, and K. Diserens. An efficient P300-based brain–computer interface for disabled subjects. *Journal of Neuroscience methods*, 167(1):115–125, 2008.

[50] D. Hübner, T. Verhoeven, K. Schmid, K.-R. Müller, M. Tangermann, and P.-J. Kindermans. Learning from label proportions in brain-computer interfaces: Online unsupervised learning with guarantees. *PloS one*, 12(4):e0175856, 2017.

[51] D. Hüebner, T. Verhoeven, K.-R. Müeller, P.-J. Kindermans, and M. Tangermann. Unsupervised learning for brain-computer interfaces based on event-related potentials: Review and online comparison. *IEEE Computational Intelligence Magazine*, 13(2): 66–77, 2018.

[52] T. M. Ingolfsson, M. Hersche, X. Wang, N. Kobayashi, L. Cavigelli, and L. Benini. EEG-TCNet: An accurate temporal convolutional network for embedded motor-imagery brain–machine interfaces. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2958–2965. IEEE, 2020.

[53] M. Jay, V. Ostapenco, L. Lefèvre, D. Trystram, A.-C. Orgerie, and B. Fichel. An experimental comparison of software-based power meters: focus on CPU and GPU. In *IEEE/ACM international symposium on cluster, cloud and internet computing*, pages 1–13, 2023.

[54] V. Jayaram and A. Barachant. MOABB: trustworthy algorithm benchmarking for bcis. *Journal of neural engineering*, 15 (6):066011, 2018.

[55] V. Jayaram, M. Alamgir, Y. Altun, B. Scholkopf, and M. Grosse-Wentrup. Transfer learning in brain-computer in-

terfaces. *IEEE Computational Intelligence Magazine*, 11(1):20–31, 2016.

[56] E. K. Kalunga, S. Chevallier, Q. Barthélemy, K. Djouani, E. Monacelli, and Y. Hamam. Online SSVEP-based BCI using Riemannian geometry. *Neurocomputing*, 191:55–68, 2016.

[57] E. K. Kalunga, S. Chevallier, and Q. Barthélemy. Transfer learning for SSVEP-based BCI using riemannian similarities between users. In *EUSIPCO*, pages 1685–1689, 2018.

[58] S. Khazem, S. Chevallier, Q. Barthélemy, K. Haroun, and C. Noûs. Minimizing subject-dependent calibration for BCI with Riemannian transfer learning. In *International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 523–526. IEEE, 2021.

[59] P.-J. Kindermans, H. Verschore, D. Verstraeten, and B. Schrauwen. A P300 BCI for the masses: Prior information enables instant unsupervised spelling. *Advances in neural information processing systems*, 25, 2012.

[60] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[61] Z. J. Koles, M. S. Lazar, and S. Z. Zhou. Spatial patterns underlying population differences in the background EEG. *Brain topography*, 2:275–284, 1990.

[62] L. Korczowski, M. Cederhout, A. Andreev, G. Cattan, P. L. C. Rodrigues, V. Gautheret, and M. Congedo. *Brain Invaders calibration-less P300-based BCI with modulation of flash duration Dataset (bi2015a)*, 2019.

[63] L. Korczowski, M. Cederhout, A. Andreev, G. Cattan, P. L. C. Rodrigues, V. Gautheret, and M. Congedo. *Brain Invaders Cooperative versus Competitive: Multi-User P300-based Brain-Computer Interface Dataset (bi2015b)*, 2019.

[64] L. Korczowski, E. Ostaschenko, A. Andreev, G. Cattan, P. L. C. Rodrigues, V. Gautheret, and M. Congedo. *Brain Invaders calibration-less P300-based BCI using dry EEG electrodes Dataset (bi2014a)*, 2019.

[65] L. Korczowski, E. Ostaschenko, A. Andreev, G. Cattan, P. L. C. Rodrigues, V. Gautheret, and M. Congedo. *Brain Invaders Solo versus Collaboration: Multi-User P300-based Brain-Computer Interface Dataset (bi2014b)*, 2019.

[66] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[67] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance. EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces. *Journal of neural engineering*, 15(5):056013, 2018.

[68] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–50. Springer, 2002.

[69] M.-H. Lee, O.-Y. Kwon, Y.-J. Kim, H.-K. Kim, Y.-E. Lee, J. Williamson, S. Fazli, and S.-W. Lee. EEG dataset and openbmi toolbox for three BCI paradigms: An investigation into BCI illiteracy. *GigaScience*, 8(5):giz002, 2019.

[70] R. Leeb, F. Lee, C. Keinrath, R. Scherer, H. Bischof, and G. Pfurtscheller. Brain–computer communication: motivation, aim, and impact of exploring a virtual apartment. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15(4):473–482, 2007.

[71] A.-L. Ligozat, J. Lefevre, A. Bugeau, and J. Combaz. Unraveling the hidden environmental impacts of AI solutions for environment life cycle assessment of AI solutions. *Sustainability*, 14(9):5172, 2022.

[72] Z. Lin, C. Zhang, W. Wu, and X. Gao. Frequency recognition based on canonical correlation analysis for SSVEP-based BCIs. *IEEE transactions on biomedical engineering*, 53(12):2610–2614, 2006.

[73] 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, 2010.

[74] 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(2):R1, 2007.

[75] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy, and F. Yger. A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update. *Journal of neural engineering*, 15(3): 031005, 2018.

[76] A. S. Luccioni, S. Viguier, and A.-L. Ligozat. Estimating the carbon footprint of bloom, a 176b parameter language model. *Journal of Machine Learning Research*, 24(253):1–15, 2023.

[77] S. J. Luck. *An introduction to the event-related potential technique*. The MIT Press, 2014.

[78] V. Martínez-Cagigal, J. Thielen, E. Santamaria-Vazquez, S. Pérez-Velasco, P. Desain, and R. Hornero. Brain-computer interfaces based on code-modulated visual evoked potentials (c-VEP): A literature review. *Journal of Neural Engineering*, 18(6):061002, 2021.

[79] M. Moakher. A differential geometric approach to the geometric mean of symmetric positive-definite matrices. *SIAM journal on matrix analysis and applications*, 26(3):735–747, 2005.

[80] T. Moreau, M. Massias, A. Gramfort, P. Ablin, P.-A. Bannier, B. Charlier, M. Dagréou, T. D. la Tour, G. Durif, C. F. Dantas, Q. Klopfenstein, J. Larsson, E. Lai, T. Lefort, B. Malézieux, B. Moufad, B. T. Nguyen, A. Rakotomamonjy, Z. Ramzi, J. Salmon, and S. Vaiter. Benchopt: Reproducible, efficient and collaborative optimization benchmarks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, New-Orleans, LA, USA, Nov. 2022. Curran Associates, Inc.

[81] J. Müller-Gerking, G. Pfurtscheller, and H. Flyvbjerg. Designing optimal spatial filters for single-trial EEG classification in a movement task. *Clinical neurophysiology*, 110(5):787–798, 1999.

[82] S. Nagel and M. Spüler. World's fastest brain-computer interface: combining EEG2Code with deep learning. *PloS one*, 14(9):e0221909, 2019.

[83] M. Nakanishi, Y. Wang, Y.-T. Wang, Y. Mitsukura, and T.-P. Jung. Enhancing unsupervised canonical correlation analysis-based frequency detection of SSVEPs by incorporating background EEG. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 3053–3056. IEEE, 2014.

[84] M. Nakanishi, Y. Wang, Y.-T. Wang, and T.-P. Jung. A comparison study of canonical correlation analysis based methods for detecting steady-state visual evoked potentials. *PloS one*, 10(10): e0140703, 2015.

[85] M. Nakanishi, Y. Wang, X. Chen, Y.-T. Wang, X. Gao, and T.-P. Jung. Enhancing detection of SSVEPs for a high-speed brain speller using task-related component analysis. *IEEE Transactions on Biomedical Engineering*, 65(1):104–112, 2017.

[86] C. S. Nam, A. Nijholt, and F. Lotte. *Brain–computer interfaces handbook: technological and theoretical advances*. CRC Press, 2018.

[87] V. P. Oikonomou, G. Liaros, K. Georgiadis, E. Chatzilari, K. Adam, S. Nikolopoulos, and I. Kompatsiaris. Comparative evaluation of state-of-the-art algorithms for SSVEP-based BCIs. *arXiv preprint arXiv:1602.00904*, 2016.

[88] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035, 2019.

[89] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[90] J. Peirce, R. Hirst, and M. MacAskill. *Building experiments in PsychoPy*. Sage, 2022.

[91] C. R. Pernet, S. Appelhoff, K. J. Gorgolewski, G. Flandin, C. Phillips, A. Delorme, and R. Oostenveld. EEG-BIDS, an extension to the brain imaging data structure for electroencephalography. *Scientific data*, 6(1):103, 2019.

[92] 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.

[93] Y. Renard, F. Lotte, G. Gibert, M. Congedo, E. Maby, V. Delannoy, O. Bertrand, and A. Lécuyer. Openvibe: An open-source software platform to design, test, and use brain–computer interfaces in real and virtual environments. *Presence*, 19(1):35–53, 2010.

[94] A. Riccio, L. Simione, F. Schettini, A. Pizzimenti, M. Inghilleri, M. O. Belardinelli, D. Mattia, and F. Cincotti. Attention and P300-based BCI performance in people with amyotrophic lateral sclerosis. *Frontiers in human neuroscience*, 7:732, 2013.

[95] B. Rivet, A. Souloumiac, V. Attina, and G. Gibert. xDAWN algorithm to en-

hance evoked potentials: application to brain–computer interface. *IEEE Transactions on Biomedical Engineering*, 56 (8):2035–2043, 2009.

[96] S. RMJ. The american soldier, vol. 1: Adjustment during army life, 1949.

[97] C. Rommel, J. Paillard, T. Moreau, and A. Gramfort. Data augmentation for learning predictive models on EEG: a systematic comparison. *Journal of Neural Engineering*, 19(6):066020, 2022.

[98] R. N. Roy. *Neuroergonomics and physiological computing contributions to human-machine interaction*. PhD thesis, Université Paul Sabatier, 2022.

[99] Y. Roy, H. Banville, I. Albuquerque, A. Gramfort, T. H. Falk, and J. Faubert. Deep learning-based electroencephalography analysis: a systematic review. *Journal of neural engineering*, 16(5): 051001, 2019.

[100] A. Salami, J. Andreu-Perez, and H. Gillmeister. EEG-ITNet: An explainable inception temporal convolutional network for motor imagery classification. *IEEE Access*, 10:36672–36685, 2022.

[101] G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, and J. R. Wolpaw. BCI2000: a general-purpose brain-computer interface (BCI) system. *IEEE Transactions on biomedical engineering*, 51(6):1034–1043, 2004.

[102] R. Scherer, J. Faller, E. V. Friedrich, E. Opisso, U. Costa, A. Kübler, and G. R. Müller-Putz. Individually adapted imagery improves brain-computer interface performance in end-users with disability. *PloS one*, 10(5):e0123727, 2015.

[103] 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*, 38(11): 5391–5420, 2017.

[104] S. Schneider, A. Baevski, R. Collobert, and M. Auli. wav2vec: Unsupervised pre-training for speech recognition. *arXiv preprint arXiv:1904.05862*, 2019.

[105] J. Shin, A. von Lühmann, B. Blankertz, D.-W. Kim, J. Jeong, H.-J. Hwang, and K.-R. Müller. Open access dataset for EEG+ NIRS single-trial classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10): 1735–1745, 2016.

[106] J. Sosulski and M. Tangermann. Spatial filters for auditory evoked potentials transfer between different experimental conditions. In *GBCIC*, 2019.

[107] J. Sosulski and M. Tangermann. UMM: Unsupervised mean-difference maximization. *arXiv preprint arXiv:2306.11830*, 2023.

[108] J. Sosulski, D. Hübner, A. Klein, and M. Tangermann. Online optimization of stimulation speed in an auditory brain-computer interface under time constraints. *arXiv preprint arXiv:2109.06011*, 2021.

[109] D. Steyrl, R. Scherer, J. Faller, 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.

[110] Q. Sun, L. Zheng, W. Pei, X. Gao, and Y. Wang. A 120-target brain-computer interface based on code-modulated vi-

sual evoked potentials. *Journal of Neuroscience Methods*, 375:109597, 2022.

[111] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K. J. Miller, G. Mueller-Putz, et al. Review of the BCI competition IV. *Frontiers in neuroscience*, page 55, 2012.

[112] J. Thielen, P. van den Broek, J. Farquhar, and P. Desain. Broad-band visually evoked potentials: re (con) volution in brain-computer interfacing. *PloS one*, 10(7):e0133797, 2015.

[113] J. Thielen, P. Marsman, J. Farquhar, and P. Desain. From full calibration to zero training for a code-modulated visual evoked potentials for brain–computer interface. *Journal of Neural Engineering*, 18(5):056007, 2021.

[114] M. Tietz, T. J. Fan, D. Nouri, B. Bossan, and skorch Developers. *skorch: A scikit-learn compatible neural network library that wraps PyTorch*, July 2017. URL https://skorch.readthedocs.io/en/stable/.

[115] E. Vaineau, A. Barachant, A. Andreev, P. C. Rodrigues, G. Cattan, and M. Congedo. Brain invaders adaptive versus non-adaptive P300 brain-computer interface dataset. *arXiv preprint arXiv:1904.09111*, 2019.

[116] G. Van Veen, A. Barachant, A. Andreev, G. Cattan, P. C. Rodrigues, and M. Congedo. Building brain invaders: EEG data of an experimental validation. *arXiv preprint arXiv:1905.05182*, 2019.

[117] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[118] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental algorithms for scientific computing in python. *Nature Methods*, 17:261–272, 2020.

[119] Z. Wan, R. Yang, M. Huang, N. Zeng, and X. Liu. A review on transfer learning in EEG signal analysis. *Neurocomputing*, 421:1–14, 01 2021.

[120] Y. Wang, X. Chen, X. Gao, and S. Gao. A benchmark dataset for SSVEP-based brain–computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10):1746–1752, 2016.

[121] X. Wei, A. A. Faisal, M. Grosse-Wentrup, A. Gramfort, S. Chevallier, V. Jayaram, C. Jeunet, S. Bakas, S. Ludwig, K. Barmpas, et al. 2021 BEETL competition: Advancing transfer learning for subject independence & heterogenous EEG data sets. In *NeurIPS 2021 Competitions and Demonstrations Track*, pages 205–219. PMLR, 2022.

[122] F. Wilcoxon. Individual comparisons by ranking methods. In *Breakthroughs in Statistics: Methodology and Distribution*, pages 196–202. Springer, 1992.

[123] Z. Wu and D. Yao. The influence of cognitive tasks on different frequencies

steady-state visual evoked potentials. *Brain topography*, 20:97–104, 2007.

[124] F. Yger, M. Berar, and F. Lotte. Riemannian approaches in brain-computer interfaces: a review. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10):1753–1762, 2016.

[125] W. Yi, S. Qiu, K. Wang, H. Qi, L. Zhang, P. Zhou, F. He, and D. Ming. Evaluation of EEG oscillatory patterns and cognitive process during simple and compound limb motor imagery. *PloS one*, 9(12):e114853, 2014.

[126] 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.

[127] Z. Zhang, S. hua Zhong, and Y. Liu. TorchEEG, 2024. URL https://torcheeg.readthedocs.io.

[128] B. Zhou, X. Wu, Z. Lv, L. Zhang, and X. Guo. A fully automated trial selection method for optimization of motor imagery based brain-computer interface. *PloS one*, 11(9):e0162657, 2016.

Appendix A. Detailed pipelines evaluation

This section provides details regarding the automatic parametrization and the evaluation conducted in the benchmark.

Appendix A.1. Specialized grid search

The parametrization of evaluated pipelines should be generic to ensure a fair evaluation and automatic to avoid information leakage. A dictionary structure containing the parameter to search for each element of the machine learning pipeline is implemented to ensure this process when an evaluation is launched. As shown below, the `param_grid` structure, with names matching the `pipeline` structure, could be passed to the function `evaluation.process()`.

```
pipelines = {}
pipelines["GridSearchEN"] = Pipeline(
    steps=[
        ("Covariances", Covariances("cov")),
        ("Tangent_Space",
            TangentSpace(metric="riemann")),
        (
            "LogistReg",
            LogisticRegression(
                penalty="elasticnet",
                l1_ratio=0.70,
                intercept_scaling=1000.0,
                solver="saga",
                max_iter=1000,
            ),
        ),
    ]
)

param_grid = {}
param_grid["GridSearchEN"] = {
    "LogistReg__l1_ratio":
        [0.15, 0.30, 0.45, 0.60, 0.75],
}

evaluation = WithinSessionEvaluation(
    paradigm=paradigm,
    datasets=dataset,
    overwrite=True,
    random_state=42,
    hdf5_path=path,
    n_jobs=-1,
)
result = evaluation.process(pipelines,
    param_grid)
```

Appendix A.2. Other evaluation types

The proposed benchmark focus on within-session evaluation, but it is possible to conduct

Table A1: Parameter used in Grid Search. For the ACM+TS+SVM pipeline, we reduce the hyperparameter search due to computational constraint (both order and lag to $[1-5]$ for datasets with more than 60 electrodes - Cho2017, Lee2019-MI, PhysionetMI and Weibo2014. While we select parameters to $[1-3]$ for datasets with more than 100 electrodes - Schirrmeister2017 and GrosseWentrup2009)

Pipeline	Parameter	Value
CSP + SVMGrid	csp_nfilter	[2 - 8]
	svc_C	[0.5, 1, 1.5]
	svc_kernel	["rbf", "linear"]
EnGrid	logisticregression_l1 ratio	[0.20, 0.30, 0.45, 0.65, 0.75]
LogVarGrid	svc_C	[0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100]
Tangent Space SVM (SVM) Grid	svc_C	[0.5, 1, 1.5]
	svc_kernel	["rbf", "linear"]
ACM + TANG + Support Vector Machine (SVM) Grid	augmenteddataset_order	[1 - 10]
	augmenteddataset_lag	[1 - 10]
	svc_C	[0.5, 1, 1.5]
	svc_kernel	["rbf", "linear"]

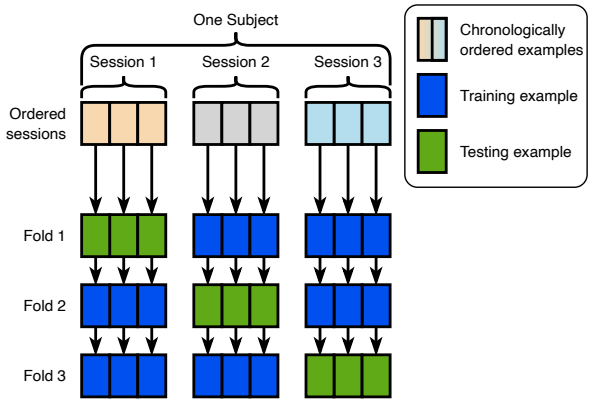


Figure A1: Cross-session evaluation

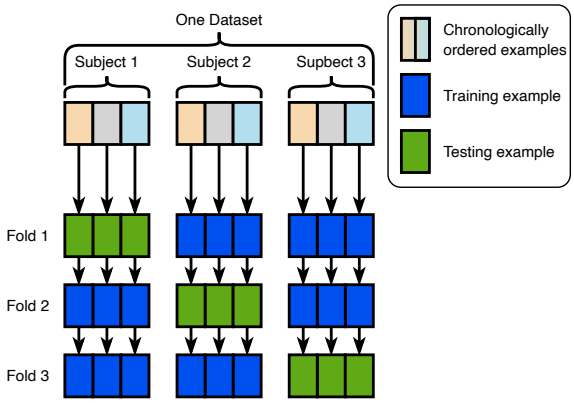


Figure A2: Cross-subject evaluation

other evaluation types like cross-session or cross-subject. The structure of the cross-session evaluation is shown in Figure A1 and Figure A2 shows the structure of the cross-subject one.

Appendix B. Machine learning algorithms

This section describes the machine learning pipelines and the choices of implementation.

Appendix B.1. CSP-baseline pipelines

Mathematically, the CSP algorithm seeks to find spatial filters by solving a generalized eigenvalue problem. Let X_1 and X_2 be the data matrices of the band-pass filtered EEG signals for two conditions, each with dimensions of (time $\times$ channel). The CSP algorithm constructs discriminative and common activity matrices, S_d and S_c, respectively. These matrices are defined as follows:

$$S_d = \Sigma_1 - \Sigma_2 \quad \text{and} \quad S_c = \Sigma_1 + \Sigma_2, \quad \text{(B.1)}$$

where $\Sigma_1 = X_1^T X_1$ and $\Sigma_2 = X_2^T X_2$ are the estimates of the condition covariance matrices. Then, the objective of CSP is to find spatial filters $v_j \in \mathbb{R}^C$ that maximize the EEG signal bandpower variance between examples from different conditions while simultaneously minimizing its variance between examples from the same condition. This translates into:

$$\text{argmax}_V \frac{V^T S_d V}{V^T S_c V} \tag{B.2}$$

which is optimized by solving the following generalised eigenvalue problem [92]:

$$S_d v = \lambda S_c v, \tag{B.3}$$

and selecting the filters v that yield the largest eigenvalues λ.

For classification purposes, CSP utilizes log-variance features extracted from the filtered signals projected onto the CSP filters. Typically, a small number of patterns (2 to 6) are selected based on the corresponding eigenvalues. The patterns, denoted as a_j, provide insights into the specific information captured by the corresponding filters v_j. Each filter v_j extracts the activity spanned by pattern, a_j while canceling out other activities spanned by different patterns. This allows for discrimination between different mental states based on the log-variance features. Linear classifiers, such as linear discriminant analysis, are commonly used due to the approximately Gaussian distribution of the log-variance features. In MOABB, we implemented two different decision head, the SVM and Linear Discriminant Analysis (LDA), with a GridSearch in some parameters [11].

Appendix B.2. CCA-based pipelines

Based on the work of Hotelling, the CCA aims at finding a canonical space where the correlation of two sets of variables is maximized. Considering the total covariance matrix C of two sets of variables x and y, the within-set covariances matrices are respectively denoted C_{xx} and C_{yy} and the between-sets covariance matrices are $C_{xy} = C_{yx}^\top$. The canonical correlation is defined as:

$$\rho = \max_{w_x, w_y} \frac{w_x^\top C_{xy} w_y}{\sqrt{w_x^\top C_{xx} w_x w_y^\top C_{yy} w_y}} \tag{B.4}$$

with w_x and w_y the projection vectors that maximizes canonical correlation. The solution is provided by a generalized eigenproblem formulated as $C_{xy} C_y^{-1} y C_{yx} w_x = \lambda^2 C_{xx} w_x$. As C_{xx} and C_{yy} are symmetric positive definite matrices, it is possible to rewrite the problem as symmetric standard eigenproblem $Ax = \lambda x$ that could be solved with any linear algebra library.

In SSVEP, the CCA is applied on a set of x EEG trials and a set y of reference sinusoid signals as described in [72]. The frequency of the sine and cosine reference signals should match the stimulus frequency and its harmonics. The obtained w_x could be seen as filter acting on EEG signal to enhance the signal components synchronized with the visual stimulus.

Indeed, the choice reference signals is crucial to obtain robust results and could be complex to parameterize. The multiset canonical correlation analysis (MsetCCA) generalizes the CCA for multiple references[126], with the objective to learn the optimal reference signals without constraint on the sine/cosine shape of the reference set.

The goal of filters learned from CCA is to enhance the EEG activity generated by cortical stimulus response. The cortical signal component unrelated to the task could be filtered out as well, as introduced in task-related component analysis (TRCA) [85] for

SSVEP. In this case, an optimization process is designed to find the w_x that maximized the inter-trial covariance of EEG signals.

Appendix B.3. Riemannian pipelines

As described in subsection 4.2, the Riemannian pipelines could be directly defined on the manifold. This is the case of the Minimum Distance to Mean (MDM) classifier, that computes the average of each class with Eq. (2) and use the distance defined in Eq. (1) between an unseen sample and each class center to make a prediction.

To mitigate the effect of increased dimensions, i.e., for EEG recorded with a high number of electrodes, a geodesic filtering could be applied before MDM classification, namely the FgMDM. This filtering step implements a linear discriminant analysis in the tangent space to project all trials on a single hyperplane and the projection back to the manifold.

The second option is to project the samples in the tangent space, vectorize samples that are symmetric matrices and train a classifier on those vectorized data. Algorithms such as logistic regression (LR), logistic regression under ℓ_1 and ℓ_2 norm penalties (ElasticNet, EL) or support vector machine (SVM) have been described in the literature.

When considering MI, Riemannian classifiers operate directly on covariance matrices estimated from the EEG signals. The transient information of ERP require to first estimate a average ERP or, better, an average ERP filtered XDAWN. For SSVEP, the relevant information is spectral and the covariances need to be estimated on the bandpass filtered signal for each stimulation frequency. All those preprocessing are described in great detail in [124, 29].

Appendix C. Experimental results

This section provides a global overview of the pipeline scores for the different paradigms with raincloud plots. The pipelines are grouped by categories: Deep learning, Riemannian, and Raw. The small points correspond to the scores obtained on the individual sessions, with an exception for the last row, i.e. *Average*, where they correspond to the average score over one dataset. The curves above indicate a density estimation of these individual scores. The diamond shapes connected by vertical lines indicate the within-dataset average. Finally, the boxes and horizontal black bars indicate the quartiles.

Appendix D. Detailed results for each pipeline

This section provides the tables indicating the ROC-AUC for binary classification problems and the accuracy for multi-class tasks. The main objective is to provide a reference benchmark that could be easily reproduced to verify the results or used as-is to compare with new pipelines. It is thus possible to save energy and resources, avoiding reproducing already existing validated results with a simple copy-paste. To this end, the tables provide here the average result of pipelines across all subjects for a given dataset using within-session evaluation. We did not provide subject-by-subject results for space reasons and we know that this benchmark is meant to evolve with new pipelines, datasets, and evaluation methods. We mirror those tables on a website that will allow further additions and available up-to-date results.

Table D1: Summary of performances via average on all the motor imagery datasets, for classification using all the labels. Intra-session validation. Bold numbers represent the best score in each dataset.

pipeline	AlexandreMotorImagery	BNCI2014-001	PhysionetMotorImagery	Schirrmeister2017	Weibo2014	Zhou2016	Average
ACM+TS+SVM	69.37±15.07	**77.82±12.23**	55.44±14.87	82.50±10.20	**63.89±11.01**	**85.25±4.06**	72.38
CSP+LDA	61.04±17.22	65.99±15.47	47.73±14.35	72.97±10.42	39.45±11.87	82.96±5.20	61.69
CSP+SVM	62.92±16.89	66.88±15.22	48.52±14.62	75.89±10.55	44.08±11.95	83.08±5.33	63.56
DLCSPauto+shLDA	60.63±17.91	66.31±15.36	46.85±14.65	72.82±10.44	38.84±11.97	82.06±5.57	61.25
DeepConvNet	37.71±4.56	35.29±8.26	27.68±3.91	56.78±18.11	24.17±9.80	55.69±5.61	39.55
EEGITNet	36.04±3.43	35.55±6.35	26.15±4.95	70.44±14.68	25.78±8.00	50.68±16.27	40.77
EEGNeX	37.71±9.64	45.62±15.29	26.69±5.64	67.56±14.15	30.22±11.02	56.42±11.29	44.03
EEGNet-8,2	43.96±8.62	60.46±20.20	29.04±7.03	76.99±13.05	35.35±14.05	83.34±3.58	54.86
EEGTCNet	34.17±1.86	41.65±13.73	25.79±3.85	71.11±11.96	17.95±3.88	37.19±2.57	37.98
FBCSP+SVM	65.00±17.56	66.53±12.05	45.49±12.54	75.94±8.59	45.21±10.05	81.99±4.65	63.36
FgMDM	65.63±15.63	70.14±15.13	55.04±14.17	82.97±10.08	56.94±9.26	83.07±4.96	68.97
MDM	60.62±13.69	61.60±14.20	42.96±12.98	52.03±10.11	33.41±8.67	76.05±7.10	54.45
ShallowConvNet	50.00±12.94	72.47±16.50	41.87±12.50	85.13±9.57	48.94±10.36	85.02±3.78	63.91
TS+EL	**69.79±13.75**	72.38±14.85	**59.93±14.07**	**85.53±9.40**	63.84±8.77	84.54±4.93	**72.67**
TS+LR	69.17±14.79	71.97±15.46	58.55±14.06	84.60±9.28	62.76±8.39	84.88±4.63	71.99
TS+SVM	67.92±12.74	70.76±15.08	58.46±15.15	84.41±9.56	61.47±9.62	83.66±4.55	71.11
Average	55.73	61.34	43.51	74.85	43.27	74.74	58.91

Table D2: Summary of performances via average on all the P300 datasets, for classification using left vs. right motor imagery task. Intra-session validation. Bold numbers represent the best score in each dataset.

pipeline	BNCI2014-001	BNCI2014-004	Cho2017	GrosseWentrup2009	Lee2019-MI	PhysionetMotorImagery	Schirrmeister2017	Shin2017A	Weibo2014	Zhou2016	Average
ACM+TS+SVM	**91.71±10.30**	**82.67±15.33**	73.56±14.54	86.60±15.12	83.05±13.97	63.55±21.24	85.82±13.98	68.97±23.45	84.78±13.33	95.03±4.76	81.57
CSP+LDA	82.34±17.26	80.10±14.93	71.38±14.54	76.44±20.95	76.88±17.41	65.75±17.37	77.23±18.43	**72.30±21.79**	80.72±15.29	93.15±6.88	77.63
CSP+SVM	83.07±16.53	79.27±15.68	71.92±14.25	77.81±21.27	77.27±16.73	65.71±17.90	79.24±20.07	70.11±22.19	79.84±15.86	92.96±7.86	77.72
DLCSPauto+shLDA	82.75±16.69	79.87±15.11	71.16±14.53	76.40±20.83	76.69±17.23	65.07±17.68	77.02±18.48	70.34±23.30	80.16±15.23	92.56±7.21	77.2
DeepConvNet	82.07±15.52	72.36±18.53	71.67±12.91	82.38±15.39	70.65±15.76	59.57±16.77	81.23±17.39	56.03±19.18	73.64±15.78	94.42±6.21	74.4
EEGITNet	75.27±16.37	65.10±15.32	57.20±12.21	72.19±14.71	59.17±11.72	52.71±11.11	74.66±20.52	52.18±16.78	59.35±14.06	69.41±14.66	63.72
EEGNeX	66.28±13.22	66.53±17.10	53.28±10.60	57.00±7.52	55.12±10.05	51.20±10.63	68.58±19.37	49.02±17.58	57.97±15.65	61.56±14.60	58.65
EEGNet-8,2	77.15±19.33	69.50±19.50	66.79±16.34	83.02±18.08	65.67±16.43	59.55±15.95	80.20±18.13	57.99±17.28	66.46±21.78	94.84±2.83	72.12
EEGTCNet	67.46±20.81	69.70±19.55	58.34±12.63	68.45±16.27	55.68±12.75	55.90±12.74	75.62±22.33	51.26±16.77	63.16±18.32	82.24±9.40	64.78
FBCSP+SVM	84.44±16.00	80.39±16.05	67.91±15.63	79.65±18.63	75.07±16.97	58.45±13.93	81.44±17.89	65.63±21.64	76.81±18.88	92.64±5.01	76.24
FgMDM	86.53±12.14	79.28±15.25	72.90±12.70	87.02±13.20	81.34±13.93	**68.46±19.06**	86.71±13.79	70.86±23.36	78.41±14.85	92.54±6.67	80.41
LogVar+LDA	77.96±15.09	78.51±15.25	64.49±10.08	78.71±11.69	66.21±12.06	61.94±14.41	78.44±13.76	61.78±22.77	74.13±10.40	88.39±8.57	73.06
LogVar+SVM	75.86±16.45	78.30±15.18	65.46±11.71	81.73±12.40	73.83±13.85	62.35±16.87	79.42±13.66	61.38±22.68	74.85±11.33	88.47±8.50	74.17
MDM	81.69±14.94	77.66±15.78	63.39±13.69	64.29±8.04	70.23±13.87	54.76±16.79	61.53±16.41	62.99±21.25	58.80±16.13	90.70±7.11	68.6
ShallowConvNet	86.17±13.74	72.36±18.05	73.84±14.95	86.53±13.00	75.83±15.04	65.19±15.80	84.82±15.29	60.80±19.27	79.10±12.63	**95.65±5.55**	78.03
TRCSP+LDA	79.84±16.28	79.78±15.22	71.85±13.84	78.29±16.66	76.26±15.41	67.24±17.23	79.14±15.91	67.30±23.19	79.33±14.43	93.53±6.38	77.25
TS+EL	86.44±13.20	79.75±15.44	**76.23±14.21**	**89.25±12.00**	**84.74±13.19**	67.91±20.03	**88.65±12.98**	68.68±23.64	**85.29±12.10**	94.35±6.04	**82.13**
TS+LR	87.41±12.58	80.09±15.01	75.01±13.71	87.60±13.20	83.09±13.46	67.28±19.19	87.22±13.83	69.31±23.06	83.62±13.88	94.16±6.33	81.48
TS+SVM	86.48±13.58	79.41±15.26	74.62±14.19	88.08±13.58	83.57±14.08	68.18±19.92	87.64±13.48	68.45±24.25	83.72±14.28	93.37±6.30	81.35
Average	81.1	76.35	68.47	79.02	73.18	62.15	79.72	63.44	74.74	89.47	74.76

Table D3: Summary of performances via average on all the P300 datasets, for classification using right hand vs. feet tasks motor imagery task. Intra-session validation. Bold numbers represent the best score in each dataset.

pipeline	AlexandreMotorImagery	BNCI2014-001	BNCI2014-002	BNCI2015-001	BNCI2015-004	PhysionetMotorImagery	Schirrmeister2017	Weibo2014	Zhou2016	Average
ACM+TS+SVM	**86.56±12.26**	**97.32±3.35**	**88.60±10.71**	**93.01±8.09**	**62.60±14.62**	93.33±8.46	98.67±3.06	**93.25±4.12**	**97.18±3.00**	**90.06**
CSP+LDA	77.19±17.58	91.52±10.39	80.98±14.79	88.52±10.75	54.02±11.33	86.41±13.96	97.02±5.17	88.59±6.36	95.20±3.17	84.38
CSP+SVM	78.59±20.14	91.04±10.35	81.21±15.30	89.19±10.08	52.08±11.05	88.04±12.57	97.50±4.90	88.64±5.90	94.95±3.53	84.58
DLCSPauto+shLDA	77.03±18.93	91.54±10.37	80.45±15.52	88.87±10.42	53.02±10.75	86.81±13.34	96.95±5.22	88.48±6.53	94.43±3.41	84.18
DeepConvNet	61.88±19.05	88.27±12.19	87.56±11.25	88.12±13.19	57.08±12.29	71.49±15.88	95.90±7.14	79.29±12.63	95.92±3.66	80.61
EEGITNet	47.50±9.46	75.98±13.09	70.90±17.50	71.95±16.76	51.41±6.40	54.69±11.97	96.04±8.62	62.54±12.32	80.40±17.12	67.93
EEGNeX	52.34±14.81	64.36±13.49	69.95±20.12	72.34±19.83	53.02±9.69	51.77±12.06	89.49±16.91	60.18±11.70	64.80±16.64	64.25
EEGNet-8,2	64.22±16.01	88.55±14.92	83.93±16.31	90.43±11.75	54.20±8.20	73.78±15.59	96.50±8.07	78.15±14.46	94.58±3.21	80.48
EEGTCNet	61.09±22.06	75.21±18.53	73.92±19.02	77.21±18.55	51.22±5.84	57.03±13.25	97.15±7.70	62.37±12.42	85.46±16.42	71.19
FBCSP+SVM	80.78±18.86	93.55±6.29	80.39±16.83	91.57±7.66	52.51±9.82	83.97±12.43	97.40±4.18	88.27±7.91	94.63±3.94	84.78
FgMDM	79.84±17.80	93.52±8.18	84.77±11.26	90.18±9.77	58.31±12.63	89.67±10.65	98.48±3.45	88.56±4.63	96.04±2.67	86.6
MDM	74.22±21.19	89.13±10.38	77.48±14.11	86.20±12.99	48.45±9.62	81.78±11.64	84.67±13.13	65.18±9.75	92.21±4.31	77.7
ShallowConvNet	64.22±18.33	93.00±8.05	87.60±12.05	91.41±10.88	57.23±12.36	74.75±14.98	98.06±4.35	88.70±5.60	97.06±1.86	83.56
TS+EL	81.41±21.36	94.45±6.74	85.98±11.38	91.19±8.49	58.70±13.37	94.09±7.17	98.56±3.01	92.32±3.98	96.59±2.82	88.14
TS+LR	83.75±17.47	94.45±7.06	85.86±11.01	91.09±8.71	61.01±14.22	93.15±7.40	98.60±3.08	91.53±4.53	96.76±2.58	88.47
TS+SVM	82.66±18.16	94.01±7.60	86.19±11.50	90.81±8.95	62.55±15.30	**94.27±7.19**	**98.72±2.92**	91.84±4.25	96.11±2.99	88.57
Average	72.08	88.49	81.61	87.01	55.46	79.69	96.23	81.74	92.02	81.59

Table D4: Summary of performances via average on all the P300 datasets, for classification using all the labels. Intra-session validation. Bold numbers represent the best score in each dataset.

pipeline	BNCI2014-008	BNCI2014-009	BNCI2015-003	BrainInvaders2012	BrainInvaders2013a	BrainInvaders2014a	BrainInvaders2014b	BrainInvaders2015a	BrainInvaders2015b	Cattan2019-VR	EPFLP300	Huebner2017	Huebner2018	Lee2019-ERP	Sosulski2019	Average
ERPCov+MDM	74.30±9.77	81.16±10.13	76.79±10.95	78.77±10.32	80.59±9.36	71.62±11.17	78.57±12.36	80.02±10.07	75.04±15.85	80.76±10.07	71.97±10.88	94.47±8.26	95.15±3.72	74.43±13.26	68.17±13.59	78.79
ERPCov(svdn4)+MDM	75.42±9.91	84.52±8.83	76.93±11.26	79.02±10.53	82.07±8.46	72.11±11.64	76.48±12.83	77.92±10.33	77.09±15.81	80.67±9.47	71.44±10.20	96.21±6.50	96.61±1.89	82.47±12.56	70.63±13.79	79.97
XDAWN+LDA	82.24±5.26	64.03±3.91	78.62±7.19	64.41±4.14	76.74±7.16	66.60±7.54	83.73±10.62	76.02±10.46	77.22±13.73	67.16±6.11	62.98±5.38	97.74±2.84	97.54±1.58	96.45±3.93	67.49±7.44	77.27
XDAWNCov+MDM	77.62±9.81	92.04±5.97	**83.08±7.55**	88.22±5.90	90.97±5.52	80.88±11.01	91.58±10.02	92.57±5.03	83.48±12.05	88.53±7.34	83.20±9.05	98.07±2.09	97.78±1.04	97.70±2.68	86.07±7.15	88.79
XDAWNCov+TS+SVM	**85.61±4.43**	**93.43±5.11**	82.95±8.57	**90.99±4.79**	**92.71±4.92**	**85.77±9.75**	**91.88±9.94**	**93.05±4.98**	**84.56±12.09**	**90.68±6.29**	**84.29±8.53**	**98.69±1.78**	**98.47±0.97**	**98.41±2.03**	**87.28±6.92**	**90.58**
Average	79.04	83.03	79.67	80.28	84.61	75.4	84.45	83.91	79.48	81.56	74.78	97.04	97.11	89.89	75.93	83.08

Table D5: Summary of performances via average on all the P300 datasets, for classification using left vs. right motor imagery task. Intra-session validation. Bold numbers represent the best score in each dataset.

pipeline	Kalunga2016	Lee2019-SSVEP	MAMEM1	MAMEM2	MAMEM3	Nakanishi2015	Wang2016	Average
CCA	25.40±2.51	23.86±3.72	19.17±5.01	23.60±4.10	13.80±7.47	8.15±0.74	2.48±1.01	16.64
MsetCCA	22.67±4.23	25.10±3.81	20.50±2.37	22.08±1.76	27.60±3.01	7.10±1.50	4.00±1.10	18.43
MDM	**70.89±13.44**	75.38±18.38	27.31±11.64	23.12±6.29	34.40±9.96	78.77±19.06	54.77±21.95	52.09
TS+LR	70.86±11.64	**89.44±13.84**	**53.71±24.25**	**39.36±12.06**	**42.10±14.33**	**87.22±15.96**	**67.52±20.04**	**64.32**
TS+SVM	68.95±13.73	88.58±14.47	50.58±23.34	34.80±11.76	40.20±14.41	86.30±15.88	59.58±20.57	61.28
TRCA	24.84±7.24	64.01±15.27	24.24±6.65	24.24±2.93	23.70±3.49	83.21±10.80	2.79±1.03	35.29
Average	47.27	61.06	32.58	27.87	30.3	58.46	31.86	41.34

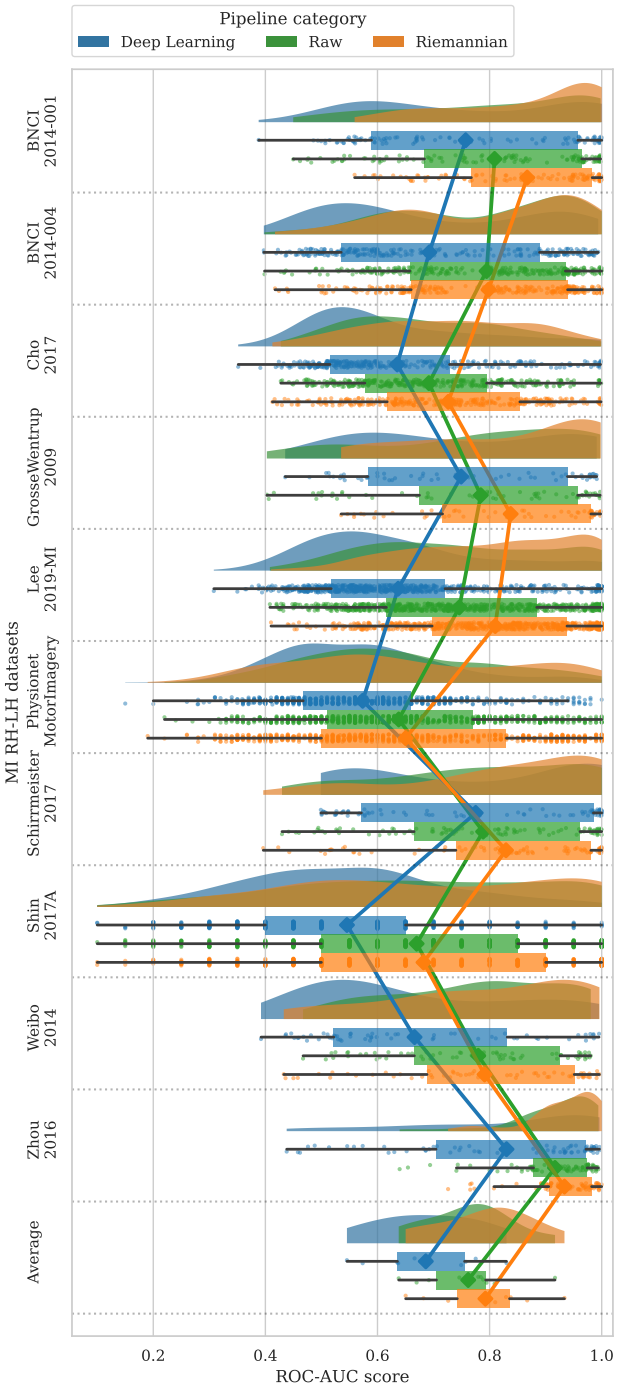


Figure C1: Distributions of ROC-AUC scores on the right hand vs left hand MI task of the pipelines grouped by category.

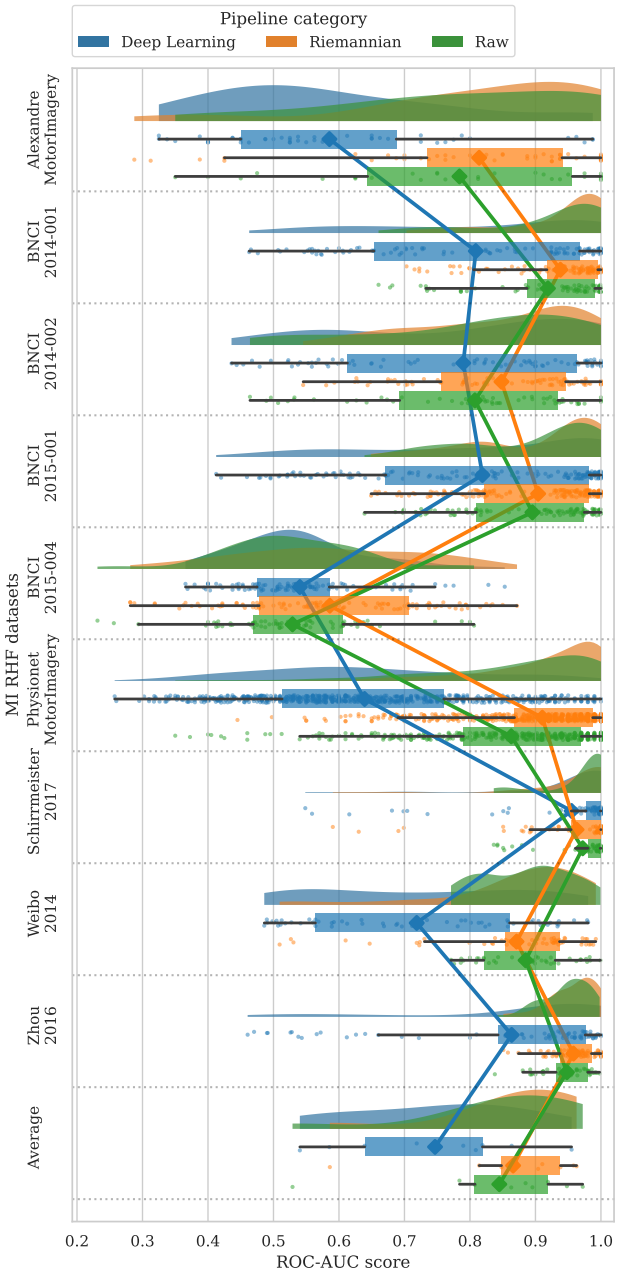


Figure C2: Distributions of ROC-AUC scores on the right hand vs feet MI task.

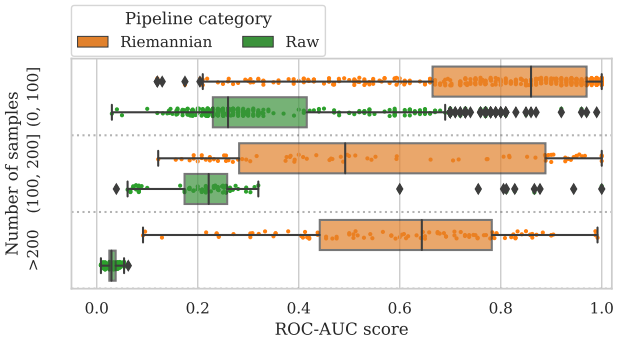


Figure C3: Accuracy scores averaged over all subjects and session of each SSVEP dataset, per pipeline category.

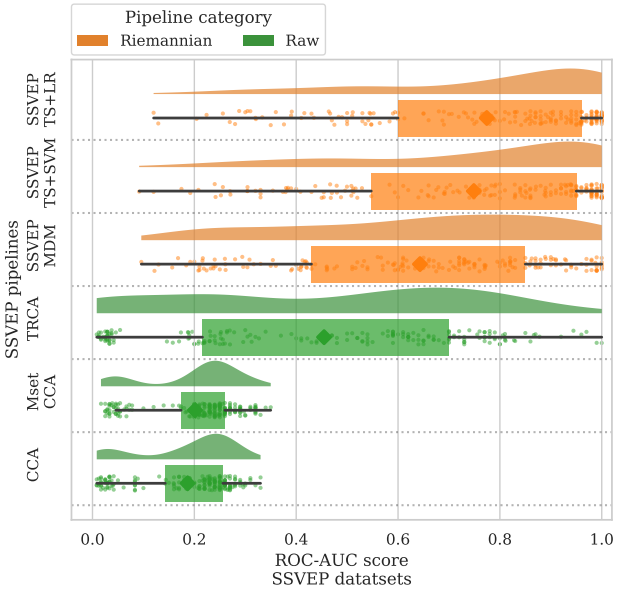


Figure C4: Box-plot representing classification accuracy averaged over all the sessions of all the subjects of all the datasets of the SSVEP paradigm and over all pipelines of the corresponding category (*Riemannian, Raw*). Box-plots are overlaid with strip-plots, where each point represents the classification accuracy of one within-session evaluation.

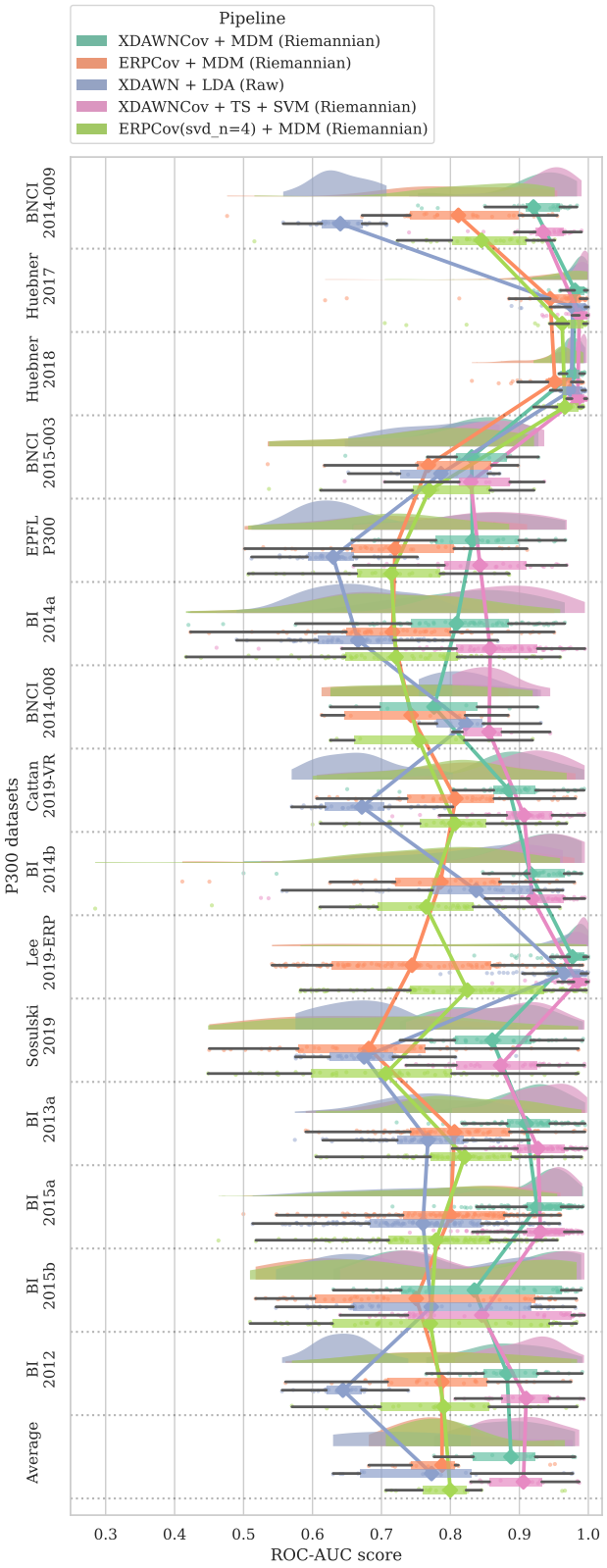


Figure C5: ROC-AUC scores on the ERP classification task of the different pipelines.