Proceedings of the
8th Graz Brain-Computer Interface Conference 2019 DOI: 10.3217/978-3-85125-682-6-07

INTERPRETABLE RIEMANNIAN CLASSIFICATION IN BRAIN-COMPUTER INTERFACING

Jiachen Xu[1,2], Moritz Grosse-Wentrup[1,3,4], Vinay Jayaram[1]

[1]Max Planck Institute for Intelligent Systems, Department of Empirical Inference, Tübingen, Germany
[2]Technical University of Munich, Department of Electrical and Computer Engineering, Munich, Germany
[3]Ludwig Maximilian University of Munich, Department of Statistics, Munich, Germany
[4]University of Vienna, Faculty of Computer Science, Vienna, Austria

E-mail: {jiachen.xu, moritzgw, vjayaram}@tuebingen.mpg.de

ABSTRACT: Riemannian methods are currently one of the best ways of building classifiers for EEG data in a brain-computer interface (BCI). However, they are computationally complex and suffer from a lack of interpretability. Since the full covariance matrix is used for each classification, it is not immediately possible to see what underlying signals are generating the classified changes in variance. Particularly in a rehabilitation context, where it is essential to control which brain signals are used for classification, this can be a severely limiting factor. Further, the requirement to perform a matrix logarithm can become prohibitively complex for real-time computation. In this work, we explore a method for extracting spatial filters from a solution in the Riemannian tangent space and compare it against common spatial patterns. We show via comparisons on multiple open-access datasets that it is possible to generate filters that approach the performance of the full Riemannian solution while maintaining interpretability.

INTRODUCTION

One of the major reasons that spatial filtering has enjoyed decades of popularity within the field of brain-computer interfacing is its interpretability. In contrast to most other methods, spatial filters can be directly evaluated to see whether they represent sources of brain activity or correlated noise sources, which is crucial in the fields of rehabilitation and neurofeedback. As they serve the dual purposes of dimensionality reduction and explainability, there has been a large amount of research on how to calculate optimal ones for a given task. From the original paper on common spatial patterns (CSP) [1] to modern divergence-based approaches [2], hundreds of manuscripts have been written on how to find them.
In parallel to this development, in the last few years, there has been an upsurge of interest in methods based on Riemannian methods [3, 4]. By considering the sensor covariance matrix as a whole, instead of trying to isolate particular sources of interest, it is possible to overcome many of the non-stationarity issues that have plagued machine learning in EEG, resulting in tangent-space methods out-performing all others in a recent meta-analysis [5]. In contrast to the often complex optimization methods required for modern approaches to spatial filtering, and the uncertain gains they provide, Riemannian tangent space methods are amenable to standard machine learning algorithms, which significantly increases their ease of use. The downside is that, by using the entire covariance matrix, it is no longer possible to determine what exact sources are generating a difference between experimental conditions. This is of little concern in consumer BCIs, where the goal of the device is usability for healthy subjects, but in the case of neurofeedback or patient groups, this is a severely limiting factor. Proof that a brain signal, and not a correlated physiological signal such as muscular activity, is the driving force for a BCI is necessary for these contexts, and so it is not yet possible to use Riemannian methods there.
Our contribution is the following: Based on recent findings [6], we validate a method for decomposing a linear function in the Riemannian tangent space into a set of spatial filters. We show in tests on multiple datasets that these spatial filters approach the performance of the full Riemannian method and further that they are more robust than CSP to different numbers of filters used. We also show associated spatial patterns to validate that the filters capture relevant brain regions. In summary, our contributions render Riemannian classification methods applicable to BCIs for rehabilitation and neurofeedback.

METHODS

BCI paradigms commonly influence the amplitude of brain rhythms. To create better features, spatial filters have been used to isolate the component of the recorded signal corresponding to a rhythm of interest. These filters are often found by maximizing variance differences, as the variance is a mathematically convenient proxy for amplitude.
One of the most representative and common data-driven spatial filters is CSP [1, 7], which is computed by find-

 DOI: 10.3217/978-3-85125-682-6-07

ing the sources that maximize the variance ratio between two conditions. CSP has made a great contribution to the stable increase of BCI's performance and has over the years spawned many improved versions, such as filter bank CSP [8], regularized CSP [9] and divergence-based CSP [2]. In addition to increased decoding performance, another important reason behind the popularity of CSP is its interpretability. By observing the spatial patterns associated with discovered spatial filters, it is possible to verify whether they are related to neural or other sources. Unlike spatial filtering, which uses variance-based features, Riemannian methods are based on directly comparing sensor covariance matrices. In this paper, the adopted Riemannian metric is called the affine-invariant Riemannian metric (AIRM) [10], where the "affine-invariance" means that the distance between two symmetric positive definite (SPD) matrices is invariant to the distance after applying an affine transformation to both matrices [11], i.e.,

$$\delta_R(\mathbf{TC}_1, \mathbf{TC}_2) = \delta_R(\mathbf{C}_1, \mathbf{C}_2) = \left\| \mathrm{Log}\left(\mathbf{C}_1^{-1}\mathbf{C}_2\right) \right\|_F, \quad (1)$$

where $\|\cdot\|_F$ represents the Frobenius norm, $\mathrm{Log}(\cdot)$ means the logarithm of a matrix, $\mathbf{C}_{(.)}$ are SPD matrices, and $\mathbf{T}$ is any affine transformation matrix. While this method returns a very robust estimate of distance, most machine learning algorithms require a Euclidean space to work. This can be done via projecting manifold points onto the tangent space to a given point.

The tangent space is defined as a real vector space centered around a reference point on the manifold. In practice, this point is normally selected as the mean point across all trials [10]. After fixing the reference point, the covariance matrices on the manifold are mapped onto this space using the logarithmic map (Eq. 2) and then vectorized into the tangent vectors (TVs).

$$\mathrm{Log}_{\mathbf{C_1}}(\mathbf{C}_2) = \mathbf{C}_1^{\frac{1}{2}} \mathrm{Log}\left(\mathbf{C}_1^{-\frac{1}{2}}\mathbf{C}_2\mathbf{C}_1^{-\frac{1}{2}}\right)\mathbf{C}_1^{\frac{1}{2}}, \quad (2)$$

$$\mathrm{Exp}_{\mathbf{C_1}}(\mathbf{C}_2) = \mathbf{C}_1^{\frac{1}{2}} \mathrm{Exp}\left(\mathbf{C}_1^{-\frac{1}{2}}\mathbf{C}_2\mathbf{C}_1^{-\frac{1}{2}}\right)\mathbf{C}_1^{\frac{1}{2}}, \quad (3)$$

where $\mathrm{Exp}(\cdot)$ means the exponential of a matrix. Eq. 2 represents the logarithm mapping of $\mathbf{C}_2$ from the manifold to the TS based on the point $\mathbf{C}_1$ and Eq. 3 shows the inverse projection, i.e., the exponential mapping. For the affine-invariant and log-Euclidean metrics, the logarithmic map requires computing the matrix logarithm of its argument, as a way of mapping the eigenvalues of the input matrices from the strictly positive space of SPD matrices to a real-valued space. As the tangent space represents a linearization of the manifold, distances between points approximate true manifold distances, and geometry on the manifold is preserved, which allows for linear machine learning methods to be used.

A major drawback of this procedure is the difficulty in interpreting the resulting classifier because the tangent space embedding cannot be easily linked to any intuitive physical process. We address this problem by introducing a procedure to extract spatial filters from a linear function in the tangent space, which allows for a unified methodology in the computation of spatial filters that can also take advantage of the large body of work on machine learning in linear spaces.

The procedure can be summarized in three steps: Classification, spatial filter generation, and filter selection. For classification, we follow the approach in [12] (linked to point 2 to point 5 in Algorithm 1, abbreviated as p2 to p5 afterwards) and find the optimal SVM solution in the affine-invariant tangent space (p6). Though any algorithm is possible, we chose a regularized SVM as maximum-margin classifiers can deal well with the high dimensionality of the tangent space. Once a linear classifier is computed, we project the weight vector in the tangent space back to the SPD manifold via the exponential map (Eq. 3) (p8 to p9). From here, we compute spatial filters by taking the solution to the generalized eigenvalue (GE) problem between the tangent space reference point and the exponential map of the weight vector (p10).

Algorithm 1 Extraction of Tangent Space Spatial Filters

INPUT: Band-pass filtered raw signal $\mathbf{X}_i,\ \forall i = 1, ..., T$
OUTPUT: Spatial filter after ordering $\mathbf{F}$

1: **procedure** I. CLASSIFICATION
2: $\mathbf{C}_i = \mathbf{X}_i\mathbf{X}_i^T\ \forall i = 1, ..., T$
3: $\mathbf{C_m} = \underset{\mathbf{C}\in C(n)}{\arg\min} \sum_{i=1}^{T} \delta_R^2(\mathbf{C}, \mathbf{C}_i)$ {AIRM mean}
4: $\mathbf{S}_i = \mathrm{Log}_{\mathbf{C_m}}(\mathbf{C}_i)$ {Mapping from manifold to TS}
5: $\overrightarrow{S}_i = \mathrm{vec}(\mathbf{S}_i)$ {Vectorize the upper triangular matrix}
6: $\overrightarrow{W} = \mathrm{SVM}\left(\overrightarrow{S}_i, y_i\right)$ {Weight vector on the TS}
7: **procedure** II. SPATIAL FILTER GENERATION
8: $\mathbf{S}_w = \mathrm{devec}\left(\overrightarrow{W}\right)$ {Convert the vector to SPD matrix}
9: $\mathbf{C}_w = \mathrm{Exp}_{\mathbf{C_m}}(\mathbf{S}_w)$ {Mapping from TS to manifold}
10: $\mathbf{C}_w\mathbf{V} = \mathbf{C_m}\mathbf{V}\Gamma$ {Mapping from TS to manifold}
11: **procedure** III. FILTER SELECTION
12: $\widetilde{\mathbf{X}}_i = \mathbf{V}^T\mathbf{X}_i$ {Filter raw signal using full components}
13: $\overrightarrow{f_i} = \log(\mathrm{var}(\widetilde{\mathbf{X}}_i))$ {Compute the log-variance}
14: $\overrightarrow{\beta} = \mathrm{LinearRegression}\left(\overrightarrow{f_i}, y_i\right)$ {Weight vector of fitted classifier}
15: $\overrightarrow{id} = \mathrm{sort}\left(\left|\overrightarrow{\beta}\right|\right)$ {Sort the absolute value of weight coeficient in descending order}
16: $\mathbf{F} = \mathbf{V}\left[:, \overrightarrow{id}\right]$ {Get the re-ordered spatial filter}

The eigenvectors that solve a GE problem simultaneously diagonalize both input matrices, which can be interpreted as finding a space where the weight SPD matrix and the input SPD matrices are all roughly diagonal. This independence means that the logarithm of the matrices in this space is close to the log-variances of each of the filtered dimensions, allowing us to approximate the tangent space linear function as a linear function of the log-variances

 DOI: 10.3217/978-3-85125-682-6-07

Table 1: Summary of all MOABB datasets with left-hand versus right-hand motor imagery. The dataset marked in red color has only 3 channels and is hence excluded from this analysis.

Dataset Name	#Channels	#Subjects	#Sessions
BNCI 2014-001	22	9	2
BNCI 2014-004	3	9	5
Cho et al. 2017	64	49	1
Munich Motor Imagery	128	10	1
Physionet Motor Imagery	64	109	1
Shin et al. 2017	25	29	3
Weibo et al. 2014	60	10	1
Zhou et al. 2016	14	4	3

in the filtered space. However, the associated eigenvalues have no relationship to the linear function, requiring another criterion in order to determine which of the computed filters are most relevant to classification. In order to do this, we fit a linear function to the log-variances of our projected training data (12 to 14) and attempt to reconstruct the function values of the true tangent space function. Based on the absolute values of the coefficients from this regression problem, we rank the GE solution filters (15 to 16).

To evaluate the performance of our method in a rigorous and reproducible fashion, we adopted the datasets and framework from the open source benchmark *Mother of all BCI Benchmark (MOABB)* in our work [5]. For simplicity, we choose left- versus right-hand motor imagery, which is one of the most common paradigms in EEG-based BCIs. We restrict our analysis to the α- and β-bands (8Hz $\sim$ 32Hz). We use all available channels except those used for electrooculography (EOG). We exclude the dataset *BNCI 2014-004* from our analysis (marked in red in Table 1), because this data set has only three channels.

After bandpass filtering the raw EEG signal, the corresponding covariance matrices are estimated via the empirical covariance estimator. Afterward, the spatial filter is extracted according to the pseudocode shown in Algorithm 1, and the classifier is fit using the log-variances of the filtered signal. As for the classification algorithm, we choose Fisher's Linear Discriminant Analysis (LDA) for simplicity. Decoding accuracy is estimated via five-fold cross-validation within each data set.

Two statistical parameters are utilized to compare our method against CSP and the full Riemannian approach: the p-values and the effect sizes. The p-value for the one-sided test is computed within each dataset, whose null hypothesis is that the median accuracy of using CSP is not larger than that of applying the spatial filter designed by us. Effect sizes show the standardized mean difference (SMD) between the accuracies of the two compared methods. All statistical tests are implemented as described in [5].

RESULTS

Meta-analysis:

As a spatial filtering technique, the natural comparison for TS SF is against CSP. Therefore, we compared them under three typical scenarios: applying very few filters (for speed), using filters corresponding to half the number of channels (for the compromise between speed and accuracy) and using all filters (for accuracy). To ensure the results are rigorous, the comparison is done across seven datasets from MOABB [5] (as shown in Table 1).

When merely applying two spatial filters (Fig. 1a), TS SF is generally better than CSP, but the positive significance is only present in two datasets: *Shin* and *Munich*. After increasing the filter number to the half the channel dimensionality (Fig. 1b), the positive significance appears in three more datasets *Weibo*, *Physionet* and *Cho*, in addition to an increased meta-effect. Moreover, the SMD of data set *Munich* has increased to 1.5 which implies a strong effect. Lastly, when using all filters (Fig. 1c), the effect sizes are lessened in the datasets with significant performance, except for *Weibo*.

Classification accuracy w.r.t. the number of applied filters:

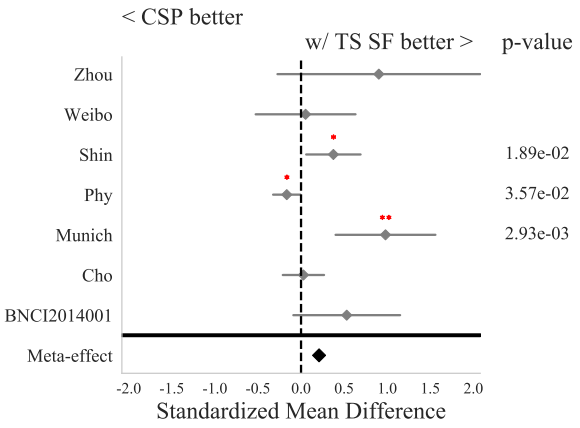
In addition to comparing against CSP, we are also interested in seeing how well TS SF can approximate the accuracy of the full tangent space classifier, as well as how this approximation evolves as a function of the number of filters. Therefore, we choose three representative datasets *Munich*, *Cho*, *Physionet* and plot their accuracy w.r.t. the number of applied filters in Fig. 2, showing the tangent space accuracy as well as an upper bound.

When using data set *Munich* (Fig. 2a), TS SF (green line) first outperforms CSP (red line) when the number of applied filters is less than 16. Afterward, their accuracies are tied with each other until using more than 40 filters. Subsequently, TS SF maintains its stable performance, but the accuracy of CSP starts to decrease.

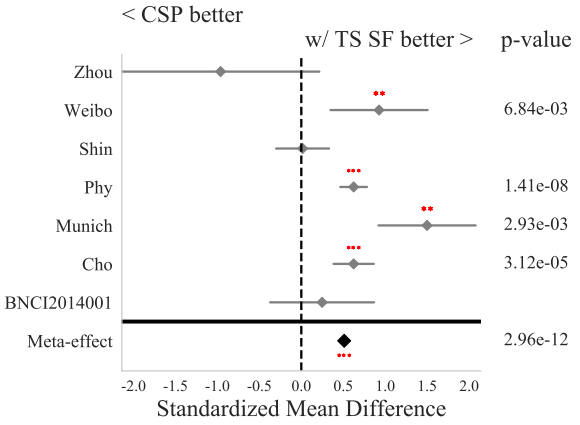
The information conveyed by Fig. 2b and Fig. 2c is quite similar to each other. TS SF and CSP perform very comparably up until around 13 filters are used. After this point, using more CSP components hurts the performance, while the performance stays much more stable when using more TS SF filters. However, in Fig. 2c, the accuracy drops when using TS SF at high numbers of filters.

Overall, the performance of the full Riemannian method (Full TS, brown line) is always superior to both spatial filtering methods because the full approach utilizes the full information decoded in the covariance matrices.

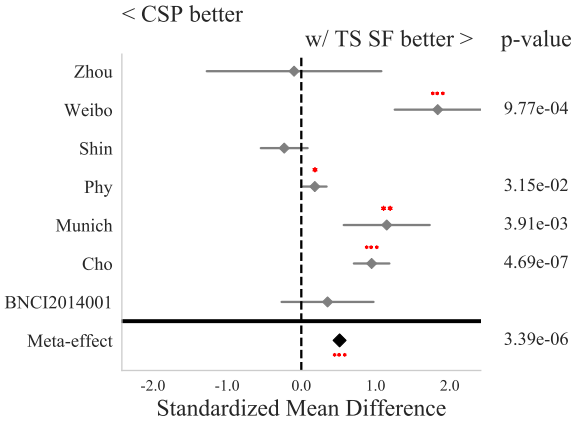
 DOI: 10.3217/978-3-85125-682-6-07



(a) Applying the first two filters



(b) Applying the first half of all filters



(c) Applying all filters

Figure 1: Meta analysis of accuracy after applying CSP or TS SF as spatial filter. Parameters: p-value p and SMD are computed within each dataset. Red *, ** and *** represent $p < 0.05, 0.01, 0.001$ respectively. Grey diamonds signify the SMD, while grey bars show the confidence interval of the mean.

Spatial patterns:
From Fig. 1 and Fig. 2 we know that TS SF on average outperforms CSP in terms of accuracy. For a spatial filtering method, however, its interpretation is as crucial as its classification accuracy. Therefore, we compared the corresponding spatial patterns of both TS SF and CSP in order to see what sources were chosen. We choose the data set *Munich* because it has the best overall data quality [5] and highest channel number (128), and plot the two best filters using both CSP and TS SF. In order to convert spatial filters into spatial patterns, we use the approach proposed by [13].

When comparing Fig. 3a and 3b there are some interesting results. First, the spatial patterns from CSP tend to be more patchy than from TS SF, e.g., S1, S2, S10, in particular for those subjects noted as having low data quality [14]. Second, for the subjects who have excellent signal quality, the spatial patterns from TS SF are highly consistent with from CSP, e.g., S3, S4, and S6. Third, when looking at S2, the pattern from TS SF is more interpretable comparing to the patchy pattern from CSP. When solely focusing on Fig. 3b, it is obvious to see that for S2, S3, S4, S5 and S6, they represent strong signal sources surrounding the sensorimotor cortex which is consistent with this paradigm. This is in contrast to, e.g., S7 and S10, where CSP clearly picks up ocular artifacts.

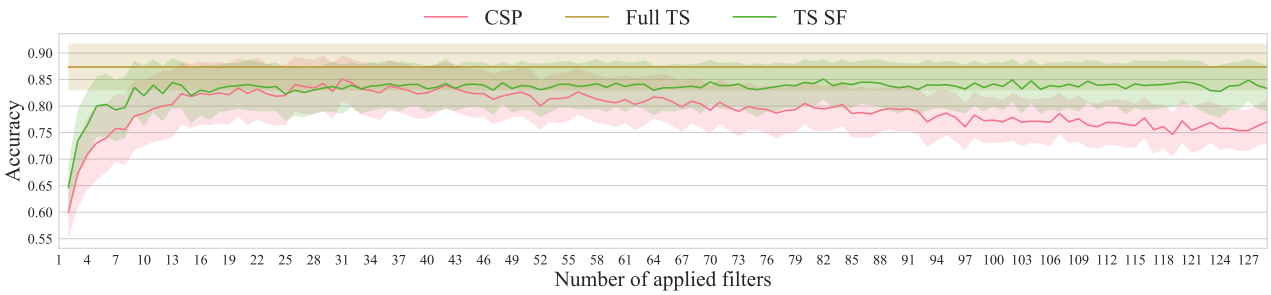
DISCUSSION

We show that it is possible to construct spatial filters that can approximate the performance of tangent space linear functions, and validate this result on a variety of open-source datasets. In our simulations, it can be seen that the TS SF out-performs CSP, in particular in the cases where the full tangent space function out-performs CSP. Further, we show in the *Munich* dataset that the spatial filters recovered by TS SF are more robust to noisy data than those from CSP. With these advances, it is possible to use Riemannian methods in neurofeedback and clinical work. To evaluate the spatial filter, the first two essential indicators are the classification accuracy and the quality of spatial patterns.

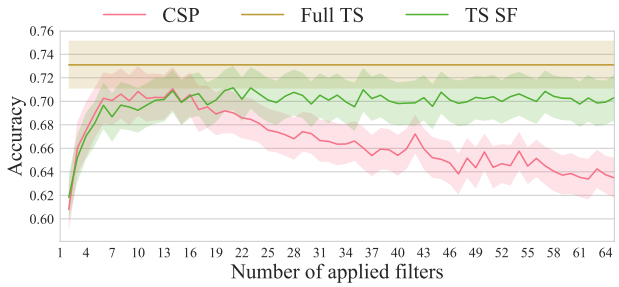
From the perspective of performance, TS SF is still imperfect. First, as shown in Fig. 2, TS SF is constantly worse than the full Riemannian methods. Hence, to move forward, it is important to diminish the approximation error as much as possible, especially when using very few filters. The choice of how to determine filter relevance, though effective, can also be improved. Regularized linear regression approaches, such as LASSO or ridge regression, may better isolate good filters, or using another target variable.

The spatial patterns associated with TS SF are robust to artifacts. As shown in Fig. 3, the reflected neural signal sources after filtering by TS SF is more reasonable and interpretable than by CSP, especially for the low-quality signal. For example, for S2 of the data set *Munich* where the percentage of trials contaminated by artifacts reaches 55% [14], CSP merely collects the artifacts whereas TS SF presents the strong sources around sensorimotor cortex which is consistent with the chosen paradigm. Nonetheless, for the subjects with high data quality, TS SF is still as good as CSP (e.g., for S3 and S4,

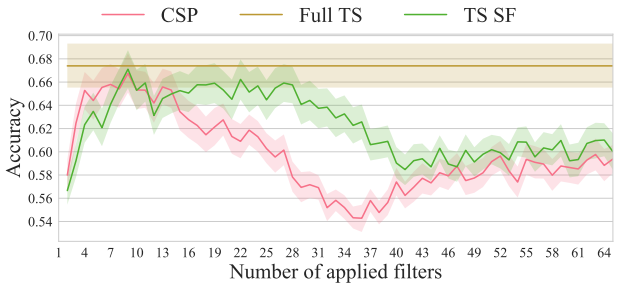
 DOI: 10.3217/978-3-85125-682-6-07



(a) Dataset: Munich MI



(b) Dataset: Cho.



(c) Dataset: Physionet

Figure 2: Accuracies comparison among three different pipelines and for each number of applied filters, the central line and the error band are computed across all subjects and sessions. **1. CSP**: Choose CSP as the spatial filter and classified by LDA. **2. Full TS**: Use the standard Riemannian classification framework with AIRM without any spatial filtering and classified by SVM. **3. TS SF**: Apply the spatial filters which are extracted from the tangent space and classified by LDA. Error band setting: confidence interval = 68%. Central line: the mean accuracy. It is not difficult to notice that TS SF has a better and more robust performance than CSP w.r.t. the increase of the number of applied filters.

where 5.0% and 6.3% of trials are affected by artifacts respectively [14]). Therefore, as far as the interpretability is concerned, TS SF is more robust than CSP.

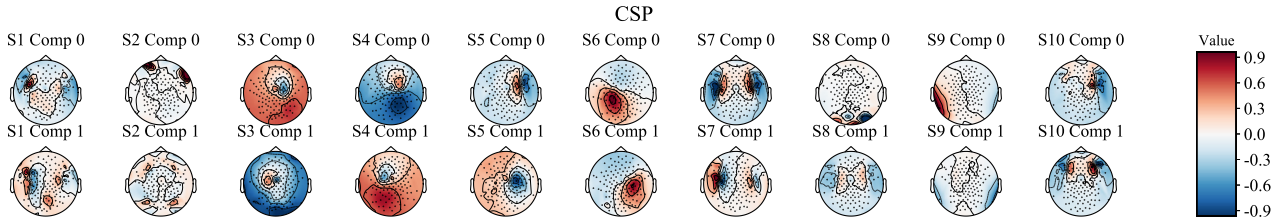
As shown in Fig. 2, it is obvious that TS SF is always more robust than CSP considering the tendency of the accuracy w.r.t. the filter numbers. However, when focusing at Fig. 2c, we could find an evident fluctuation for TS SF. Accuracies drop at high numbers of filters in contrast to the other datasets, and the overall range of mean accuracies is far lower as well. Much of this can likely be explained by the low data quality of data set *Physionet*, which would lead to worse solutions of the initial tangent space problem that likely translate into poorer spatial filters. However, it is important to understand what conditions lead to better or worse TS SF solutions, especially when applying full filters. Also, it should be explored more that why is TS SF still more robust than CSP to filter number even when TS SF performs badly.

Usually, the more robust performance demand more complex computation, but this argument does not always hold. Although TS SF requires a tangent space function to be computed as an initial step which means the computation of the filters themselves is far more intensive than that of CSP, however, in the online application, the procedure is identical: the data is filtered via a linear projection and then the log variances of the resulting time series are used as features for a final classifier. When comparing to the standard Riemannian classification framework, TS SF leads to an approximation error. However, the computation for online classification would be significantly reduced, especially with large channel numbers. Given that the loss in accuracy on average is relatively small, this may represent a feasible way of using Riemannian methods in high-dimensional, online settings, and further of using them to increase the efficacy of clinical neurofeedback.

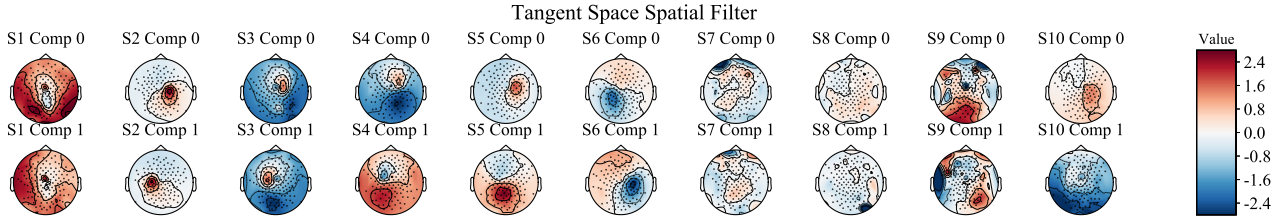
CONCLUSION

In this paper, we propose a new method to generate spatial filters from the Riemannian classification framework, Tangent Space Spatial Filters (TS SF), and after comparing between CSP and TS SF, we found that TS SF has better and more robust performance than CSP with the same computational cost online. By extracting spatial filters, the Riemannian approach becomes more interpretable without much loss of accuracy.
However, we still do not have a strong mathematical argument for why this simplification works and the interpretation of the weight vector after projection to the SPD manifold. Nonetheless, the undeniable fact is that the performance of TS SF reveals to us the potential of interpretable Riemannian approaches in BCIs.

 DOI: 10.3217/978-3-85125-682-6-07



(a) Applying CSP as spatial filter



(b) Applying tangent space spatial filter (TS SF) as spatial filter

Figure 3: The spatial patterns of data set *Munich* associated with applying the first two spatial filters.

REFERENCES

[1] Zoltan J Koles, Michael S Lazar, and Steven Z Zhou. “Spatial patterns underlying population differences in the background EEG”. In: *Brain Topography* 2.4 (1990), pp. 275–284.

[2] Wojciech Samek, Motoaki Kawanabe, and Klaus-Robert Müller. “Divergence-based framework for common spatial patterns algorithms”. In: *IEEE Reviews in Biomedical Engineering* 7 (2014), pp. 50–72.

[3] Marco Congedo, Alexandre Barachant, and Rajendra Bhatia. “Riemannian geometry for EEG-based brain-computer interfaces; a primer and a review”. In: *Brain-Computer Interfaces* 4.3 (2017), pp. 155–174.

[4] Florian Yger, Maxime Berar, and Fabien Lotte. “Riemannian approaches in brain-computer interfaces: a review”. In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 25.10 (2017), pp. 1753–1762.

[5] Vinay Jayaram and Alexandre Barachant. “MOABB: Trustworthy algorithm benchmarking for BCIs”. In: *Journal of Neural Engineering* (2018).

[6] Alexandre Barachant, Jason Carmel, Kathleen Friel, and Disha Gupta. “Extraction of motor patterns from joint EEG/EMG recording: A Riemannian Geometry approach”. In: *6th International Brain-Computer Interface Meeting*. 2016.

[7] Herbert Ramoser, Johannes Muller-Gerking, and Gert Pfurtscheller. “Optimal spatial filtering of single trial EEG during imagined hand movement”. In: *IEEE Transactions on Rehabilitation Engineering* 8.4 (2000), pp. 441–446.

[8] Kai Keng Ang, Zheng Yang Chin, Chuanchu Wang, Cuntai Guan, and Haihong Zhang. “Filter bank common spatial pattern algorithm on BCI competition IV datasets 2a and 2b”. In: *Frontiers in Neuroscience* 6 (2012), p. 39.

[9] Fabien Lotte and Cuntai Guan. “Regularizing common spatial patterns to improve BCI designs: unified theory and new algorithms”. In: *IEEE Transactions on Biomedical Engineering* 58.2 (2011), pp. 355–362.

[10] Xavier Pennec, Pierre Fillard, and Nicholas Ayache. “A Riemannian framework for tensor computing”. In: *International Journal of Computer Vision* 66.1 (2006), pp. 41–66.

[11] Xavier Pennec and Nicholas Ayache. “Uniform distribution, distance and expectation problems for geometric features processing”. In: *Journal of Mathematical Imaging and Vision* 9.1 (1998), pp. 49–67.

[12] Alexandre Barachant, Stéphane Bonnet, Marco Congedo, and Christian Jutten. “Riemannian geometry applied to BCI classification”. In: *International Conference on Latent Variable Analysis and Signal Separation*. Springer. 2010, pp. 629–636.

[13] Stefan Haufe et al. “On the interpretation of weight vectors of linear models in multivariate neuroimaging”. In: *Neuroimage* 87 (2014), pp. 96–110.

[14] Moritz Grosse-Wentrup, Christian Liefhold, Klaus Gramann, and Martin Buss. “Beamforming in noninvasive brain–computer interfaces”. In: *IEEE Transactions on Biomedical Engineering* 56.4 (2009), pp. 1209–1219.