

# Feature Engineering: Motor Imagery Electroencephalogram

Ethan Davis

January 2026

## 1 Common Spatial Pattern

The common spatial pattern (CSP) allows one to maximize the variance of signals from one condition and at the same time minimize the variance of signals from another condition [Bla18]. For example, consider two three-dimensional tensors (trials  $\times$  channels  $\times$  time) of electroencephalogram (EEG) from separate classes  $\mathbf{X}_1, \mathbf{X}_2 \in \mathbb{R}^{p \times q \times r}$ . For each trial let us assume zero mean and compute between channel across time sample covariance matrices (SCMs)  $\Sigma_i^{(k)} = \frac{1}{r-1} \mathbf{X}_i^{(k)} \mathbf{X}_i^{(k)\top} \in \mathbb{R}^{q \times q}$  for  $i = 1, 2$  and  $k = 1, \dots, p$ . Furthermore, let us take  $\bar{\Sigma}_i \in \mathbb{R}^{q \times q}$  as the arithmetic mean SCM for each class across all trials  $k$ .

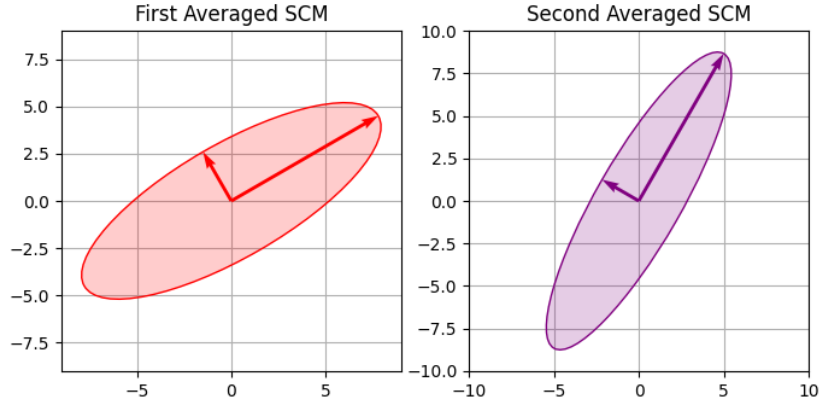


Figure 1: Example of averaged sample covariance matrices (SCMs).

Our example is a standard data and preprocessing pipeline for CSP used by software libraries like PyRiemann [Bar+25]. We assume the SCMs are conditioned so that they are symmetric positive definite (SPD). By definition, for a matrix  $\mathbf{A} \in \mathbb{R}^{n \times n}$  and any non-zero vector  $\mathbf{x} \in \mathbb{R}^n$ , if  $\mathbf{A} = \mathbf{A}^\top$  and  $\mathbf{x}^\top \mathbf{A} \mathbf{x} > 0$ , then  $\mathbf{A}$  is SPD. As a result, all eigenvalues are positive  $\lambda_i > 0$  from the eigendecomposition  $\mathbf{A} = \mathbf{Q} \mathbf{D} \mathbf{Q}^\top$  where  $\mathbf{D} = \text{diag}(\lambda_1, \dots, \lambda_n)$  [Mur22]. Furthermore, the sum of two SPD matrices  $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times n}$  is also SPD since  $\mathbf{x}^\top (\mathbf{A} + \mathbf{B}) \mathbf{x} = \mathbf{x}^\top \mathbf{A} \mathbf{x} + \mathbf{x}^\top \mathbf{B} \mathbf{x} > 0$ . Let us diagonalize the mean SCMs from our example.

$$\bar{\Sigma}_1 + \bar{\Sigma}_2 = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^\top. \quad (1)$$

The whitening transformation of an SCM results in zero mean, unit variance, and zero covariance [Bla18]. In other words, the whitening transformation maps an SCM to the identity matrix  $\Sigma \mapsto \mathbf{I}$ . We define the whitening matrix as  $\mathbf{W} = \mathbf{V} \mathbf{\Lambda}^{-1/2} \mathbf{V}^\top \in \mathbb{R}^{n \times n}$ . The result is no longer an ellipse when plotted but rather the unit circle.

$$\mathbf{W}(\bar{\Sigma}_1 + \bar{\Sigma}_2) \mathbf{W}^\top = \mathbf{V} \mathbf{\Lambda}^{-1/2} \mathbf{V}^\top (\mathbf{V} \mathbf{\Lambda} \mathbf{V}^\top) \mathbf{V} \mathbf{\Lambda}^{-1/2} \mathbf{V}^\top = \mathbf{I}. \quad (2)$$

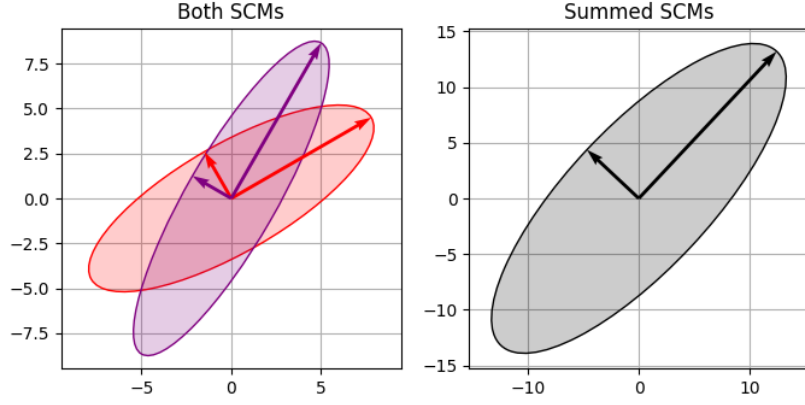


Figure 2: The averaged SCMs summed.

We can rewrite our whitening transformation to show that the variance of the signals in our first class  $\bar{\Sigma}_1$  is maximized, while the variance of the signals in our second class  $\bar{\Sigma}_2$  is also minimized. We do this by distributing the whitening matrix throughout the sum of our SCMs. The result is two transformed SCMs we denote by  $\bar{\Sigma}'_1, \bar{\Sigma}'_2 \in \mathbb{R}^{n \times n}$  whose sum is the identity matrix.

$$\mathbf{W}(\bar{\Sigma}_1 + \bar{\Sigma}_2)\mathbf{W}^\top = \mathbf{W}\bar{\Sigma}_1\mathbf{W}^\top + \mathbf{W}\bar{\Sigma}_2\mathbf{W}^\top = \bar{\Sigma}'_1 + \bar{\Sigma}'_2 = \mathbf{I}. \quad (3)$$

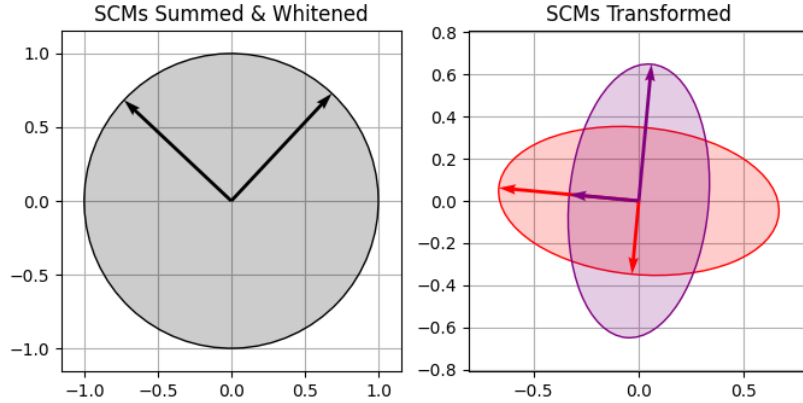


Figure 3: Both the whitened sum of the SCMs and the sum of each SCM transformed by the whitening matrix equals the identity matrix.

Let us use  $\mathbf{V}'_i \mathbf{\Lambda}'_i \mathbf{V}'_i{}^\top$  where  $i = 1, 2$  to denote the diagonalization of our transformed SCMs. Since  $\bar{\Sigma}'_1 + \bar{\Sigma}'_2 = \mathbf{I}$  the sum of the diagonals  $d_{ij} = \text{diag}(\mathbf{\Lambda}'_i)_j$  is  $d_{1j} + d_{2j} = 1$ . Furthermore,  $d_{ij} > 0$  since  $\bar{\Sigma}'_i$  is SPD. Therefore, the diagonals are bounded between 0 and 1. When the variance  $d_{aj} \gg d_{bj}$  for  $a \neq b$ , the classification problem is more or less solved for unseen EEG trials. However, when  $d_{aj}$  is close to  $d_{bj}$  the discrimination between classes is more ambiguous [Bla18].

## 2 Tangent Space

Current state-of-the-art (SOTA) machine learning (ML) for electroencephalogram (EEG) uses across channel covariance matrices and Riemannian geometric statistics on the symmetric positive definite (SPD) manifold for classification [Bar+12; Che+18]. These methods either discriminate directly on the SPD manifold or in the vector space that is tangent to the manifold. In this section we analyze the latter method of discrimination that performs classification in the tangent space (TS) of the SPD manifold.

### 2.1 Definitions

A manifold is a collection of points that locally, but not globally, resembles Euclidean space. A Riemannian metric is defined by a smoothly varying collection of scalar products  $\langle \cdot, \cdot \rangle_x$  in each tangent space  $T_x\mathcal{M}$  at points  $x$  on a manifold  $\mathcal{M}$ . For example, the inner product gives a norm  $\|\cdot\|_x: T_x\mathcal{M} \rightarrow \mathbb{R}$  by  $\|v\|_x^2 = \langle v, v \rangle_x$ . The shortest distance between two points on a manifold is a geodesic  $\gamma(t)$ , and it is computed by integrating the norm along its curve [SFP20].

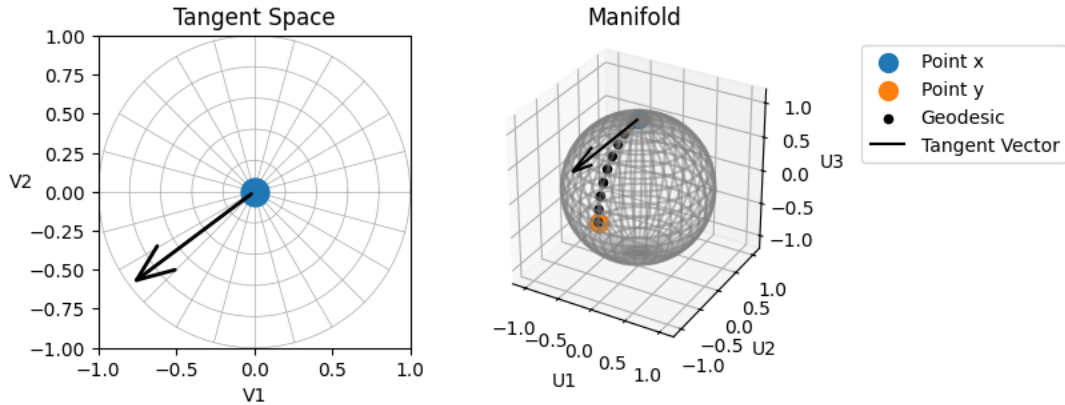


Figure 4: Example of a tangent space, manifold, and geodesic. (Left) The tangent space  $T_x\mathcal{M}$  that is a vector space at some point  $x$  on a manifold  $\mathcal{M}$ . (Right) The curved manifold  $\mathcal{M}$  with geodesic  $\gamma(t)$  from the point  $x$  to some other point  $y$  on the manifold.

We can map vectors in the tangent space to the manifold using geodesics. The vector  $v \in T_x\mathcal{M}$  can be mapped to the point of the manifold that is reached after a unit time  $t = 1$  by the geodesic  $\gamma(t)$  starting at  $x$  with initial velocity  $\gamma'(0) = v$ . This mapping  $\text{Exp}_x: T_x\mathcal{M} \rightarrow \mathcal{M}$  is called the exponential map where  $\text{Exp}_x(v) = \gamma(1)$ . The inverse of the exponential map is the logarithmic map  $\text{Log}_x(y)$  and is the smallest vector  $v$  as measured by the Riemannian metric so that  $\text{Exp}_x(v) = y$  [SFP20].

In general, finding geodesics involves solving second-order ordinary differential equations (ODEs) [SFP20]. However, SPD matrices have additional Lie group structure that can be used to simplify algorithms and speed up computations of Riemannian metrics and geodesics [Pen20]. A group is a set  $G$  with an associative binary operation satisfying  $(xy)z = x(yz)$  for all  $x, y, z \in G$ , contains an identity element  $e \in G$  so that  $ex = xe = x$  for all  $x \in G$ , and contains an inverse element  $x^{-1} \in G$  for each  $x \in G$  where  $x^{-1}x = xx^{-1} = e$ . A Lie group is a smooth manifold and also a group [SFP20].

For each  $x, y$  in a Lie group  $G$ , left-translations by  $y$  are denoted  $L_y: x \mapsto yx$ . The differential  $(L_y)_*$  of left-translation maps the tangent space  $T_xG$  to the tangent space  $T_{yx}G$ . In particular,  $(L_y)_*$  maps any vector  $u \in T_xG$  to the vector  $(L_y)_*u \in T_{yx}G$ . The vector field  $\tilde{u}(y) = (L_y)_*u$  is called left-invariant since it is invariant under left-translations  $\tilde{u} \circ L_y = (L_y)_*\tilde{u}$  for all  $y \in G$ .

$G$  [SFP20]. In other words, pullback to the identity and pushforward to another object are commutative under left-translation.

$$\begin{aligned}
(L_y)_* \tilde{u}(x) &= (L_y)_*(L_x)_* u \\
&= (L_{yx})_* u \\
&= \tilde{u}(yx) \\
&= \tilde{u}(L_y x) \\
&= (\tilde{u} \circ L_y)x.
\end{aligned}$$

$$\begin{array}{ccc}
u & \xrightarrow{(L_y)_*} & (L_y)_* u \\
\tilde{u} \uparrow & & \uparrow \tilde{u} \\
e & \xrightarrow{L_y} & y
\end{array}$$

The left-translation maps give a useful way of defining Riemannian metrics on Lie groups. By definition the tangent space  $T_e G$  of a Lie group  $G$  at the identity element, typically denoted  $\mathfrak{g}$ , is called a Lie algebra. Given an inner product  $\langle \cdot, \cdot \rangle_{\mathfrak{g}}$  on the Lie algebra, we can extend it to an inner product on tangent spaces at all elements of the group by setting  $\langle u, v \rangle_g = \langle (L_{g^{-1}})_* u, (L_{g^{-1}})_* v \rangle_{\mathfrak{g}}$ . This defines a left-invariant Riemannian metric on  $G$  since  $\langle (L_h)_* u, (L_h)_* v \rangle_{hg} = \langle u, v \rangle_g$  for any  $u, v \in T_g G$  [SFP20].

$$\begin{aligned}
\langle (L_h)_* u, (L_h)_* v \rangle_{hg} &= \langle (L_{(hg)^{-1}})_* (L_h)_* u, (L_{(hg)^{-1}})_* (L_h)_* v \rangle_{\mathfrak{g}} \\
&= \langle (L_{(hg)^{-1}h})_* u, (L_{(hg)^{-1}h})_* v \rangle_{\mathfrak{g}} \\
&= \langle (L_{g^{-1}h^{-1}h})_* u, (L_{g^{-1}h^{-1}h})_* v \rangle_{\mathfrak{g}} \\
&= \langle (L_{g^{-1}})_* u, (L_{g^{-1}})_* v \rangle_{\mathfrak{g}} \\
&= \langle u, v \rangle_g.
\end{aligned}$$

$$\begin{array}{ccc}
\langle \cdot, \cdot \rangle_g & & \\
(L_g)_* \uparrow & \searrow (L_h)_* & \\
\langle \cdot, \cdot \rangle_{\mathfrak{g}} & \xrightarrow{(L_{hg})_*} & \langle \cdot, \cdot \rangle_{hg}
\end{array}$$

For a matrix  $P \in \mathbb{R}^{n \times n}$  and any non-zero vector  $x \in \mathbb{R}^n$ , if  $P = P^\top$  and  $x^\top P x > 0$ , then  $P$  is an SPD matrix [Mur22]. The space of SPD matrices is a smooth manifold, and not a vector space since it lacks an additive identity, additive inverses, and zero and negative real scalar multiplication [Pen20]. The SPD manifold  $\text{Sym}_n^+$  of all SPD matrices is a Lie group with matrix multiplication as its group operation [SFP20]. Therefore, we can derive invariant metrics and geodesics on the SPD manifold.

Let  $A$  be a matrix from the general linear group  $\text{GL}(n)$  of non-singular matrices. Then  $APA^\top \in \text{Sym}_n^+$  since  $(APA^\top)^\top = (A^\top)^\top P A^\top = APA^\top$  and for any non-zero vector  $x \in \mathbb{R}^n$  then  $x^\top (APA^\top) x = (x^\top A) P (A^\top x) = (A^\top x)^\top P (A^\top x) > 0$ . Given two matrices  $P, Q \in \text{Sym}_n^+$  we can derive the distance between them as a norm from the identity by choosing  $A = P^{-1/2}$  [Pen20]. That is, with left-invariance we can pullback to the identity before pushforward.

$$\text{dist}(P, Q) = \text{dist}(\text{Id}, P^{-1/2} Q P^{-1/2}) = N(P^{-1/2} Q P^{-1/2}). \quad (4)$$

By left-invariance at the identity of a Lie group we have the vector field  $\tilde{u}(x) = (L_x)_* u$ . The geodesic starting at the identity  $e$  with initial velocity  $u$  satisfies  $x(0) = e$  and  $x'(t) = \tilde{u}(x(t))$ . For matrix groups, this ODE becomes  $x'(t) = x(t)u$  and is uniquely solved by the matrix exponential  $x(t) = \exp(tu)$ . In other words, the exponential map is the matrix exponential and its inverse the logarithmic map is the matrix logarithm [SFP20].

From Equation 4 take  $R = P^{-1/2} Q P^{-1/2}$ . The matrix logarithm  $\log(R)$  produces the vector for the norm  $N(R)$ . This norm is given by the Frobenius norm  $\|\cdot\|_F$ . Furthermore, since  $R$  is SPD, it has eigendecomposition  $R = V \Lambda V^\top$  where  $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$  and  $\lambda_i > 0$ . By the matrix logarithm  $\log(R) = V \text{diag}(\log \lambda_1, \dots, \log \lambda_n) V^\top$ . Therefore, by the orthogonal invariance of the Frobenius norm, the norm  $N(R)$  can be written as the square root of the sum of squared eigenvalue logarithms [Pen20].

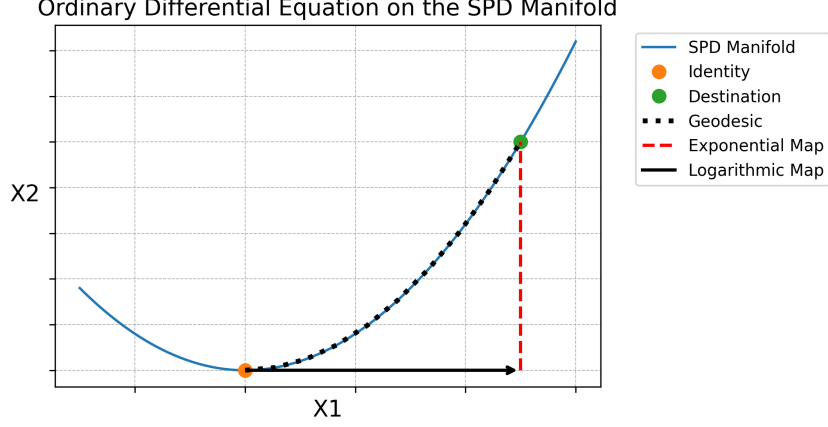


Figure 5: The exponential map  $\text{Exp}_x(v) = \gamma(1) = y$  produces the point on the manifold  $\mathcal{M}$  reached after a unit time  $t = 1$  along the geodesic  $\gamma(t)$  starting at point  $x$  with initial velocity  $\gamma'(0) = v$ . The logarithmic map  $\text{Log}_x(y) = v$  is its inverse and produces the initial velocity needed to reach  $y$  from  $x$  after the unit time  $t = 1$ . On  $\text{Sym}_n^+$  the exponential map is the matrix exponential and the logarithmic map is the matrix logarithm.

$$\text{dist}(P, Q) = N(P^{-1/2}QP^{-1/2}) = \|\log(R)\|_F = \left( \sum_{i=1}^n \log^2 \lambda_i \right)^{1/2}. \quad (5)$$

Equation 5 gives a closed form solution for the distance between two SPD matrices on  $\text{Sym}_n^+$ . Not only is it exact and so does not require optimization, but it can be computed in the vector space of the tangent space and so does not require integration on the curve. Furthermore, by the nature of logarithms we see that matrices with zero or negative eigenvalue are in fact infinite distance from SPD matrices on  $\text{Sym}_n^+$ , contrary to Euclidean space and metrics [Pen20].

We can also find closed form solutions for the logarithmic and exponential maps on  $\text{Sym}_n^+$ . Rather than solve for the norm of the logarithmic map, we can solve for the vector itself. This is done by pushing forward to  $P$  after we pulled back to the identity from  $P$ . Furthermore, as the inverse of the logarithmic map, the closed form solution for the exponential map takes a vector as input and outputs the point on  $\text{Sym}_n^+$  reached after unit time elapsed along the geodesic.

$$\text{Log}_P(Q) = P^{1/2} \log(P^{-1/2}QP^{-1/2})P^{1/2}. \quad (6)$$

$$\text{Exp}_P(W) = P^{1/2} \exp(P^{-1/2}WP^{-1/2})P^{1/2}. \quad (7)$$

At the foundation of statistics is the notion of a distance. The closed form solution of the mean in Euclidean space  $\frac{1}{N} \sum_{i=1}^N X_i$  relies on the additive structure of vector space and does not generalize to Riemannian space. However, there are defining properties of the mean that do generalize: The geometric mean is a least-squares centroid that minimizes the sum-of-squared distances. A natural strategy for computing the geometric mean is gradient descent optimization [Fle20].

$$\bar{y} = \arg \min_{y \in \mathcal{M}} \sum_{i=1}^N \text{dist}(y, y_i)^2. \quad (8)$$

The exponential and logarithmic map, distance, and mean on  $\text{Sym}_n^+$  are common Riemannian geometric statistics used by SOTA ML for EEG that takes across channel sample covariance matrices for classification [Bar+12; Che+18]. The minimum distance to the mean (MDM) classifier is trained by memorizing the mean of each class and is tested by predicting classes based on the minimum distance to the memorized means [Bar+12]. All operations occur on the SPD manifold. In the next section we describe in more detail an alternative algorithm that performs classification in the tangent space of  $\text{Sym}_n^+$ .

## 2.2 Example

Consider two three-dimensional tensors of EEG recordings  $X_1, X_2 \in \mathbb{R}^{p \times q \times r}$  from separate classes where the dimensions  $p, q, r$  are the trials, channels, and time, respectively. For each trial, let sample covariance matrices (SCMs) be computed between channels across time so that  $\Sigma_1^{(k)}, \Sigma_2^{(k)} \in \mathbb{R}^{q \times q}$  for  $k = 1, \dots, p$ . Assume each SCM is conditioned so that it is SPD. Furthermore, let us define  $\bar{\Sigma}_1, \bar{\Sigma}_2 \in \mathbb{R}^{q \times q}$  as geometric mean SCMs averaged across all trials  $k$  of each class.

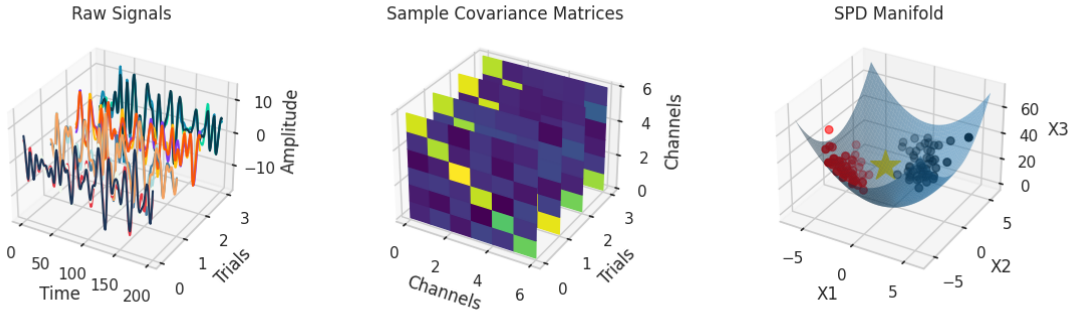


Figure 6: Example of our data preprocessing pipeline. (Left) We receive a three-dimensional tensor (trials  $\times$  channels  $\times$  time) of raw EEG signals recorded from a brain-computer interface session. (Center) We transform each trial of signals into between channel sample covariance matrices (SCMs). (Right) We represent SCMs on the SPD manifold where they are naturally clustered by their label and the golden star is the geometric mean whose tangent space we map all SCMs to for ML in vector space.

With the geometric mean of each class  $\bar{\Sigma}_1, \bar{\Sigma}_2 \in \text{Sym}_n^+$  we define  $\bar{\Sigma} \in \text{Sym}_n^+$  as the geometric mean of these means. Then, using the logarithmic map we find the vectors between all points in all clusters to  $\bar{\Sigma}$ . We represent these vectors in the tangent space  $T_{\bar{\Sigma}}\text{Sym}_n^+$  of the mean point  $\bar{\Sigma}$ . This linearization of SPD matrices is done in a way that respects the metric space of  $\text{Sym}_n^+$ . Once in the tangent space  $T_{\bar{\Sigma}}\text{Sym}_n^+$  we can train standard ML algorithms that are optimized for vector space.

Once in the tangent space  $T_{\bar{\Sigma}}\text{Sym}_n^+$  we can train standard ML algorithms that are optimized for vector space. This method of linearization from the space of SPD matrices to vector space is the standard used by software libraries like PyRiemann [Bar+25]. As of now, SOTA EEG classifiers are those that learn and predict in the tangent space of the SPD manifold rather than on the manifold itself [Bar+12; Che+18]. An explanation for this is that machine learning is traditionally optimized for vector space.

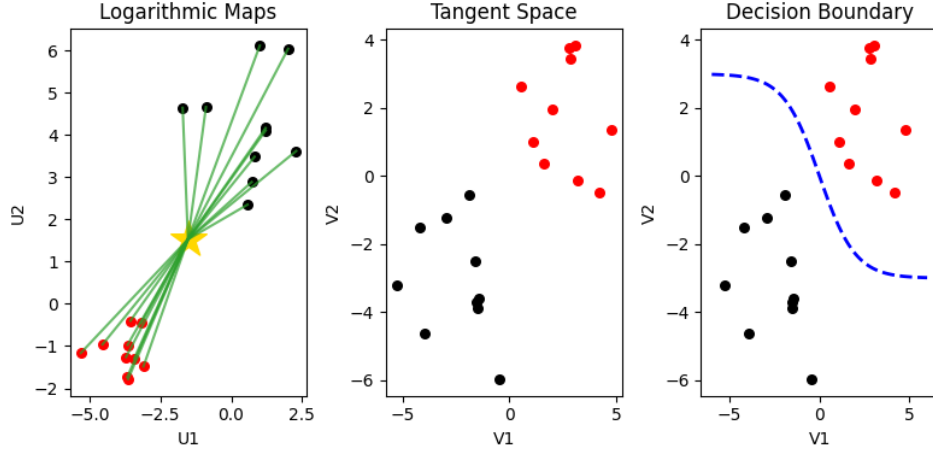


Figure 7: Example of our data preprocessing and classification pipeline. (Left) All SPD matrices are vectorized by the logarithmic map. (Center) The vectorized SPD matrices in tangent space. (Right) A learned decision boundary from a standard ML algorithm.

## References

- [Bar+12] Alexandre Barachant et al. “Multiclass Brain–Computer Interface Classification by Riemannian Geometry”. In: *IEEE Transactions on Biomedical Engineering* 59.4 (2012), pp. 920–928. DOI: 10.1109/TBME.2011.2172210.
- [Bar+25] Alexandre Barachant et al. *pyRiemann*. Version v0.9. July 2025. DOI: 10.5281/zenodo.593816. URL: <https://doi.org/10.5281/zenodo.593816>.
- [Bla18] Benjamin Blankertz. “Gentle Introduction to Signal Processing and Classification for Single-Trial EEG Analysis”. In: *Brain–Computer Interfaces Handbook: Technological and Theoretical Advances*. Ed. by Chang S. Nam, Anton Nijholt, and Fabien Lotte. CRC Press, 2018. Chap. 18, pp. 343–370.
- [Che+18] Sylvain Chevallier et al. “Riemannian Classification for SSVEP-Based BCI: Offline versus Online Implementations”. In: *Brain–Computer Interfaces Handbook: Technological and Theoretical Advances*. Ed. by Chang S. Nam, Anton Nijholt, and Fabien Lotte. CRC Press, 2018. Chap. 19, pp. 371–396.
- [Fle20] Tom Fletcher. “Statistics on manifolds”. In: *Riemannian Geometric Statistics in Medical Image Analysis*. Ed. by Xavier Pennec, Stefan Sommer, and Tom Fletcher. Academic Press, 2020. Chap. 2, pp. 39–74. ISBN: 978-0-12-814725-2. DOI: <https://doi.org/10.1016/B978-0-12-814725-2.00009-1>. URL: <https://www.sciencedirect.com/science/article/pii/B9780128147252000091>.
- [Mur22] Kevin P. Murphy. *Probabilistic Machine Learning: An Introduction*. MIT Press, 2022. URL: <http://probml.github.io/book1>.
- [Pen20] Xavier Pennec. “Manifold-valued image processing with SPD matrices”. In: *Riemannian Geometric Statistics in Medical Image Analysis*. Ed. by Xavier Pennec, Stefan Sommer, and Tom Fletcher. Academic Press, 2020. Chap. 3, pp. 75–134. ISBN: 978-0-12-814725-2. DOI: <https://doi.org/10.1016/B978-0-12-814725-2.00010-8>. URL: <https://www.sciencedirect.com/science/article/pii/B9780128147252000108>.

- [SFP20] Stefan Sommer, Tom Fletcher, and Xavier Pennec. “Introduction to differential and Riemannian geometry”. In: *Riemannian Geometric Statistics in Medical Image Analysis*. Ed. by Xavier Pennec, Stefan Sommer, and Tom Fletcher. Academic Press, 2020. Chap. 1, pp. 3–37. ISBN: 978-0-12-814725-2. DOI: <https://doi.org/10.1016/B978-0-12-814725-2.00008-X>. URL: <https://www.sciencedirect.com/science/article/pii/B978012814725200008X>.