Unraveling Temporal Dynamics of SPSD matrices via Riemmanian Multi-Scale Decomposition

Yonatan Kleerekoper, Mohammad Kurtam, Yitzhak Schiller, Hadas Benisty

Abstract

Understanding the temporal dynamics of high-dimensional, time-varying systems remains a fundamental challenge across scientific disciplines. In many real-world systems, the interactions between sensors are also time-varying, which presents an additional level of complexity in extracting a meaningful representation and subsequent modeling. In this work, we introduce a novel multi-resolution framework for analyzing the temporal dynamics of Symmetric Positive Semi-Definite (SPSD) matrices, such as correlation matrices. A key innovation of our framework is a difference operator that respects the symmetry of the Riemannian manifold. We apply this operator recursively and extract a hierarchical representation of the temporal dynamics of SPSD matrices with a fine frequency resolution. We employ spectral analysis to select informative frequency components and extract key structural patterns. Specifically, our approach reveals which sensors of the original signal drive the correlational dynamics, thus enhancing both denoising and the interoperability of SPSD matrix evolution over time. We validate our method using synthetic data and real neural recordings from mice trained to perform a motor task. Our results demonstrate its ability to extract meaningful structures from complex temporal datasets and provide deeper insights into evolving network connectivity.

1 Introduction

Understanding the temporal dynamics of second-order statistics is often used to uncover meaningful patterns in complex data. In neuroscience, for example, measuring the synaptic strength between cells in behaving animals is impossible. Instead, researchers use the correlation between temporal readouts as a proxy for functional connectivity Benisty et al. [2024], Ghanayim et al. [2025]. This, in combination with advances in understanding the manifold on which these matrices lie Bonnabel and Sepulchre [2010], has allowed the emergence of algorithms such as Riemannian multi-resolution analysis,Shnitzer et al. [2024], and feature selection, Cohen et al. [2023], Barachant et al. [2012] that take into account the geometric structure of these matrices. This, in combination with implementations in the field of deep learning Saxe et al. [2013], Yair et al. [2019], Huang and Van Gool [2017], Wang et al. [2022], and covariance estimation Kalaganis et al. [2020]. Brain-computer interface (BCI) applications Kalaganis et al. [2019], Barachant et al. [2012] have brought to the forefront the importance of utilizing the Symmetric Positive Semi-Definite (SPSD) manifold when observing these dynamics.

In this paper, we expand the operations available on this manifold by introducing a novel time-frequency analysis framework for studying the temporal dynamics of second-order statistics. Our approach constructs a recursive decomposition tailored to investigate the temporal dynamics of SPSD matrices. Due to their inherent symmetry, these matrices reside on a Riemannian manifold with a conical structure. Our method is designed to fully respect this geometric constraint, providing a principled multi-resolution framework for analyzing the dynamics of symmetric operators, such as correlation matrices. The main contributions of this paper are:

Preprint. Under review.

1. We present a new operator for detecting differences between SPSD matrices, which yields an SPSD matrix as its output.
2. We formulate a Wavelet-Packet-like decomposition for a temporal sequence of SPSD matrices.
3. We propose an overall framework for detecting network components driving the temporal dynamics of SPSD matrices.

2 Related Works

The study of structured data on Riemannian manifolds, particularly the manifold of Symmetric Positive-Definite matrices, has gained significant attention in signal processing, machine learning, and neuroscience. Our work relies on advances in Riemannian geometry and wavelet-based analysis while addressing key challenges that have hindered the development of a multi-resolution analysis framework thus far.

2.1 Multi-Resolution Analysis in Riemannian Geometry

Wavelet transforms have long been a fundamental tool for multiscale analysis Daubechies [1992], yet their extension to Riemannian manifolds remains a significant challenge. A notable advancement in this direction is the work of Shnitzer et al. [2024], who introduced a Riemannian composition of diffusion operators tailored for multi-resolution analysis on the Symmetric Positive Definite (SPD) manifold, as well as providing an extension for Symmetric Positive Semi Definite (SPSD) matrices based on the work of Bonnabel and Sepulchre [2010]. Their approach models temporal dynamics using diffusion-based methods, aligning with our goal of decomposing SPD sequences while preserving geometric consistency. Specifically, Shnitzer et al. define two key operators between SPD matrices: a similarity operator that preserves shared structural components and a difference operator that captures differences between them. However, their formulation of the difference operator maps the matrices onto the tangent space, leading to an inherent limitation. Once the transformed matrix leaves the SPD manifold, further decomposition while maintaining geometric fidelity becomes infeasible. Therefore, their algorithm provides a Wavelet-like analysis by recursively applying the similarity operator only. Addressing this limitation is central to our approach, which aims to recursively apply both operators entirely within the Riemannian framework, thus providing a Wavelet-Packet-like decomposition that respects the Riemannian manifold of SPSD matrices.

Beyond the scope of wavelet-like analysis, these operators have found applications in feature selection tasks. Cohen et al. [2023] leverage the same geometric operators in few-sample settings, demonstrating their efficacy in selecting meaningful features while preserving the underlying manifold structure. Their findings underscore the importance of respecting the geometric nature of SPD matrices, a principle our work builds upon to develop a more robust decomposition technique.

2.2 Riemannian geometry for neural data

Understanding the dynamics of functional connectivity in the brain, particularly in the context of learning, has been a growing focus in neuroscience. Recent studies have explored the temporal evolution of functional connectivity and its relationship to behavioral and cognitive states. For example, Benisty et al. [2024] shows that spontaneous behavior can be meaningfully represented through the dynamics of functional connectivity, while Ghanayim et al. [2025] investigates how proficiency in motor learning is reflected in the evolution of neural correlation structures. These studies typically model functional connectivity using correlation matrices, where high correlation scores indicate strong neural interactions. Since such matrices lie on the symmetric positive semi-definite (SPSD) manifold, tracking their evolution within a Riemannian framework has provided valuable insights into brain dynamics. However, existing approaches primarily focus on analyzing the overall dynamics of these correlation matrices without decomposing them into meaningful sub-components. A key question remains: which structural components drive neural adaptation and learning? Our work addresses this gap by introducing a structured decomposition of SPD sequences that respects the manifold geometry. By doing so, we provide a novel perspective on how specific components contribute to neural learning dynamics, offering a complementary approach to existing studies in Riemannian neural data analysis.

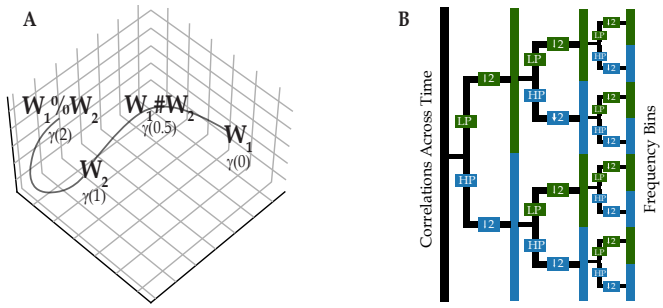

Figure 1: A Geometric Representation of Manifold Movement and Wavelet Packet Decomposition. (A) A schematic illustration of the geodesic connecting two matrices where the similarity operator is positioned at the midpoint, while the difference operator extends the geodesic such that W_2 becomes the midpoint between $W_1 \% W_2$ and W_1. (B) An illustration of the proposed analysis consisting of recursive application of low-pass and high-pass filters, followed by down-sampling by a factor of two.

3 Methods

3.1 Problem Formulation

We consider a time-varying system observed through d sensors at two different time scales. Let $\{\mathbf{x}_n[t]\}_{t=1}^{T}$ be a sequence of observations where t denotes the faster temporal resolution and $n = 1, \ldots, N$ denotes the index of the slower one. We assume that variations of the intrinsic properties of the system are relatively slow. Consequently, the second-order statistics of the system remain constant for a given n. Let $\mathbf{C}_n$ be Pearson's correlation evaluated using$\{\mathbf{x}_n[t]\}_{t=1}^{T}$, for a given n. The set $\{\mathbf{C}_n\}_{n=1}^{N}$ defines a $d(d-1)/2$ time traces expressing the temporal dynamics of the second-order statistics of the observed signals. This setup is common in the study of learning in neuroscience, where instantaneous neuronal activity is recorded as animals are trained to perform a task. In this case, the slower temporal resolution could be trials during a training session or even days between training sessions. Analysis of this high-dimensional temporal sequence raises two main questions: 1) Can we identify a low-dimensional representation to infer specific dynamics? 2) Which components in the original, high-dimensional space drive these dynamics?

Answering these questions is not straightforward in the case of SPSD matrices since they do not obey Euclidean geometry. Consequently, standard methods for modeling or dimensionality reduction that rely on mean squared distances will not be accurate and may introduce unexpected artifacts. To answer the first question, kernel-based dimensionality reduction techniques and Riemannian geometry are typically used to extract a low-dimensional representation for the temporal dynamics Benisty et al. [2024], Ghanayim et al. [2025]. However, these methods are not interpretable: they do not provide a way to identify which sensors, i.e., elements of $\mathbf{x}_n[t]$, drive the temporal dynamics of the correlations, so they cannot be used to answer the second question. A simplified approach is applying a univariate analysis and exploring each interaction term's temporal evolution, $C_n(l,k)$, separately. However, this approach does not account for the overall correlational structure of the network, and it is unclear how to aggregate the outcomes across all interactions.

Here, we present a novel approach inspired by Wavelet Packet Decomposition for analyzing temporal sequences of SPSD matrices. Our proposed analysis respects the geometric structure of the Riemannian manifold and identifies the elements driving these dynamics. First, we define low-pass and high-pass filters respecting the structure of the Riemannian manifold. Second, we present our construction for a recursive decomposition of this sequence. Third, we propose an algorithm for detecting the elements driving the correlational dynamics.

3.2 Low-pass and High-pass filters for SPSD matrices

A key aspect of the wavelet packet decomposition is the choice of low-pass and high-pass filters. Discrete Haar functions are a common choice for low-pass and high-pass filters Daubechies [1992]:

$$\begin{aligned} h_{LP}[n] &= \frac{\sqrt{2}}{2}[1,1] \\ h_{HP}[n] &= \frac{\sqrt{2}}{2}[-1,1] \end{aligned} \tag{1}$$

For Haar functions, the outputs of the low-pass and high-pass filters are the mean and difference of two consecutive samples, respectively. Inspired by Haar functions, we follow Shnitzer et al. [2024] and use the similarity operator, defined by the midpoint on the geodesic between the two kernels, W_1 and W_2.

$$W_1 \# W_2 \triangleq \gamma_{W_1 \to W_2}(0.5) = W_1^{0.5}(W_1^{-0.5} W_2 W_1^{-0.5})^{0.5} W_1^{0.5} \tag{2}$$

where $\gamma_{W_1 \to W_2}(t)$, $0 \leq t \leq 1$ is the geodesic connecting W_1 and W_2 on the Riemannian manifold. Applying this operator to a sequence of SPSD matrices would result in the Riemannian mean of every consecutive pair:

$$\begin{aligned} \mathbf{C}_n^{LP} &= \mathbf{C}_n \# \mathbf{C}_{n+1} \\ &\forall \quad n = 1, ..., N-1 \end{aligned} \tag{3}$$

We note that the filtered sequence, $\mathbf{C}_n^{LP}$, is also SPSD, since every matrix is the mid-point on the geodesic between two other SPSD matrices. This property is fundamental, since Wavelet Packet decomposition is based on the recursive application of these filters. The difference operator defined by Shnizter et al. is based on the tangent and, therefore, does not conform to this requirement. Instead, we propose the following operator:

$$W_1 \% W_2 \triangleq \gamma_{W_1 \to W_2}(2) = W_1^{0.5}(W_1^{-0.5} W_2 W_1^{-0.5})^{2} W_1^{0.5} \tag{4}$$

Proposition 1. For any pair of SPSD matrices W_1 and W_2, the geodesic can be extended from $\gamma(-\infty)$ to $\gamma(\infty)$. Moreover, the SPSD manifold is geodesically complete.

Proof. Proof given in Bonnabel and Sepulchre [2010] Proposition 3. □

Proposition 1 ensures that the resulting matrix is indeed SPSD, allowing us to recursively apply it. Figure 1A illustrates the geodesic between two matrices, the similarity operator as the midpoint on the geodesic, and the difference operator as an extension of the geodesic such that W_2 is the midpoint - thus making the similarity and difference operators mutually inverse. (See Proposition 2 in appendix)

Examining the spectrum of the difference operator (see sec. 5, Theorems 1-4), we observe that it emphasizes differences. We, therefore, use it to define the following high-pass operation:

$$\begin{aligned} \mathbf{C}_n^{HP} &= \mathbf{C}_n \% \mathbf{C}_{n+1} \\ &\forall \quad n = 1, ..., N-1 \end{aligned} \tag{5}$$

3.3 Wavelet Packet Decomposition on the Riemannian Manifold

Standard Wavelet Packet Decomposition consists of a recursive application of low-pass and high-pass filters followed by down-sampling by a factor of two, as illustrated in Fig. 1B. Similarly, we construct a recursive multi-resolution decomposition for a sequence of SPSD matrices based on the low-pass and high-pass operations defined above (eqns. (3) and (5)).

We calculate the low-pass and high-pass operations and down-sample the resulting signals by a factor of 2, producing two frequency bins of length $N/2$: one corresponding to high-frequency components and the other to low-frequency components. This process is repeated recursively for $\log_2(N)$ steps, ultimately generating a hierarchical decomposition where the last level consists of N SPSD matrices corresponding to frequency components denoted by $\left\{\hat{\mathbf{C}}_f\right\}_{f=1}^{N}$. In the next section, we use this decomposition and propose an algorithm for detecting elements driving the dynamics.

3.4 Detection of Dynamics Drivers

Standard Wavelet analysis is typically followed by a thresholding operation to extract frequency bins containing meaningful information. The proposed above analysis does not yield a scalar per frequency bin, but a matrix, so detecting meaningful frequency bins is not straightforward. In this section, we propose an algorithm for the detection of dynamic drivers - elements of $\mathbf{x}_n[t]$ - which are driving the temporal evolution of $\mathbf{C}_n$. First, we apply eigen-decomposition to each matrix in the sequence $\left\{\hat{\mathbf{C}}_f\right\}_{f=1}^{N}$ and extract the eigenvector related to the highest eigenvalue:

$$\hat{\mathbf{C}}_f \psi_f = \lambda_f \psi_f \tag{6}$$

We quantify how informative each eigenvector is using the following normalized entropy:

$$h_f \triangleq -\frac{1}{\log d} \sum_{i=1}^{d} p_f\left(i\right) \log p_f\left(i\right) \tag{7}$$

where $p_f\left(i\right)$ are the normalized entries of each eigenvector:

$$p_f\left(i\right) = \frac{\psi_f\left(i\right)}{\sum_{i=1}^{d} \psi_f\left(i\right)} \tag{8}$$

We retain the eigenvectors with the lowest entropy scores as they capture more structured information. The proportion of selected eigenvectors depends on the number of frequency bins; empirically, we find that taking the bottom 20% typically yields good results.

We then apply k-means clustering to the selected eigenvectors, partitioning their entries into high-value and low-value components. We further refine this partition by counting the fraction of times each element appears as high-valued across all frequency bins. We identify elements having significantly high scores as dynamic drivers, as they consistently appear in the high-value cluster. Figure 2D presents the voting scores where the red and blue bars correspond to the six elements having sinusoidal interactions while the gray bars relate to the noisy elements.

Next, we concentrate on the $d^* \leq d$ elements identified as dynamic drivers and aim to capture structural relationships by clustering them into sub-groups. We construct a graph of d^* nodes. The edges connecting the nodes are weighted according to the number of times each pair of dynamic elements co-occurs across low entropy frequency bins. We then apply the Louvain algorithm Blondel et al. [2008] to cluster this graph into communities, grouping tightly connected components.

Finally, returning to the sequence of matrices $\mathbf{C}_n$, we construct a mask that preserves intra-community dynamics while all other elements, interactions between different communities, or interactions involving non-dynamic drivers are set to zero. This results in a denoised version of the original sequence of SPSD matrices. Each matrix is now block-diagonal, retaining the essential dynamic structures while filtering out unstructured noise (See Algorithm 1).

4 Results

In this section, we present the results of our proposed algorithm on a toy example and neural data. We note that these results were obtained using a CPU due to the algorithm's short run times. Still, given the design of the proposed low-pass and high-pass operators, the computations could be efficiently performed on a GPU, further improving runtime.

4.1 Toy Example

We consider a sequence of 512 SPSD $20X20$ matrices where several elements follow structured sinusoidal dynamics, and the rest are randomly drawn from a uniform distribution:

$$\begin{aligned} C_n\left(k,l\right) &= 0.5\sin\left(2\pi f_1 n\right), \ \ \forall n < 256, \ \ k,l = 1,2,3 \\ C_n\left(k,l\right) &= 0.5\sin\left(2\pi f_2 n\right), \ \ \forall n \geq 256, \ \ k,l = 7,8,9 \\ C_n\left(k,l\right) &\sim U\left(0.25, 0.25\right), \qquad l,k \neq 1,2,3,7,8,9 \\ k &\neq l \end{aligned} \tag{9}$$

Algorithm 1 Riemannian Multi Scale Decomposition for Dynamics Detection

Require: SPSD matrix sequence C_n of length N

Construct wavelet packet decomposition:
for $k = 1$ to $\log_2(N)$ **do**
  for each node in level $k - 1$ **do**
    Apply low-pass and high-pass filtering using eqns. 3 and 5
    Downsample the resulting signals by 2
  end for
end for
for each frequency bin f **do**
  Extract the eigenvector corresponding to the highest eigenvalue (eqn. 6).
  Compute the normalized entropy using eqns. 7 and 8.
end for
Select eigenvectors with the lowest entropy scores.
Apply k-means clustering to classify components.
Identify dynamic drivers as mostly voted by the vectors in the high-value cluster.
Construct a co-occurrence graph from clustering results.
Apply Louvain community detection to extract significant structures.
Zero out uninformative elements in original matrices.
return Refined matrices containing informative dynamic structures.

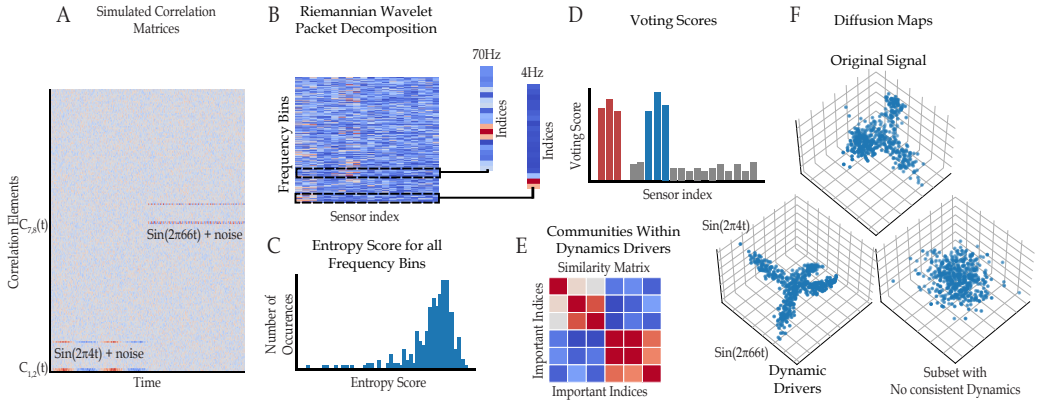

Figure 2: A toy example: a simulated sequence of SPSD matrices. (A) A heatmap presenting the time traces of all individual $d(d-1)$ elements. (B) Left - spectral energy distribution across frequency bins. Right - two selected frequency bins corresponding to the high- and low-frequency sinusoidal components. (C) A histogram of the entropy scores computed for all frequency bands. (D) A similarity matrix related to the graph of the elements identified as dynamic drivers. The block diagonal structure indicates two communities. (E) The importance scores of each matrix component within the network are colored according to clusters identified by the Louvain algorithm. (F) Diffusion embedding of the sequence of matrices considering: top — all elements (original data), bottom left — dynamics drivers, bottom right - non-drivers only.

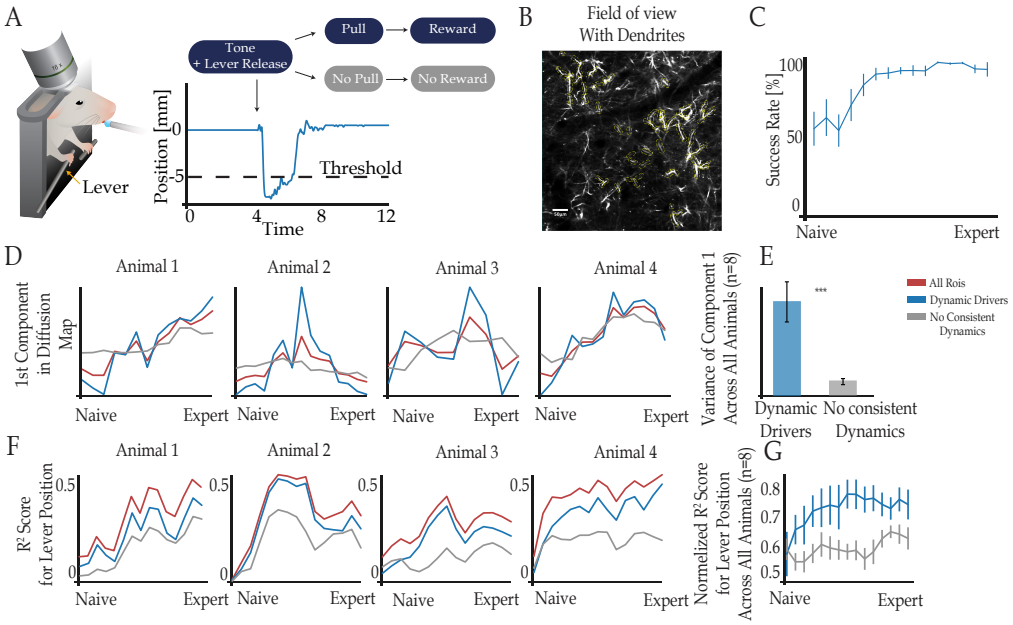

Figure 3: Neural Dynamics in the Lever-Pull Task (A) Schematic of the experimental setup. (B) Example field of view from the calcium imaging data, where white streaks indicate ROI labeling of dendritic activity. (C) Success rate of 8 mice performing the lever-pull task across training days, showing an increase in performance with training. (D) First component of the low-dimensional representation of correlational dynamics using diffusion map applied to correlation matrices for four out of the eight animals, considering ROIs identified as dynamics drivers (blue) or those that were not (gray). The overall network dynamics (red) appear to represent the mean of these two groups. (E) Variance across sessions of the first diffusion map component across all animals, showing a statistically significant difference ($p = 0.002$, Wilcoxon rank-sum test) between the two groups. (F) R^2 scores of a linear regression model predicting lever position from neuronal activity for the same four animals across all training days, considering dynamics driving ROIs (blue), non-driving ROIs (gray), and all ROIs (red). (G) R^2 achieved by dynamic drivers (blue) and non-driving ROIs (gray), normalized by the R^2 score achieved by all ROIs, averaged across eight animals, expressed as a percentage.

where $f_1 = 4Hz$ and $f_2 = 66Hz$ and the main diagonal is all ones: $C_n(k,k) = 1, \forall 0 \leq n < 512,\ k = 1, ..., d$. We applied Algorithm 1 to this data and obtained the first eigenvector of each frequency bin, which reveals a distinct pattern indicating elements 1-3 at 4Hz and elements 7-9 at approximately 70Hz, as illustrated in Figure 2B. Applying the proposed entropy-based analysis to these eigenvectors correctly identifies the elements driving the sinusoidal dynamics and their partition into two communities, one related to 4 Hz and one to approximately 70 Hz, as illustrated in Fig. 2 C-E. To demonstrate the effectiveness of the proposed method, we extracted a low dimensional representation of the original temporal sequence of SPSD matrices using a diffusion map Coifman and Lafon [2006] fed with Riemannian distances between the matrices Benisty et al. [2024], Ghanayim et al. [2025] as illustrated in 2 F (top). Keeping only intra-community interactions and eliminating all the rest (by setting them to zero at all time points) yields a cleaner representation of the manifold, showing two segments, each related to half of the time points and a specific community (left). Alternatively, eliminating all elements identified as dynamic drivers and keeping all the rest results in a noisy point cloud, as expected (right). Overall, this toy example demonstrates the proposed analysis framework's ability to capture the intrinsic structure of the two sub-networks driving these dynamics successfully while effectively removing noise.

4.2 Detection of dynamics in neural data

We applied our proposed algorithm to real neural data recorded from the primary motor cortex (M1) of mice performing a lever-pull task. As motor learning is related to mechanisms of synaptic plasticity, this study aims to investigate temporal changes in functional connectivity between dendritic spines of cells in layer 5 of M1 during motor learning. The dataset consists of Calcium imaging recordings from 8 mice collected over a 12-day training period. Mice were trained to pull a lever in response to an auditory tone, which was rewarded by water (Figure 3A-C). Each training session consisted of 30 trials, each lasting 12 seconds, with the auditory tone presented at the 4th second of each trial. Neuronal activity was recorded from manual labeling of ROIs across the dendritic trees at different locations. Overall, we recorded between 6 and 12 cells per animal. Neuronal activity was sampled at 30Hz, resulting in 360 time samples per trial. As our goal was to infer changes in functional connectivity across trials (and even days in training), we evaluated the sample correlation between ROIs using time samples related to each trial (i.e., 360 time samples). This resulted in a temporal sequence of hundreds of matrices related to all trials and training sessions per animal. We applied our proposed analysis to identify the regions of interest (ROIs) that drove temporal dynamics during motor learning. Figure 3D presents the first component of the low-dimensional representation (using diffusion maps) of the correlational dynamics for four example animals. The gray trace, indicating the temporal variation of the dynamic drivers (blue), is higher compared to the gray trace achieved using elements not detected as dynamic drivers (gray traces). This phenomenon is further verified across all eight animals, as the temporal variance of the dynamic drivers is significantly higher than that of the other elements, as shown in Figure 3E.

To further assess the functional relevance of the identified components, we trained a linear regression model to predict lever position based on neuronal activity. We found that the neurons identified by our method achieved a higher accuracy score compared to those not classified as dynamic, demonstrating that our approach effectively isolates the key contributors to functional network activity, Figure 3F, G. These results highlight the ability of our framework to extract meaningful dynamic components from high-dimensional neural data, offering a powerful tool for studying the evolution of functional connectivity in learning tasks.

5 Spectral analysis

In this section, we conduct a spectral analysis of our proposed operators, examining their effects on both similar and different components. Additionally, we assess the stability of our method by analyzing how small perturbations in a common eigenvector influence the results.

5.1 Strictly common components

Theorem 1. Consider a vector ψ, which is an eigenvector of both W_1 and W_2 with possibly different eigenvalues:

$$W_1\psi = \lambda_1\psi, \quad W_2\psi = \lambda_2\psi.$$

Shnitzer et al. [2024] show that ψ is also an eigenvector of $S = W_1 \# W_2$ with the corresponding eigenvalue:

$$\lambda_S = \sqrt{\lambda_1\lambda_2}.$$

This result implies that strictly common components that are dominant and similarly expressed in both matrices (with similar large eigenvalues) are dominant in the result of the $\#$ operator.

Theorem 2. Consider a vector ψ, which is an eigenvector of both W_1 and W_2 with possibly different eigenvalues:

$$W_1\psi = \lambda_1\psi, \quad W_2\psi = \lambda_2\psi.$$

Then, ψ is also an eigenvector of the difference operator $Q = W_1 \% W_2$ with the corresponding eigenvalue:

$$\lambda_Q = \lambda_1^{-1}\lambda_2^2.$$

Proof.

$$\begin{aligned} Q\psi &= W_1 \% W_2 \psi \\ &= W_1^{0.5} \left(W_1^{-0.5} W_2 W_1^{-0.5}\right)^2 W_1^{0.5} \psi \\ &= W_1^{0.5} \left(W_1^{-0.5} W_2 W_1^{-0.5}\right)^2 \sqrt{\lambda_1} \psi \\ &= W_1^{0.5} \left(\lambda_1^{-1} \lambda_2\right)^2 \sqrt{\lambda_1} \psi \\ &= \lambda_1^{-1} \lambda_2^2 \psi. \end{aligned}$$

□

This result implies that components that are different between the two matrices (meaning the same eigenvector with different eigenvalues) will be dominant in Q, whereas similarly expressed values will be similarly expressed.

5.2 Weakly common components

In order to provide the stability analysis, we consider the case of weakly common components. We denote $\Lambda_\epsilon(W)$ as the ϵ pseudo spectrum of the matrix W as defined by Trefethen and Embree [2020].

Definition 1. Let $W \in \mathbb{R}^{N \times N}$, the following definitions of the ϵ-psuedo-spectrum are equivalent for a small $\epsilon > 0$:

1. The set of $\lambda \in \mathbb{R}$ such that $\|(\lambda I - W)^{-1}\| \geq \epsilon^{-1}$
2. The set of $\lambda \in \mathbb{R}$ such that $\lambda \in \Lambda(W + E)$ for some $E \in \mathbb{R}^{N \times N}$ with $\|E\| \leq \epsilon$
3. The set of $\lambda \in \mathbb{R}$ such that $\|(\lambda - W)v\| \leq \epsilon$ for some $v \in \mathbb{R}^N$ with $\|v\| = 1$

Theorem 3. Suppose there exists an eigen-pair (λ_k, ψ_k) of W_k for $k = 1, 2$ so that $\psi_1 = \psi_2 + \psi_\epsilon$, where $\|\psi_\epsilon\|_2 \leq \frac{\sqrt{\lambda_2}}{\tilde{\lambda}_{2\max}\sqrt{\lambda_1}}\epsilon$ for a small $\epsilon > 0$, where $\tilde{\lambda}_{2\max} = \|W_2 - \lambda_2 I\|$. Then we have: $\sqrt{\lambda_1 \lambda_2} \in \Lambda_\epsilon(S)$. Specifically, we have: $\|(S - \sqrt{\lambda_1 \lambda_2} I)\psi_1\|_2 \leq \epsilon$, and ψ_1 is a corresponding ϵ–pseudo-eigenvector of S.

Proof. See Theorem 3 in Shnitzer et al. [2024]. □

Theorem 4. Suppose there exists an eigenpair λ_k and ψ_k of W_k for $k = 1, 2$ so that $\psi_1 = \psi_2 + \psi_\varepsilon$, where $\|\psi_\varepsilon\|_2 \leq \frac{1}{(\sigma_{\max}(W_2 {W_1}^{-1}) + \lambda_2 \lambda_1^{-1})\tilde{\lambda}_2^{\max}}\varepsilon$, where $\sigma_{\max}$ denotes the maximum eigenvalue, for a small $\varepsilon > 0$, and where $\tilde{\lambda}_2^{\max} = \|W_2 - \lambda_2 I\|$. Then we have $\lambda_2^2 \lambda_1^{-1} \in \Lambda_\varepsilon(Q)$. Specifically, we have $\|(Q - \lambda_2^2 \lambda_1^{-1} I)\psi_1\|_2 \leq \varepsilon$, and ψ_1 is a corresponding ε-pseudo-eigenvector of Q.

Proof. By using (11) (see appendix), we have:

$$Q = {W_1}^{0.5} \left({W_1}^{-0.5} W_2 {W_1}^{-0.5}\right)^2 {W_1}^{0.5} = (W_2 {W_1}^{-1})^2 W_1.$$

Since ψ_1 is an eigenvector of W_1 with eigenvalue λ_1, we obtain:

$$Q\psi_1 = (W_2 {W_1}^{-1})^2 W_1 \psi_1 = \lambda_1 (W_2 {W_1}^{-1})^2 \psi_1.$$

Thus, it suffices to show that ψ_1 is an ε-pseudo-eigenvector of $(W_2 {W_1}^{-1})^2$. By direct expansion, we have:

$$\begin{aligned} W_2 {W_1}^{-1} \psi_1 = W_2 {W_1}^{-1} \psi_1 &= \frac{1}{\lambda_1} W_2 \psi_1 = \frac{1}{\lambda_1} W_2 (\psi_2 + \psi_\varepsilon). \\ &= \frac{\lambda_2}{\lambda_1} \psi_1 + \frac{1}{\lambda_1} (W_2 - \lambda_2 I)\psi_\varepsilon. \end{aligned}$$

This implies:

$$\left((W_2 {W_1}^{-1})^2 - \lambda_2^2 \lambda_1^{-2} I\right) \psi_1 = (W_2 {W_1}^{-1} + \lambda_2 \lambda_1^{-1} I)(W_2 {W_1}^{-1} - \lambda_2 \lambda_1^{-1} I)\psi_1.$$

Multiplying both sides by λ_1, we get:

$$(Q - \lambda_2^2 \lambda_1^{-1} I)\psi_1 = (W_2 {W_1}^{-1} + \lambda_2 \lambda_1^{-1} I)(W_2 - \lambda_2 I)\psi_\varepsilon.$$

Taking the ℓ_2 norm, we obtain:

$$\|(Q - \lambda_2^2\lambda_1^{-1}I)\psi_1\|_2 \leq \|W_2{W_1}^{-1} + \lambda_2\lambda_1^{-1}I\|_2\|W_2 - \lambda_2 I\|_2\|\psi_\varepsilon\|_2.$$

Applying the given bound on $\|\psi_\varepsilon\|_2$, we conclude:

$$\|(Q - \lambda_2^2\lambda_1^{-1}I)\psi_1\|_2 \leq \varepsilon.$$

Thus, $\lambda_2^2\lambda_1^{-1}$ is an ε pseudo-eigenvalue of Q, where ψ_1 is a corresponding ε pseudo-eigenvector. □

6 Conclusion

In this paper, we introduced a novel multi-resolution hierarchical framework tailored for the analysis of temporal sequences of SPSD matrices. Our approach respects the geometric structure of SPSD matrices, thus enabling recursive filtering for both low and high-pass filtering operations. Our proposed algorithm, based on spectral analysis of the resulting representation, successfully identifies the key structural components that drive temporal changes. Our results, obtained from both simulated and neural activity, demonstrate the efficacy of our method in extracting meaningful dynamics while rejecting noisy elements. This framework offers a principled approach to studying evolving connectivity patterns, providing new insights into time-varying network structures. Future work may extend this approach to other structured data types, incorporate adaptive filtering strategies, and explore applications in broader domains, such as climate science, finance, and graph-based learning.

References

Alexandre Barachant, Stéphane Bonnet, Marco Congedo, and Christian Jutten. Multiclass brain–computer interface classification by riemannian geometry. *IEEE Transactions on Biomedical Engineering*, 59(4):920–928, 2012. doi: 10.1109/TBME.2011.2172210.

Hadas Benisty, Daniel Barson, Andrew H Moberly, Sweyta Lohani, Lan Tang, Ronald R Coifman, Michael C Crair, Gal Mishne, Jessica A Cardin, and Michael J Higley. Rapid fluctuations in functional connectivity of cortical networks encode spontaneous behavior. *Nature Neuroscience*, 27(1):148–158, 2024.

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008 (10):P10008, 2008.

Silvère Bonnabel and Rodolphe Sepulchre. Riemannian metric and geometric mean for positive semidefinite matrices of fixed rank. *SIAM Journal on Matrix Analysis and Applications*, 31(3): 1055–1070, 2010. doi: 10.1137/080731347.

David Cohen, Tal Shnitzer, Yuval Kluger, and Ronen Talmon. Few-sample feature selection via feature manifold learning. In *International Conference on Machine Learning*, pages 6296–6319. PMLR, 2023.

Ronald R. Coifman and Stéphane Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, 2006. ISSN 1063-5203. doi: https://doi.org/10.1016/j.acha.2006.04.006. Special Issue: Diffusion Maps and Wavelets.

Ingrid Daubechies. Ten lectures on wavelets. *Society for industrial and applied mathematics*, 1992.

Amir Ghanayim, Hadas Benisty, Avigail Cohen Rimon, Sivan Schwartz, Sally Dabdoob, Shira Lifshitz, Ronen Talmon, and Jackie Schiller. Vta projections to m1 are essential for reorganization of layer 2-3 network dynamics underlying motor learning. *Nature Communications*, 16(1):200, 2025.

Zhiwu Huang and Luc Van Gool. A riemannian network for spd matrix learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

Fotis P Kalaganis, Nikos A Laskaris, Elisavet Chatzilari, Spiros Nikolopoulos, and Ioannis Kompatsiaris. A riemannian geometry approach to reduced and discriminative covariance estimation in brain computer interfaces. *IEEE Transactions on Biomedical Engineering*, 67:245–255, 2019.

Fotis P. Kalaganis, Nikos A. Laskaris, Elisavet Chatzilari, Spiros Nikolopoulos, and Ioannis Kompatsiaris. A riemannian geometry approach to reduced and discriminative covariance estimation in brain computer interfaces. *IEEE Transactions on Biomedical Engineering*, 67:245–255, 2020. doi: 10.1109/TBME.2019.2912066.

Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013.

Tal Shnitzer, Hau-Tieng Wu, and Ronen Talmon. Spatiotemporal analysis using riemannian composition of diffusion operators. *Applied and Computational Harmonic Analysis*, 68:101583, 2024.

Lloyd N Trefethen and Mark Embree. Spectra and pseudospectra: the behavior of nonnormal matrices and operators. *Princeton university press*, 2020.

Rui Wang, Xiao-Jun Wu, Ziheng Chen, Tianyang Xu, and Josef Kittler. Dreamnet: A deep riemannian network based on spd manifold learning for visual classification. *arXiv preprint arXiv:2206.07967*, 2022.

Or Yair, Mirela Ben-Chen, and Ronen Talmon. Parallel transport on the cone manifold of spd matrices for domain adaptation. *IEEE Transactions on Signal Processing*, 67(7):1797–1811, 2019.

7 Appendix

7.1 The similarity and difference operators are mutually inverse.

Proposition 2. Let W_1 and W_2 be SPSD matrices, the operators defined in (2)(4) are mutually inverse, meaning:

$$\begin{aligned} W_2\%(W_1\#W_2) &= W_1 \\ W_2\#(W_2\%W_1) &= W_1 \end{aligned} \tag{10}$$

Proof. We will first prove that $W_2\%(W_1\#W_2) = W_1$:

$$W_1\#W_2 = W_1^{0.5}(W_1^{-0.5}W_2W_1^{-0.5})^{0.5}W_1^{0.5} = W_3$$

$$\begin{aligned} W_2\%W_3 &= W_2^{0.5}(W_2^{-0.5}W_3W_2^{-0.5})^2W_2^{0.5} \\ &= W_2^{0.5}(W_2^{-0.5}W_1^{0.5}(W_1^{-0.5}W_2W_1^{-0.5})^{0.5}W_1^{0.5}W_2^{-0.5})^2W_2^{0.5} \\ &= W_1^{0.5}(W_1^{-0.5}W_2W_1^{-0.5})^{0.5}W_1^{0.5}W_2^{-1}W_1^{0.5}(W_1^{-0.5}W_2W_1^{-0.5})^{0.5}W_1^{0.5} \end{aligned}$$

Assume this expression equals W_1:

$$W_1 = W_1^{0.5}(W_1^{-0.5}W_2W_1^{-0.5})^{0.5}W_1^{0.5}W_2^{-1}W_1^{0.5}(W_1^{-0.5}W_2W_1^{-0.5})^{0.5}W_1^{0.5}$$

$$W_1^{-0.5}(W_1^{-0.5}W_2W_1^{-0.5})^{-0.5}W_1^{-0.5}W_1W_1^{-0.5}(W_1^{-0.5}W_2W_1^{-0.5})^{-0.5}W_1^{-0.5} = W_2^{-1}$$

$$W_1^{-0.5}(W_1^{-0.5}W_2W_1^{-0.5})^{-0.5}(W_1^{-0.5}W_2W_1^{-0.5})^{-0.5}W_1^{-0.5} = W_2^{-1}$$

$$W_1^{-0.5}(W_1^{-0.5}W_2W_1^{-0.5})^{-1}W_1^{-0.5} = W_2^{-1}$$

$$W_1^{-0.5}W_1^{0.5}W_2^{-1}W_1^{0.5}W_1^{-0.5} = W_2^{-1}$$

$$W_2^{-1} = W_2^{-1}$$

$$\Rightarrow W_1 = W_2\%(W_1\#W_2)$$

Now, we prove that $W_2\#(W_2\%W_1) = W_1$:

$$W_2\%W_1 = W_2^{0.5}(W_2^{-0.5}W_1W_2^{-0.5})^2W_2^{0.5} = W_4$$

$$\begin{aligned} W_2\#W_4 &= W_2^{0.5}(W_2^{-0.5}W_4W_2^{-0.5})^{0.5}W_2^{0.5} \\ &= W_2^{0.5}(W_2^{-0.5}W_2^{0.5}(W_2^{-0.5}W_1W_2^{-0.5})^2W_2^{0.5}W_2^{-0.5})^{0.5}W_2^{0.5} \\ &= W_2^{0.5}((W_2^{-0.5}W_1W_2^{-0.5})^2)^{0.5}W_2^{0.5} \\ &= W_2^{0.5}W_2^{-0.5}W_1W_2^{-0.5}W_2^{0.5} \\ &= W_1 \end{aligned}$$

□

7.2 Equivalent form

Proposition 3. Equivalent form of the filters. We have

$$\mathbf{W}_1^{0.5}\left(W_1^{-0.5}W_2W_1^{-0.5}\right)^p W_1^{0.5} = \left(W_2W_1^{-1}\right)^p W_1$$

Proof. For some $p \geq 0$, consider the expression:

$$W_1^{0.5}\left(W_1^{-0.5}W_2W_1^{-0.5}\right)^p W_1^{0.5}.$$

Define the matrices:

$$M = W_1^{-0.5}W_2W_1^{-0.5}, \quad \tilde{M} = W_2W_1^{-1}.$$

Since W_1 and W_2 are positive definite, so is M, which implies that M and $\tilde{M}$ are similar via:

$$M = W_1^{-0.5}\tilde{M}W_1^{-0.5}.$$

Let Λ^M and V^M be the eigenvalue and eigenvector matrices of M, respectively. Then the eigenvalue matrix of $\tilde{M}$ satisfies:

$$\Lambda^{\tilde{M}} = \Lambda^M,$$

with right and left eigenvectors given by:

$$V_R^{\tilde{M}} = W_1^{0.5}V^M, \quad V_L^{\tilde{M}} = W_1^{-0.5}V^M.$$

Thus, we obtain the matrix powers:

$$M^p = V^M\left(\Lambda^M\right)^p\left(V^M\right)^T,$$

$$\tilde{M}^p = W_1^{0.5}V^M\left(\Lambda^M\right)^p\left(V^M\right)^T W_1^{-0.5}.$$

From this, we conclude:

$$\tilde{M}^p = W_1^{0.5}M^pW_1^{-0.5}.$$

Therefore, we derive:

$$W_1^{0.5}M^pW_1^{0.5} = W_1^{0.5}M^pW_1^{-0.5}W_1 = \tilde{M}^pW_1.$$

Substituting back for M and $\tilde{M}$, we obtain:

$$W_1^{0.5}\left(W_1^{-0.5}W_2W_1^{-0.5}\right)^p W_1^{0.5} = \left(W_2W_1^{-1}\right)^p W_1. \tag{11}$$

□