SiBBlInGS: Similarity-driven Building-Block Inference using Graphs across States

Anonymous Authors[1]

Abstract

Time series data across scientific domains are often collected under distinct states (e.g., tasks), wherein latent processes (e.g., biological factors) create complex inter- and intra-state variability. A key approach to capture this complexity is to uncover fundamental interpretable units within the data, Building Blocks (BBs), which modulate their activity and adjust their structure across observations. Existing methods for identifying BBs in multi-way data often overlook inter- vs. intra-state variability, produce uninterpretable components, or do not align with properties of real-world data, such as missing samples and sessions of different durations. Here, we present a framework for Similarity-driven Building Block Inference using Graphs across States (SiBBlInGS). SiBBlInGS offers a graph-based dictionary learning approach for discovering sparse BBs along with their temporal traces, based on co-activity patterns and inter- vs. intra-state relationships. Moreover, SiBBlInGS captures per-trial temporal variability and controlled cross-state structural BB adaptations, identifies state-specific vs. state-invariant components, and accommodates variability in the number and duration of observed sessions across states. We demonstrate SiBBlINGS's ability to reveal insights into complex biological and medical phenomena through several synthetic and real-world examples. Specifically, we found that SiBBlInGS recovers meaningful functional neural ensembles underlying Macaque neural recordings and can leverage human EEG data to localize the source of epileptic seizures moments before their onset.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Under review at ICML 2024 AI for Science workshop

1. Introduction

The analysis of high-dimensional time-series is increasingly important across various scientific disciplines, ranging from neuroscience (Kala et al., 2009; Mudrik et al., 2024) to social sciences (Jerzak et al., 2023) to genetics (Bar-Joseph et al., 2012; Tanvir Ahmed et al., 2023). These data, however, present a daunting challenge in terms of comprehensibility as they are often highly heterogeneous. Specifically, data in many domains are gathered under multiple states (e.g., clinical interventions), while latent factors may introduce variability across trials within states (e.g., internal biological processes that lead to variations in patient responses to treatment). Current analysis methods often struggle to capture the full variability in such multi-state data. Additionally, integrating data from repeated observations (trials) under the same state into a coherent representation is often challenged by missing samples or variable trial duration and sampling rates (Goris et al., 2014; Charles et al., 2018; Duncker & Sahani, 2018). The common practice of within-state trial averaging, for example, obscures important patterns within individual trials.

A promising approach for analyzing multi-state data involves identifying fundamental representational units—Building Blocks (BBs)—whose composition remains similar across states, while their temporal profiles can modulate across trials to capture trial-to-trial variability, both within the same state and across states. These BBs can represent, for instance, neural ensembles in the brain; social groups in diverse contexts; gene clusters under regulatory mechanisms, etc. Identifying these BBs and understanding how they change across states is a key step for recognizing the latent processes underlying the data and providing valuable insights into core commonalities and differences among states. However, uncovering these BBs poses a challenge, as their individual activities or compositions are often unobservable. This challenge is further complicated by potential variations across states, not only in the temporal activity of these BBs, but also in the subtle structural adjustments of the BBs' compositions between states. For example, a neural ensemble may display varying temporal activity between normal (non-seizure) brain activity sessions and during seizures, along with subtle structural adaptations in

the ensemble composition during seizures (van den Berg & Friedlander, 2008), e.g., neurons that are not typically part of the ensemble might become involved during a seizure.

Here, we present SiBBlInGS, a graph-based data driven framework to unravel the complexities of high-dimensional multi-state time-series data, by unveiling its underlying sparse, similarity-driven BBs along with their temporal activity. Our main contributions include:

- We develop a novel framework to find interpretable hidden BBs underlying high-dimensional multi-way data while extracting their cross-trial temporal activities and inter-state structural variability, and address real-world challenges unmet by existing methods.
- We accommodate varying trial conditions, including different time durations, sampling rates, missing samples, and per-state counts, and enable overlapping BB composition.
- We highlight our method's promise by demonstrating its ability to recover ground-truth components in synthetic data and meaningful latents in several real-world examples.

2. Background and Related Work

In the case of single-trial analysis, methods for identifying BBs often rely on matrix decomposition including Singular Value Decomposition (SVD) (Kogbetliantz, 1955), Principal Components Analysis (PCA) (Hotelling, 1933), Independent Components Analysis (ICA) (Hyvarinen et al., 2001), or Non-negative Matrix Factorization (NMF) (Lee & Seung, 1999), where sparsity constraints can be added to improve interpretability (e.g., sparse PCA (SPCA)) (Zou et al., 2006). Extending these to the multi-trial setting can be addressed by either concatenating trials end-to-end to create a single wide matrix or by applying these methods individually to each trial. However, this either overlooks the temporal scales of the data (within trial and cross-trial) or ignores shared factors across trials.

A more suitable extension to multi-trial observations is tensor methods (TF, e.g., PARAFAC and Tucker decomposition) (Harshman, 1970; Williams et al., 2018; Mishne et al., 2016; De Lathauwer et al., 2000; Wu et al., 2018), which consider the trials as an additional dimension of the data. However, none of these methods or their combinations with Gaussian processes (GP) (Tillinghast et al., 2020; Xu et al., 2011; Zhe et al., 2016) and dynamic information (e.g., NNDTN and NONFAT (Wang & Zhe, 2022b)) cannot naturally handle variability in trial duration or address state variability as a fourth dimension. Extensions of TF methods seek either identical BBs across states with flexible temporal patterns or fixed temporal traces across states with flexible cross-state BBs. For example, the Shared Response Model (SRM) (Chen et al., 2015) models a multi-subject fMRI model with the same temporal activity across all individuals with different spatial responses between subjects. SRM, however, requires that components be orthogonal, which may not align with biological plausibility. Hyperalignment (HA) (Haxby et al., 2011) addresses a similar setting as SRM by rotating the subjects' time series responses to optimize inter-subject correlation. However, HA does not explicitly reduce the dimension of the feature space.

Other existing methods, such as Dynamic Mode Decomposition (DMD) (Schmid, 2010), model the temporal dynamics explicitly as dynamical systems, however, these methods are tailored for 2D analysis and thus are not designed to simultaneously model multi-state, multi-trial data. State-Space Models (SSMs) (Auger-Méthé et al., 2021) represent another approach to explore time-series data by describing the latent states evolution by a state-transition matrix; however, they do not aim to find sparse interpretable ensembles with cross-trial structural and temporal variability. Other methods include demixed PCA (dPCA) (Brendel et al., 2011), Targeted Dimensionality Reduction (TDR) (Mante et al., 2013) and model-based TDR (mTDR) (Aoi & Pillow, 2018; Aoi et al., 2020). The latter two directly regress rank-1 (TDR) or low-rank (mTDR) components that explicitly target task-relevant variables. However, TDR/mTDR similarly cannot handle trials of varying duration and do not incorporate sparsity in the identified ensembles. dPCA falls short in addressing missing data, different trial durations, and varied sampling rates.

Closer to our approach, dictionary learning (Olshausen & Field, 2004; 1996; Aharon et al., 2006), provides more interpretable representations (Tošić & Frossard, 2011) by learning a feature dictionary where each data point can be linearly reconstructed using only a few of the feature vectors. While traditional dictionary learning treats each data point as independent, recent advances based on re-weighted ℓ_1 (Candes et al., 2008; Garrigues & Olshausen, 2010) can account for spatio-temporal similarities in the sparse feature representations between data points (Garrigues & Olshausen, 2010; Charles & Rozell, 2013; Charles et al., 2016; Zhang & Rao, 2011; Qin et al., 2017; Mishne & Charles, 2019). In particular, re-Weighted ℓ_1 Graph Filtering (RWL1-GF) (Charles et al., 2022) was recently developed for demixing fluorescing components in calcium imaging recordings by correlating the sparse decompositions across a data-driven graph defined by pixel similarity. While RWL1-GF proves the efficacy of graph embeddings in extracting meaningful features, it is constrained to single-trial data and confines its graph construction to a single dimension of the data—the pixel space—overlooking possibly meaningful structures in other dimensions.

Fuzzy clustering (Yang, 1993; Wei et al., 2020; He et al.,

2018) allows data points to exhibit varying degrees of membership in multiple clusters, addressing limitations of methods that restrict data points to a single BB. However, these approaches focus solely on BB structures rather than their temporal activities, and do not integrate within-state and between-state variability information.

Nonlinear deep learning models have also been developed to extract latent factors, however, these typically lack interpretability in mapping back to the feature/sample space, are limited in their ability to produce sparse latent factors, and require large amounts of training data. Variational Autoencoders (VAEs) (e.g., (Xu & Durrett, 2018; Tillinghast & Zhe, 2021)) offer recovery of nonlinear latent low-dimensional representations; however, they do not naturally consider the data's temporal structure, and their elements do not directly represent the contribution of individual channels from the input space. Sparse (Ashman et al., 2020; Barello et al., 2018) or dynamical (Girin et al., 2020) variants of VAEs do not consider within-vs between state variability of the latent representations. Transformer models, e.g., (Liu et al., 2022) jointly model individual and collective dynamics via an individual module for each of several component dynamics and an interaction module that captures pairwise interactions. However, this necessitates prior knowledge about the system's separation and requires large-scale data due to the encoder-decoder architecture. Recently, for neuronal data analysis, CEBRA (Schneider et al., 2023) incorporates auxiliary labels and temporal information in contrastive optimization, however, it produces a latent state that requires additional interpretation steps to connect to the neuronal space.

Notably, all the above approaches are constrained in their capacity to identify fundamental hidden sparse components while capturing multi-state, multi-trial variability.

3. Problem Definition and Notations

Consider a system with N channels organized into at most p BBs, with each BB representing a group of channels with shared functionality. These BBs serve as the fundamental constituents of a complex process, however their composition is not directly observed nor explicitly known. In particular, let the columns of $\boldsymbol{A} \in \mathbb{R}^{N\times p}$ represent the BBs, such that $\boldsymbol{A}_{ij}$ is the contribution of the i-th channel to the j-th BB, with $\boldsymbol{A}_{ij} = 0$ indicating that channel i does not belong to BB j. We assume that each channel can belong to multiple BBs, and that each BB is sparse (i.e., $\|\boldsymbol{A}_{:j}\|_0 = K << N$ for all $j = 1 \dots p$).

First, we consider a single trial of the system $\boldsymbol{Y} \in \mathbb{R}^{N\times T}$ over T time points. During this trial, each BB exhibits temporal activity denoted by $\boldsymbol{\Phi} \in \mathbb{R}^{T\times p}$, that might reflect current hidden properties of the system, where $\boldsymbol{\Phi}_{t,j}$ is the activity of the j-th BB at time t. These temporal profiles are assumed to be smooth, bounded (i.e., $||\boldsymbol{\Phi}||_F < \epsilon_1$, for some ϵ_1), and have a low correlation between distinct BBs' activity (i.e., $\rho(\boldsymbol{\Phi}_{:j}, \boldsymbol{\Phi}_{:i}) < \epsilon_2 \quad \forall i \neq j$, for some ϵ_2). In this single trial, our observations, $\boldsymbol{Y}$, arise from the collective activity of all BBs operating together, $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{\Phi}^T + \eta$, where η denotes *i.i.d.* Gaussian observation noise. However, the individual composition ($\boldsymbol{A}$) or activity ($\boldsymbol{\Phi}$) of each BB is unknown.

In the more general setting, we observe a set of M trials, $\{\boldsymbol{Y}_m\}_{m=1}^M$, where the duration of each trial $m = 1 \dots M$ may vary, i.e., $\boldsymbol{Y}_m \in \mathbb{R}^{N\times T_m}$. The BBs ($\boldsymbol{A}$) remain constant across trials while their corresponding temporal activity ($\boldsymbol{\Phi}_m \in \mathbb{R}^{T_m\times p}$) may vary across trials to capture trial-to-trial variability. The setting we focus on extends beyond a single set of trials; instead, we deal with a collection of D such multi-trial sets, each associated with a known state $d = 1 \dots D$ (Fig. 1A). Across these sets, both the number of trials per set (M_d for each set d), and the durations of the trials, may vary. Thus, the full observation dataset includes the collection of D multi-trial sets, $\{\boldsymbol{Y}_m^1\}_{m=1}^{M_1}, \dots \{\boldsymbol{Y}_m^D\}_{m=1}^{M_D}$, each representing a different state $d = 1 \dots D$, such that $\boldsymbol{Y}_m^d = \boldsymbol{A}^d(\boldsymbol{\Phi}_m^d)^T + \eta_m^d$.

We assume that the BBs' temporal activities ($\{\boldsymbol{\Phi}_m^d\}$) can vary between trials, both within and between states, and the compositions of the BB ($\{\boldsymbol{A}^d\}$) might present subtle controlled adaptations between, but not within, states. Specifically, we posit that the BBs' dissimilarity between any pair of distinct states d and d' reflects the dissimilarity between those states, such that the distance between $\boldsymbol{A}^d$ and $\boldsymbol{A}^{d'}$ is constrained by $\|\boldsymbol{A}^d - \boldsymbol{A}^{d'}\|_F < \epsilon_3(d, d')$ for some threshold $\epsilon_3(d, d')$ determined by the application. For example, if considering different disease stages as states, we assume that consecutive disease stages are more similar to each other than to a healthy state, such that $\epsilon_3(d_{\text{disease}_1}, d_{\text{disease}_2}) < \epsilon_3(d_{\text{healthy}}, d_{\text{disease}})$. The main challenge that SiBBlInGS addresses is recovering the unknown BBs ($\boldsymbol{A}^d$) and their temporal activities ($\boldsymbol{\Phi}_m^d$) for all states and trials given only their combined simultaneous activity (Fig. 1A,B).

4. SiBBlInGS

In this section, we present SiBBlInGS—our framework to identify interpretable BBs along with their temporal traces based on shared activation patterns across trials and states. Unlike existing methods, SiBBlInGS identifies BBs in high-dimensional data based on temporal similarity without assuming orthogonality, enables BB interdependency or overlap, and can tackle trials of different duration, sampling rates, or count per state (Tab. 2). SiBBlInGS also offers both expert-based supervised and data-driven unsupervised approaches for integrating inter-state similarities, thus pro-

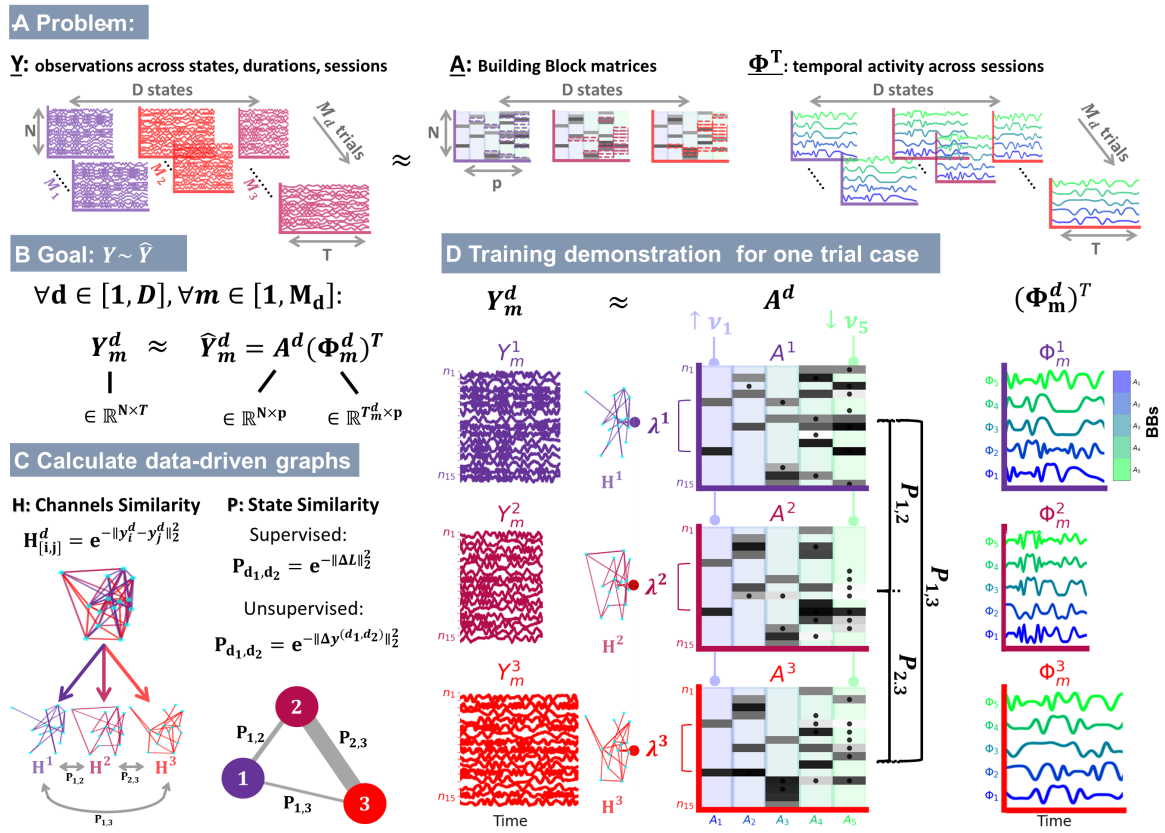

Figure 1. **SiBBlInGS Schematic A** SiBBlInGS adapts to real-world datasets with varying session durations, sampling rates, and state-specific data by learning interpretable graph-driven hidden patterns and their temporal activity. **B** SiBBlInGS is based on a per-state-and-trial matrix factorization where the BBs ($\boldsymbol{A}^d$) are identical across trials and similar across states. **C** SiBBlInGS controls the BB similarity via data-driven channel graphs ($\boldsymbol{H}^d \in \mathbb{R}^{N\times N}$) and state-similarity graph ($\boldsymbol{P} \in \mathbb{R}^{D\times D}$), which can be either predefined (supervised) or data-driven. **D** The learning schematic with an exemplary trial for each of the 3 exemplary states. The BBs of each state d (columns of $\mathbf{A}^d$) are constrained with two regularization terms: 1) state-specific $\boldsymbol{\lambda}^d$ captures similar activity between channels by leveraging the channel-similarity graph $\boldsymbol{H}^d$, and 2) $\boldsymbol{P}$, captures BB consistency across states via the state similarity graph. $\boldsymbol{\nu}$ controls the relative level of cross-state similarity between BBs, allowing the discovery of both background and state-specific BBs. Higher (lower) $\boldsymbol{\nu}$ values promote greater (lesser) consistency of specific BBs across states (e.g. $\boldsymbol{\nu}_1$ v.s $\boldsymbol{\nu}_5$).

viding the flexibility to choose between data-driven or predefined approaches based on the specific data structure.

SiBBlInGS is based on an extended dictionary learning-like procedure that alternates between updating the BBs ($\{\boldsymbol{A}^d\}$) and their temporal profiles ($\{\boldsymbol{\Phi}_m^d\}$) for all states. Critical to our approach is the integration of the non-linear similarities between both channels and states. We capture these relationships via two graphs (Fig. 1C), one over channels to identify cross-channel regularities, and one over states to promote cross-state similarity in BB structure. Mathematically, we formulate the fit $\{\widehat{\boldsymbol{A}}^d\}, \{\widehat{\boldsymbol{\Phi}}_m^d\}$ for all $d = 1 \dots D$ and $m = 1 \dots M_d$ by minimizing the cost function

$$\min_{\{\boldsymbol{A}^d\},\{\boldsymbol{\Phi}_m^d\}} \sum_d^D \left(\sum_m^{M_d} \left[\|\boldsymbol{Y}_m^d - \boldsymbol{A}^d (\boldsymbol{\Phi}_m^d)^T\|_F^2 + \mathcal{R}(\boldsymbol{\Phi}_m^d) \right] + \mathcal{R}(\boldsymbol{A}^d) + \sum_{d' \neq d}^D P_{d,d'} \|(\boldsymbol{A}^d - \boldsymbol{A}^{d'})\boldsymbol{V}\|_F^2 \right)$$

where the first term is a data fidelity term, and the second term regularizes the BBs' temporal traces. The term $\mathcal{R}(\boldsymbol{A}^d)$ regularizes each BB to be a sparse group of channels based on shared temporal patterns, and the last term regularizes the BBs' similarity across states (Fig. 1). The use of $\boldsymbol{V} = \text{diag}(\boldsymbol{\nu}) \in \mathbb{R}^{p\times p}$, accompanied by the weight vector $\boldsymbol{\nu} \in \mathbb{R}^p$, allows assigning varying weights to cross-state BBs' similarities to facilitate the discovery of state-invariant vs. state-specific BBs. SiBBlInGS iteratively updates $\boldsymbol{A}^d$ and

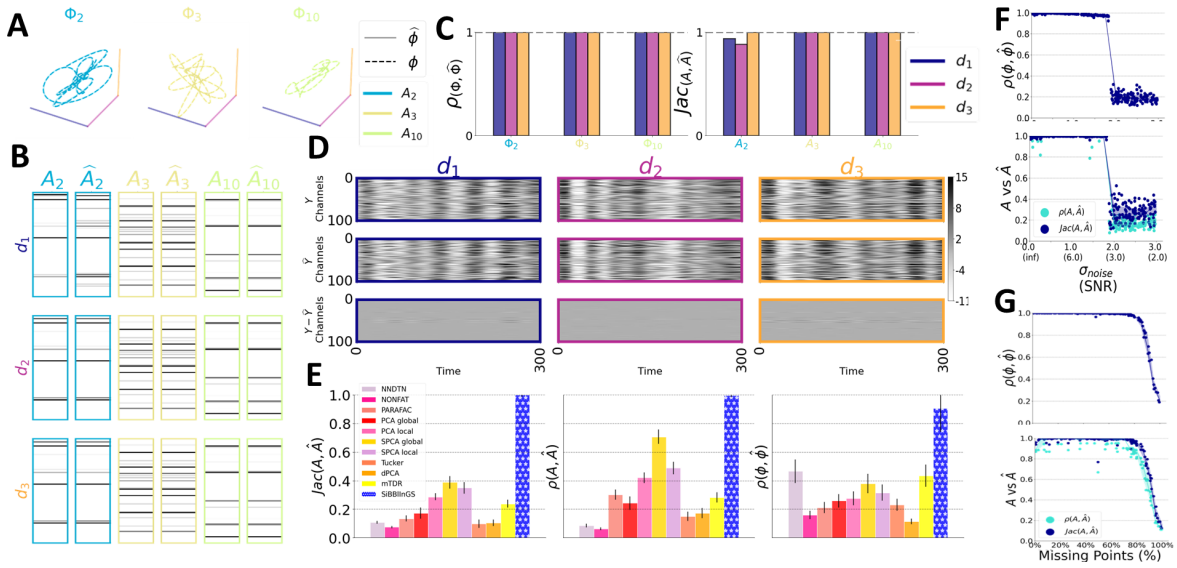

Figure 2. **Synthetic data results. A** Three example time traces identified by SiBBlInGS vs. ground truth traces, projected into the three synthetic states. SiBBlInGS recovers both traces that are highly correlated with specific states (e.g., $\mathbf{\Phi}_{10}$; green), as well as traces that exhibit similar activation across states (e.g., $\mathbf{\Phi}_2$; blue). **B** Comparison between the identified example BBs and the ground-truth BBs. **C** Correlation between the example identified time traces and the ground truth (left), and Jaccard index of the identified BBs compared to the ground truth (right). **D** Comparison between the ground-truth data (top), SiBBlInGS reconstruction (middle), and the residual data (bottom). **E** Comparison to baseline methods (Sec. 5, App. L). **F** Performance under noise and random initializations (300 repetitions). Each dot is a model instance. The curve shows the median values, and the shading corresponds to the 25%-75% percentiles. While SiBBlInGS remains robust under varying noise ($\sigma_{\text{signal}}/\sigma_{\text{noise}} > 3$), it experiences a phase transition at a specific noise level, aligning with the dictionary-learning literature (e.g. (Studer & Baraniuk, 2012)). **G** Performance with increasing levels of missing samples (200 repeats). The scattered dots represent model repetitions, the curves depict the median values calculated by rounding to the nearest 5%, and the background shading corresponds to 25%-75% percentiles.

$\mathbf{\Phi}^d_m$ for each trial and state (Alg. 1 and Fig. 1; computational complexity is described in App. E), as described below:

Updating $\boldsymbol{A}^d$: Since we assume that BBs may require subtle state-to-state adaptations but must remain constant within a state, SiBBlInGS demands that the BB matrix ($\boldsymbol{A}^d$) is shared across same-state trials but undergoes subtle adjustments across states, proportionate to the corresponding states' similarities ($\boldsymbol{P}$). The update of $\boldsymbol{A}^d$ for each state d, is achieved via an extended re-weighted ℓ_1 graph filtering with an integration of the channel-similarity graph, $\boldsymbol{H} \in \mathbb{R}^{N\times N}$, in a way that promotes channels with similar temporal activity to be grouped into the same BB. In each updating iteration of $\boldsymbol{A}^d$, as a pre-calculation step, we first horizontally concatenate the observations from all M_d trials of that d state to receive the matrix $\boldsymbol{Y}^{d*} \in \mathbb{R}^{N\times(\sum_{m=1}^{M_d} T^d_m)}$, and vertically concatenate the last estimates of the temporal traces from all trials of that state to build the matrix $\mathbf{\Phi}^{d*} \in \mathbb{R}^{(\sum_{m=1}^{M_d} T^d_m)\times p}$.

We then update each row $n = 1 \dots N$ of $\widehat{\boldsymbol{A}}^d$ ($\widehat{\boldsymbol{A}}^d_{n:}$) via a re-weighted procedure that alternates between updating $\widehat{\boldsymbol{A}}^d_{nj}$ and $\boldsymbol{\lambda}^d_{n,j}$:

$$\widehat{\boldsymbol{A}}^d_{n:} = \arg\min_{\boldsymbol{A}^d_{n:}} \|\boldsymbol{Y}^{d*}_n - \boldsymbol{A}^d_{n:}(\mathbf{\Phi}^{d*})^T\|_2^2 + \sum_{j=1}^{p} \boldsymbol{\lambda}^d_{n,j}|\boldsymbol{A}^d_{n,j}| + \sum_{d'\neq d} \boldsymbol{P}_{dd'}\|(\boldsymbol{A}^d_{n:} - \boldsymbol{A}^{d'}_{n:}) \circ \boldsymbol{\nu}\|_2^2, \quad (1)$$

and

$$\boldsymbol{\lambda}^d_{n,j} = \frac{\epsilon}{\beta + |\widehat{\boldsymbol{A}}^d_{n,j}| + w_{\text{graph}}|\boldsymbol{H}^d_{n:}\widehat{\boldsymbol{A}}^d_{:j}|}. \quad (2)$$

Above, $\circ$ is element-wise multiplication, and β, ϵ, and w_{graph} are model hyper-parameters. The matrices $\boldsymbol{H}^d$ and $\boldsymbol{P}^d$ are channel and state similarity graphs, described below:

State similarity graph: $\boldsymbol{P} \in \mathbb{R}^{D\times D}$ is a state-similarity graph that determines the effect of the similarity between each pair of states on the regularization of the distance between their BB representations. $\boldsymbol{P}$ can be set manually (supervised $\boldsymbol{P}$) or in a data-driven way (unsupervised $\boldsymbol{P}$), thus allowing selection based on specific goals, data type, and knowledge of data labels (Fig. 1C). Each of these two options offers unique benefits: the supervised variant enables explicit regulation of the similarity and the incorporation of

human-expert knowledge into the model, whereas the unsupervised variant leverages the data itself to learn similarities and patterns without preconceived biases.

Here, we present the supervised version of $\boldsymbol{P}$, which is particularly useful when one has prior knowledge or expectations about quantitative state values that can be leveraged to integrate desired information into the model, while the data-driven approach is presented in Appendix B.2. This supervised version, unlike the data-driven option, assumes that a numerical label $\boldsymbol{L}_d$, associated with each state d, can provide valuable information for constructing the state-similarity graph $\boldsymbol{P}$ (e.g., vector labels that denote x-y coordinates in a movement task). In this way, the similarity $\boldsymbol{P}_{d,d'}$ between each pair of states (d, d') is calculated based on the distance between the labels ($\boldsymbol{L}_d$, $\boldsymbol{L}_{d'}$) associated with these states: $\boldsymbol{P}_{d,d'} = \exp\left(-\|\boldsymbol{L}_d - \boldsymbol{L}_{d'}\|_2^2/\sigma_{\boldsymbol{P}}^2\right)$ where σ_P^2 controls how the similarities in labels scale to similarities in BBs. The supervised approach easily extends to both data with identical or different session durations and can also handle categorical states as described in the Appendix B.1.1.

Channel similarity graphs: $\boldsymbol{H}^d \in \mathbb{R}^{N\times N}$ is the channel graph for each state d and is calculated by $\boldsymbol{H}^d_{i,j} = \exp\left(-\|\boldsymbol{Y}^{d*}_{i:} - \boldsymbol{Y}^{d*}_{j:}\|_2^2/\sigma^2_{\boldsymbol{H}}\right)$, where $\sigma_{\boldsymbol{H}}$ is an hyperparameter that controls the kernel bandwidth and $\boldsymbol{Y}^{d*}_{j:}$ is the horizontally concatenated observations under state d described before. To enhance the robustness of $\{\boldsymbol{H}^d\}_{d=1}^D$, we add a post-processing step and utilize the state-graph ($\boldsymbol{P}$) to re-weigh each $\boldsymbol{H}^d$ along the states dimension: $\boldsymbol{H}^d \leftarrow \sum_{d=1}^D \boldsymbol{P}_{d,d'}\boldsymbol{H}^{d'}$. We then retain only the k largest values in each row, setting the rest to zero, symmetrize, and row normalize $\boldsymbol{H}^d$ so that each row sums to one (App. C). This process mitigates the influence of outliers and encourages the clustering of similarly-behaving channels into the same BB. The advantage of graph-driven re-weighting, compared to other TF and dictionary learning procedures, is that the updated weighted regularization ($\boldsymbol{\lambda}^d \in \mathbb{R}^{N\times p}$) promotes the grouping (separating) of channels with similar (dissimilar) activity into the same (different) BBs by integrating the channel similarity graph $\boldsymbol{H}^d$ into the regularization. Specifically, in the last term of the $\boldsymbol{\lambda}^d_{n,j}$'s denominator, for a given state d, a strong (weak) correlation between the temporal neighbors of the n^{th} channel (captured by $\boldsymbol{H}^d_{n:}$) and the members of the j-th BB ($\widehat{\boldsymbol{A}}^d_{:j}$) results in a decreased (increased) $\boldsymbol{\lambda}^d_{n,j}$. Consequently, ℓ_1 regularization on $\widehat{\boldsymbol{A}}^d_{n:}$ is reduced (increased)—promoting the inclusion (exclusion) of each channel into BBs that include (exclude) its temporal neighbors.

After each update of all rows in $\boldsymbol{A}^d$, each column is normalized to have a maximum absolute value of 1. In practice, we update $\boldsymbol{A}$ (Eq. (1)) for a random subset of trials in each iteration to improve robustness and computational speed.

Updating $\boldsymbol{\Phi}^d_m$: The update step over $\boldsymbol{\Phi}^d_m$ uses the current estimate of $\boldsymbol{A}^d$ to re-estimate the temporal profile matrix $\boldsymbol{\Phi}^d_m$ independently over each state d and trial m. Note that we do not enforce cross-trial similarity in $\boldsymbol{\Phi}^d_m$ to allow for flexibility in capturing trial-to-trial variability both within and across states. Thus, for each trial m and state d, $\boldsymbol{\phi} = \boldsymbol{\Phi}^d_m$ is updated by solving:

$$\widehat{\boldsymbol{\phi}} = \arg\min_{\boldsymbol{\phi}\geq 0} \left\|\boldsymbol{Y}^d_m - \boldsymbol{A}^d\boldsymbol{\phi}^T\right\|_F^2 + \gamma_1\|\boldsymbol{\phi}\|_F^2 + \gamma_2\|\boldsymbol{\phi} - \widehat{\boldsymbol{\phi}}^{\text{iter}-1}\|_F^2 + \gamma_3\|\boldsymbol{\phi} - \boldsymbol{\phi}^{t-1}\|_F^2 + \gamma_4\mathcal{R}_{\text{corr}}(\boldsymbol{\phi}) \quad (3)$$

where the first term preserves data fidelity, the second term regularizes excessive activity, the third term encourages continuity across iterations ($\widehat{\boldsymbol{\phi}}^{\text{iter}-1}$ is $\boldsymbol{\phi}$ from the previous iteration), and the fourth term is a diffusion term that promotes temporal consistency of the dictionary ($\boldsymbol{\phi}^{t-1}$ is $\boldsymbol{\phi}$ shifted by one time point). $\mathcal{R}_{\text{corr}}(\boldsymbol{\phi}) = \|\left(\boldsymbol{\phi}^T\boldsymbol{\phi} - \text{diag}(\boldsymbol{\phi}^T\boldsymbol{\phi})\right)\circ\boldsymbol{D}\|_{\text{sav}}$ promotes decorrelation of distinct temporal traces, where sav is sum-of-absolute-values and $\boldsymbol{D}\in\mathbb{R}^{p\times p}$ is a normalization matrix with $\boldsymbol{D}_{ij} = \frac{1}{\|\boldsymbol{\phi}_{:i}\|_2\|\boldsymbol{\phi}_{:j}\|_2}$ (App. D).

5. Experiments

SiBBlInGS recovers ground truth BBs in synthetic data: Synthetic data were generated with $D = 3$ states, each consisting of a single trial, with $p = 10$ ground-truth BBs, and $N = 100$ channels. Each i-th BB was generated with a maximum cardinality of $\max_{d,i}\|A^d_{:,i}\|_0 = 21$ channels, and on average each channel was associated with 2.1 BBs. While the BBs were designed to be non-orthogonal, we constrained their pairwise correlations to be below a threshold of $\max\rho < 0.6$. The temporal dynamics of the synthetic data were generated by summing 15 trigonometric functions with different frequencies (App. H.2 for details).
SiBBlInGS demonstrated monotonically improving performance during training (Fig. 6A-D), and at convergence was able to successfully recover the underlying BBs in the synthetic data and their temporal traces (Fig 2A-C). In particular, example traces demonstrate a high precision of the recovered temporal traces, with correlation to the ground truth traces being close to one (Fig. 2A, C, 6F). Furthermore, the identified BB components align closely with the ground-truth BBs (Fig. 2B,C), as indicated by high Jaccard index values. We compared SiBBlInGS to existing methods, including Tucker Decomposition, PARAFAC, (S)PCA "global" (a single (S)PCA for all states), (S)PCA "local" ((S)PCA for each state), NONFAT, NNDTN, mTDR, and dPCA, with details in App. L). Notably, SiBBlInGS outperforms existing baselines both in terms of identifying the ground truth BBs and their traces (Fig. 2E, 6F, 8).

SiBBlInGS finds interpretable BBs in Google Trends: We used Google Trends to demonstrate SiBBlInGS' capabil-

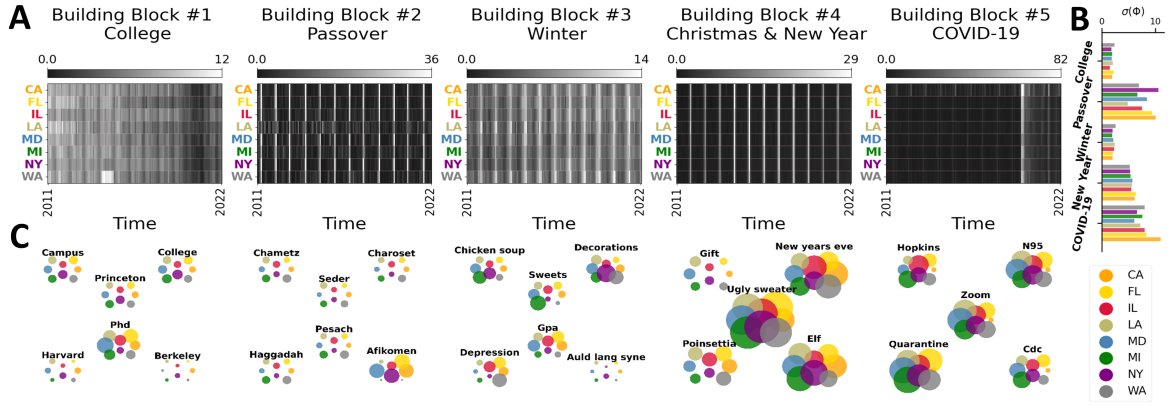

Figure 3. **Demonstration on Google Trends Data. A** The BBs' temporal traces, as SiBBlInGS found, demonstrate seasonal trends consistent with the terms associated with each BB. **B** Standard deviation of temporal traces over time for the different states align with variability in the states' demographics (Sec. 5). **C** The BBs SiBBlInGS identified along with their per-state dominancy produce more meaningful clusters than baselines (Fig. 10). States are marked by colors; dot sizes represent the contribution of a term in the BB.

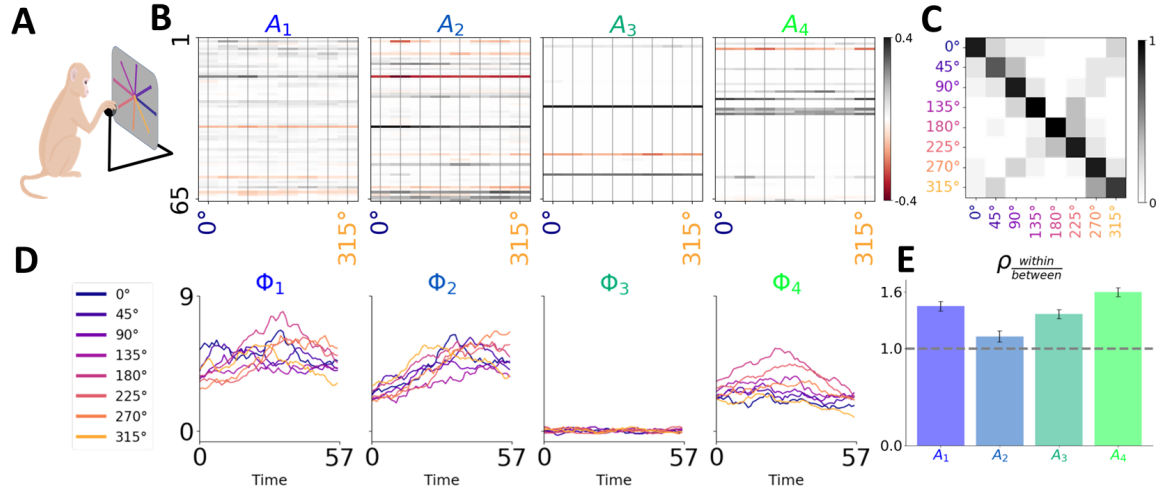

Figure 4. **Identification of Temporal Patterns in Monkey Somatosensory Cortex. A** The reaching out task ((Rodriguez, 2023)). **B** Sparse clusters of neurons representing the identified BBs. **C** Confusion matrix of a multi-class logistic regression model using the inferred temporal traces to predict the state label. **D** The BBs' temporal traces as they vary across states and time. **E** Ratios of within-to-between states temporal correlations for each BB, with $\frac{\rho_{\text{within}}}{\rho_{\text{between}}} > 1$, indicating states distinguishability.

ity in identifying temporal and structural patterns by querying search term frequency on Google over time. We used normalized monthly Trends volume of 44 queries (from Jan. 2011 to Oct. 2022) related to various topics, as searched in 8 US states selected for their diverse characteristics (Coulby, 2000) (see pre-processing in App. I.1). The $p = 5$ BBs identified by SiBBlInGS reveal meaningful clusters of terms, whose time traces convey the temporal evolution of user interests per region (Fig. 3A), while aligning with the seasonality of the BBs' components. For instance, the first BB represents college-related terms and shows a gradual annual decrease with periodic activity and a notable deviation during the COVID pandemic, possibly reflecting factors such as the shift to remote learning (Fig. 3A, 11). The second and third BBs, respectively, demonstrate periodic patterns associated with Passover in April (Fig. 12) and winter terms in December. Interestingly, CA, FL, MD, and NY—all states with larger Jewish populations*—show more pronounced peaks of the "Passover" BB activity in April (when Passover is celebrated) compared to the other states (Fig. 3B and 12). The last BB represents COVID-related terms and exhibits temporal patterns with a sharp increase around Jan. 2020, coinciding with the onset of the COVID pandemic in the US. Remarkably, 'Hopkins' exhibits a less pronounced COVID-related search peak in MD (blue), where the university and hospital are located, likely attributed to its well-established local presence (Fig. 3C, right). Conversely, other states witnessed a more significant surge in Hopkins-related searches

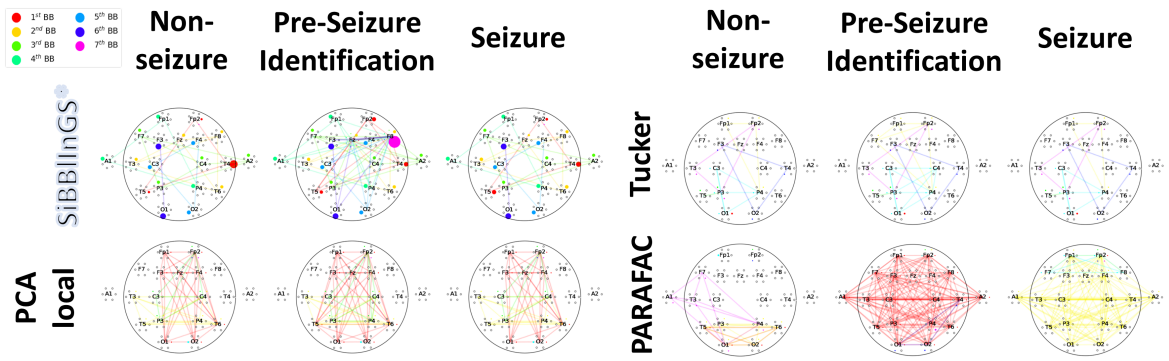

Figure 5. **Emerging local BBs in Epilepsy**. The recovered BBs under 1) normal activity, 2) activity during the 8 seconds proceedings CPS seizures located around the F8 area, and 3) activity during the seizures. Colors represent different BBs, and the size of the dots corresponds to the contribution of the respective electrode to each BB.

at the onset of the COVID outbreak, as Hopkins suddenly garnered increased attention during this period. This emphasizes our model's interpretability and the need to capture similar yet distinct BBs across states. Other methods applied with the same number of BBs ($p = 5$) as used in SiBBlInGS produced less meaningful BBs (Fig. 10).

SiBBlInGS identifies meaningful patterns in brain recordings: We tested SiBBlInGS on neural activity recorded in the somatosensory cortex in a monkey performing a reaching task, as described by (Chowdhury & Miller, 2022). The data we used include 18 trials under each of the 8 hand directions, with each direction corresponding to a unique state. The raw spike data were convolved over time with a Gaussian kernel. When applying SiBBlInGS with a maximum of $p = 4$ BBs, it identified sparse functional BBs (Fig. 4B) along with meaningful temporal traces (Fig. 4D) that exhibit state-specific patterns. Interestingly, the third BB consistently shows minimal activity across all states, suggesting that it captures background or noise activity. The structure of the identified BBs exhibits subtle yet significant adaptations across states in terms of neuron weights and BB assignments. Furthermore, SiBBlInGS finds neurons belonging to multiple neural clusters, suggesting their involvement in multiple functions. When examining the temporal correlations of the corresponding BBs within and between states, all BBs exhibited a within/between correlations ratio > 1 (Fig. 4E, 13C, App. J.4) indicating robust within state trajectories and distinctions between states. Furthermore, multi-class logistic regression based only on the identified temporal traces accurately predicted the states (Fig. 4C).

SiBBlInGS discovers emerging BBs preceding seizure: We applied SiBBlInGS to EEG recordings of an epileptic patient from (Handa et al., 2021; Nasreddine, 2021) (App. K). We examined data from an 8-year-old individual who had experienced 5 complex partial seizures (CPS) localized around electrode F8. SiBBlInGS unveiled interpretable and localized EEG activity in the period preceding seizures (Fig. 5), a feat not achieved by other methods. It identified a BB specific to the region around the clinically labeled area (F8) that emerged during the 8 seconds prior to the seizure (Fig. 5, pink circle in SiBBlInGS's middle). Additionally, it found several alterations in the BB composition during the seizure in comparison to the normal activity. E.g., the contribution of T4 to the red BB during normal activity is higher than its contribution during a seizure, while the contribution of T5 to the same BB is larger during a seizure. This underscores the potential of SiBBlInGS in discovering BBs that uniquely emerge under specific states, made possible by the flexibility of $\boldsymbol{\nu}$ to support both state-variant and state-invariant BBs.

6. Conclusion

We propose SiBBlInGS for graphs-driven identification of interpretable cross-state BBs with their temporal profiles in multi-way time-series data—providing insights into system structure and variability. Unlike other approaches, SiBBlInGS supports the discovery of BBs with subtle changes in cross-state structures, allows each channel to belong to a few with varying contributions, and promotes the discovery of both state-invariant and state-specific BBs. We demonstrate SiBBlInGS' capacity to identify functional neural ensembles and discern cross-state variations in web-search data structures, showcasing its promise in additional domains, including, e.g., detecting gene expression clusters in health vs. disease, unveiling local financial patterns based on stock data, and more. Regarding limitations, SiBBlInGS assumes Gaussian statistics, yet Poisson may sometimes be more suitable. Additionally, exploring advanced distance metrics for the graphs construction holds promise for future research. Finally, the identified BBs currently do not consider potential directed connectivity, presenting an exciting future direction.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning and Time Series Data Analysis. There are several potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Aharon, M., Elad, M., and Bruckstein, A. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322, 2006.

Aoi, M. and Pillow, J. W. Model-based targeted dimensionality reduction for neuronal population data. *Advances in Neural Information Processing Systems*, 31, 2018.

Aoi, M. C., Mante, V., and Pillow, J. W. Prefrontal cortex exhibits multidimensional dynamic encoding during decision-making. *Nature Neuroscience*, 23(11):1410–1420, 2020.

Ashman, M., So, J., Tebbutt, W., Fortuin, V., Pearce, M., and Turner, R. E. Sparse gaussian process variational autoencoders. *arXiv preprint arXiv:2010.10177*, 2020.

Auger-Méthé, M., Newman, K., Cole, D., Empacher, F., Gryba, R., King, A. A., Leos-Barajas, V., Mills Flemming, J., Nielsen, A., Petris, G., et al. A guide to state–space modeling of ecological time series. *Ecological Monographs*, 91(4):e01470, 2021.

Bar-Joseph, Z., Gitter, A., and Simon, I. Studying and modelling dynamic biological processes using time-series gene expression data. *Nature Reviews Genetics*, 13(8): 552–564, 2012.

Barello, G., Charles, A. S., and Pillow, J. W. Sparse-coding variational auto-encoders. *BioRxiv*, pp. 399246, 2018.

Berndt, D. J. and Clifford, J. Using dynamic time warping to find patterns in time series. In *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, pp. 359–370. AAAI Press, 1994.

Brendel, W., Romo, R., and Machens, C. K. Demixed principal component analysis. *Advances in neural information processing systems*, 24, 2011.

Candes, E. J., Wakin, M. B., and Boyd, S. P. Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier analysis and applications*, 14:877–905, 2008.

Charles, A. S. and Rozell, C. J. Spectral superresolution of hyperspectral imagery using reweighted ℓ_1 spatial filtering. *IEEE Geoscience and Remote Sensing Letters*, 11 (3):602–606, 2013.

Charles, A. S., Balavoine, A., and Rozell, C. J. Dynamic filtering of time-varying sparse signals via ℓ_1 minimization. *IEEE Transactions on Signal Processing*, 64(21): 5644–5656, 2016.

Charles, A. S., Park, M., Weller, J. P., Horwitz, G. D., and Pillow, J. W. Dethroning the fano factor: a flexible, model-based approach to partitioning neural variability. *Neural Computation*, 30(4):1012–1045, 2018.

Charles, A. S., Cermak, N., Affan, R. O., Scott, B. B., Schiller, J., and Mishne, G. Graft: Graph filtered temporal dictionary learning for functional neural imaging. *IEEE Transactions on Image Processing*, 31:3509–3524, 2022.

Chen, P.-H. C., Chen, J., Yeshurun, Y., Hasson, U., Haxby, J., and Ramadge, P. J. A reduced-dimension fmri shared response model. In Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

Chowdhury, R. and Miller, L. Area2_bump: macaque somatosensory area 2 spiking activity during reaching with perturbations. Data set, 2022.

Chowdhury, R. H., Glaser, J. I., and Miller, L. E. Area 2 of primary somatosensory cortex encodes kinematics of the whole arm. *Elife*, 9:e48198, 2020.

Coulby, D. *Beyond the national curriculum: Curricular centralism and cultural diversity in Europe and the USA*. Psychology Press, 2000.

De Lathauwer, L., De Moor, B., and Vandewalle, J. Multilinear subspace learning: dimensionality reduction of multidimensional data. In *Proceedings of the 2000 Conference on Advances in Neural Information Processing Systems*, pp. 485–491, 2000.

Duncker, L. and Sahani, M. Temporal alignment and latent gaussian process factor inference in population spike trains. *Advances in Neural Information Processing Systems*, 31, 2018.

Garrigues, P. and Olshausen, B. Group sparse coding with a Laplacian scale mixture prior. *Advances in Neural Information Processing Systems*, 23, 2010.

Girin, L., Leglaive, S., Bie, X., Diard, J., Hueber, T., and Alameda-Pineda, X. Dynamical variational autoencoders: A comprehensive review. *arXiv preprint arXiv:2008.12595*, 2020.

Golub, G. H. and Van Loan, C. F. *Matrix computations*. JHU press, 2013.

Google Trends. Google Trends. https://trends.google.com/trends/, Accessed 11 November 2022.

Goris, R. L., Movshon, J. A., and Simoncelli, E. P. Partitioning neuronal variability. *Nature Neuroscience*, 17(6): 858–865, 2014.

Gower, J. Dijksterhuis, gb: Procrustes problems, 2004.

Handa, P., Mathur, M., and Goel, N. Open and free eeg datasets for epilepsy diagnosis. *arXiv preprint arXiv:2108.01030*, 2021.

Harshman, R. A. oundations of the parafac procedure: Models and conditions for an "explanatory" multimodal factor analysis. *UCLA Working Papers in Phonetics*, 16(1):1–84, 1970.

Haxby, J. V., Guntupalli, J. S., Connolly, A. C., Halchenko, Y. O., Conroy, B. R., Gobbini, M. I., Hanke, M., and Ramadge, P. J. A common, high-dimensional model of the representational space in human ventral temporal cortex. *Neuron*, 72(2):404–416, 2011.

He, H., Tan, Y., and Zhang, W. A wavelet tensor fuzzy clustering scheme for multi-sensor human activity recognition. *Engineering Applications of Artificial Intelligence*, 70:109–122, 2018. ISSN 0952-1976. doi: https://doi.org/10.1016/j.engappai.2018.01.004. URL https://www.sciencedirect.com/science/article/pii/S0952197618300046.

Hotelling, H. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417, 1933.

Hyvarinen, A., Karhunen, J., and Oja, E. Independent component analysis and blind source separation, 2001.

Jerzak, C. T., King, G., and Strezhnev, A. An improved method of automated nonparametric content analysis for social science. *Political Analysis*, 31(1):42–58, 2023.

Kala, R., Shulkla, A., and Tiwari, R. Fuzzy neuro systems for machine learning for large data sets. In *2009 IEEE International Advance Computing Conference*, pp. 541–545. IEEE, 2009.

Kobak, D., Brendel, W., Constantinidis, C., Feierstein, C. E., Kepecs, A., Mainen, Z. F., Qi, X.-L., Romo, R., Uchida, N., and Machens, C. K. Demixed principal component analysis of neural population data. *elife*, 5:e10989, 2016.

Kogbetliantz, E. Solution of linear equations by diagonalization of coefficients matrix. *Quarterly of Applied Mathematics*, 13(2):123–132, 1955.

Kossaifi, J., Panagakis, Y., Anandkumar, A., Atanasijevic, M., Balntas, N., Bursuc, A., Chen, S., Cohen, T., Contal, E., Couellan, B., Housni, K. E., Krol, D., Kusmenko, E., Marteau, D., Mocanu, A., Perrot, M., Prouvost, A., Remme, R., Schock, J., and Varikooty, K. R. TensorLy: Tensor learning in python. http://tensorly.org/stable/, 2021.

Lee, D. D. and Seung, H. S. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755): 788–791, 1999.

Liu, R., Azabou, M., Dabagia, M., Xiao, J., and Dyer, E. Seeing the forest and the tree: Building representations of both individual and collective dynamics with transformers. *Advances in Neural Information Processing Systems*, 35:2377–2391, 2022.

Mante, V., Sussillo, D., Shenoy, K. V., and Newsome, W. T. Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature*, 503(7474):78–84, 2013.

Mishne, G. and Charles, A. S. Learning spatially-correlated temporal dictionaries for calcium imaging. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1065–1069. IEEE, 2019.

Mishne, G., Talmon, R., Meir, R., Schiller, J., Lavzin, M., Dubin, U., and Coifman, R. R. Hierarchical coupled-geometry analysis for neuronal structure and activity pattern discovery. *IEEE Journal of Selected Topics in Signal Processing*, 10(7):1238–1253, 2016. doi: 10.1109/JSTSP.2016.2602061.

Mudrik, N., Chen, Y., Yezerets, E., Rozell, C. J., and Charles, A. S. Decomposed linear dynamical systems (dlds) for learning the latent components of neural dynamics. *Journal of Machine Learning Research*, 25(59): 1–44, 2024.

Nasreddine, W. Epileptic eeg dataset, 2021.

Olshausen, B. A. and Field, D. J. Wavelet-like receptive fields emerge from a network that learns sparse codes for natural images. *Nature*, 381:607–609, 1996.

Olshausen, B. A. and Field, D. J. Sparse coding of sensory inputs. *Current Opinion in Neurobiology*, 14(4):481–487, 2004.

pandas development team, T. pandas-dev/pandas: Pandas, February 2020. URL https://doi.org/10.5281/zenodo.3509134.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A.,

Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. URL http://jmlr.org/papers/v12/pedregosa11a.html.

Pei, F., Ye, J., Zoltowski, D. M., Wu, A., Chowdhury, R. H., Sohn, H., O'Doherty, J. E., Shenoy, K. V., Kaufman, M. T., Churchland, M., Jazayeri, M., Miller, L. E., Pillow, J., Park, I. M., Dyer, E. L., and Pandarinath, C. Neural latents benchmark '21: Evaluating latent variable models of neural population activity. In *Advances in Neural Information Processing Systems (NeurIPS), Track on Datasets and Benchmarks*, 2021. URL https://arxiv.org/abs/2109.04463.

Qin, J., Li, S., Needell, D., Ma, A., Grotheer, R., Huang, C., and Durgin, N. Stochastic greedy algorithms for multiple measurement vectors. *arXiv preprint arXiv:1711.01521*, 2017.

Ravasi, M. and Vasconcelos, I. Pylops–a linear-operator python library for scalable algebra and optimization. *SoftwareX*, 11:100495, 2020. doi: https://doi.org/10.1016/j.softx.2020.100495. URL https://www.sciencedirect.com/science/article/pii/S2352711020302449.

Rodriguez, A. C. Monkey (arm movement), 2023. URL https://doi.org/10.5281/zenodo.4662738.

Schmid, P. J. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656: 5–28, 2010.

Schneider, S., Lee, J. H., and Mathis, M. W. Learnable latent embeddings for joint behavioural and neural analysis. *Nature*, pp. 1–9, 2023.

Studer, C. and Baraniuk, R. G. Dictionary learning from sparsely corrupted or compressed signals. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3341–3344. IEEE, 2012.

Tanvir Ahmed, K., Cheng, S., Li, Q., Yong, J., and Zhang, W. Incomplete time-series gene expression in integrative study for islet autoimmunity prediction. *Briefings in Bioinformatics*, 24(1):bbac537, 2023.

Tillinghast, C. and Zhe, S. Nonparametric decomposition of sparse tensors. In *International Conference on Machine Learning*, pp. 10301–10311. PMLR, 2021.

Tillinghast, C., Fang, S., Zhang, K., and Zhe, S. Probabilistic neural-kernel tensor decomposition. In *2020 IEEE International Conference on Data Mining (ICDM)*, pp. 531–540. IEEE, 2020.

Tošić, I. and Frossard, P. Dictionary learning. *IEEE Signal Processing Magazine*, 28(2):27–38, 2011.

van den Berg, E. and Friedlander, M. P. Probing the pareto frontier for basis pursuit solutions. *SIAM Journal on Scientific Computing*, 31(2):890–912, 2008.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., and Contributors, S. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

Wang, Z. and Zhe, S. NONFAT: Nonparametric Factor Trajectory Learning for Dynamic Tensor Decomposition. https://github.com/wzhut/NONFAT, 2022a.

Wang, Z. and Zhe, S. Nonparametric factor trajectory learning for dynamic tensor decomposition. In *International Conference on Machine Learning*, pp. 23459–23469. PMLR, 2022b.

Wei, H., Chen, L., Ruan, K., Li, L., and Chen, L. Low-rank tensor regularized fuzzy clustering for multiview data. *IEEE Transactions on Fuzzy Systems*, 28(12):3087–3099, 2020. doi: 10.1109/TFUZZ.2020.2988841.

Wes McKinney. Data Structures for Statistical Computing in Python. In Stéfan van der Walt and Jarrod Millman (eds.), *Proceedings of the 9th Python in Science Conference*, pp. 56 – 61, 2010. doi: 10.25080/Majora-92bf1922-00a.

Williams, A. H., Kim, T. H., Wang, F., Vyas, S., Ryu, S. I., Shenoy, K. V., Schnitzer, M., Kolda, T. G., and Ganguli, S. Unsupervised discovery of demixed, Low-Dimensional neural dynamics across multiple timescales through tensor component analysis. *Neuron*, 98(6):1099–1115.e8, June 2018.

Wu, X., Shi, B., Dong, Y., Huang, C., and Chawla, N. Neural tensor factorization. *arXiv preprint arXiv:1802.04416*, 2018.

Xu, J. and Durrett, G. Spherical latent spaces for stable variational autoencoders. *arXiv preprint arXiv:1808.10805*, 2018.

Xu, Z., Yan, F., et al. Infinite tucker decomposition: Nonparametric bayesian models for multiway data analysis. *arXiv preprint arXiv:1108.6296*, 2011.

Yang, M.-S. A survey of fuzzy clustering. *Mathematical and Computer Modelling*, 18(11):1–16, 1993.

Zhang, Z. and Rao, B. D. Sparse signal recovery with temporally correlated source vectors using sparse bayesian learning. *IEEE Journal of Selected Topics in Signal Processing*, 5(5):912–926, 2011.

Zhe, S., Zhang, K., Wang, P., Lee, K.-c., Xu, Z., Qi, Y., and Ghahramani, Z. Distributed flexible nonlinear tensor factorization. *Advances in neural information processing systems*, 29, 2016.

Zou, H., Hastie, T., and Tibshirani, R. Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265–286, 2006.

Appendix

A. Notations

Throughout the paper, we adopt the following notations: the superscript $()^d$ refers to state d, and the subscript $()_m$ refers to trial m. Specifically, $\boldsymbol{Y}_m^d$ and Φ_m^d denote the observations and temporal traces of trial m of state d, while $\boldsymbol{A}^d$ represents the BBs of state d. Additionally, for a general matrix $\boldsymbol{Z}$, $\boldsymbol{Z}_{i:}$ ($\boldsymbol{Z}_{:j}$) denotes its i-th row (j-th column).

Table 1. Notations used in the paper.

Symbol	Description
BBs	Building Blocks
channels	Each feature in the observations, e.g., neurons in recordings
states	Different "views" of the observations. e.g., different cognitive tasks
trials/sessions	Repeated observations within state
p	Number of BBs
D	Number of states
M_d	Number of trials for state d
N	Number of channels
$\boldsymbol{Z}_{n:}$ (or $\boldsymbol{Z}_{[n,:]}$)	The n-th row of a general matrix $\boldsymbol{Z}$
$\boldsymbol{Z}_{:i}$ (or $\boldsymbol{Z}_{[:,i]}$)	The i-th column of a general matrix $\boldsymbol{Z}$
L_d	Label of state d (optional, can be a scalar or a vector)
$\boldsymbol{Y}_m^d \in \mathbb{R}^{N \times T_m^d}$	Observation for trial m and state d
$\boldsymbol{A}^d \in \mathbb{R}^{N \times p}$	Matrix of BBs for state d.
$\Phi_m^d \in \mathbb{R}^{T_m^d \times p}$	Matrix of temporal traces for trial m of state d.
$\boldsymbol{P} \in \mathbb{R}^{D \times D}$	States similarity graph
$\{\boldsymbol{H}^d\}_{d=1}^D$, s.t. $\boldsymbol{H}^d \in \mathbb{R}^{N \times N}$	Channel similarity graphs
$\boldsymbol{\nu} \in \mathbb{R}^p$	Controls the relative level of cross-state similarity for each BB
$\boldsymbol{V} = diag(\boldsymbol{\nu})$	A diagonal matrix whose entry in index ii is the i-th entry of $\boldsymbol{\nu}$
ϵ, β, w_{graph}	Hyperparameters controlling the strength of regularization
$\gamma_1, \gamma_2, \gamma_3, \gamma_4$	Hyperparameters to regularize $\boldsymbol{\Phi}_m^d$
$\sigma_{\boldsymbol{H}}, \sigma_{\boldsymbol{P}}$	Hyperparameters that control the bandwidth of the kernel
$\boldsymbol{\psi}_n^{ij} \in \mathbb{R}^{M_j, M_i}$	Transformation of the data from state i to state j for channel n

B. Further options for $\boldsymbol{P}$ computation

Here, we explore additional approaches for computing the state-similarity graph $\boldsymbol{P}$. These options take into account factors like data properties, single vs. multi-trial cases, variations in trial duration, and the desired approach (supervised or data-driven).

B.1. Supervised $\boldsymbol{P}$

In addition to the case presented in the paper, for sequential/ordered states, here we introduce the supervised version designed for categorical or similar-distanced states.

B.1.1. Categorical or Similar-Distanced States

For cases where observation states are represented by categorical labels, and we expect a high degree of similarity between all possible pairs of states (i.e., no pair of labels is closer to each other than to another pair), we can define the state similarity matrix $\boldsymbol{P}$ to ensure uniform values for all entries of distinct states, with larger values assigned to same-state entries located on the matrix diagonal. $\boldsymbol{P}$ is then constructed as

$$\boldsymbol{P} = \mathbf{1} \otimes \mathbf{1}^T + c\boldsymbol{I}, \tag{4}$$

where $\mathbf{1} \otimes \mathbf{1}^T \in \mathbb{R}^{D \times D}$ is a matrix of all ones, $\boldsymbol{I} \in \mathbb{R}^{D \times D}$ is the identity matrix, and c is a weight that scales the strength of same-state similarity with respect to cross-state similarities.

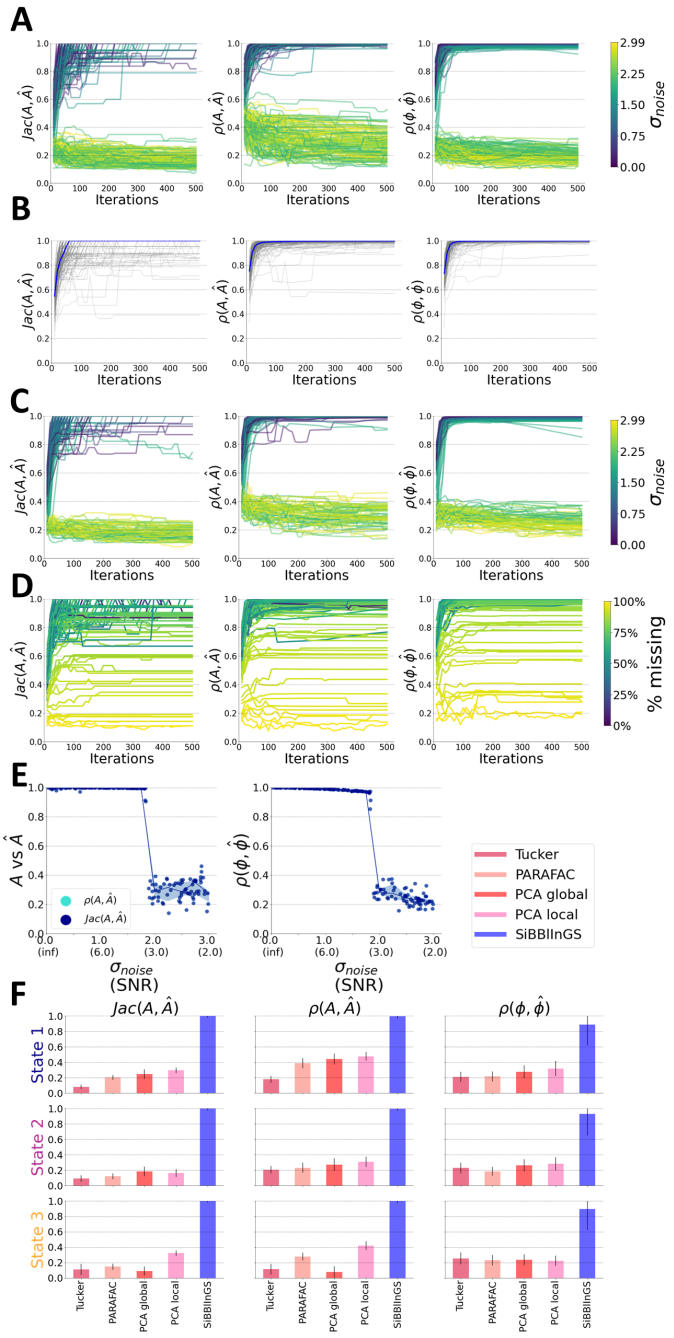

Figure 6. **Synthetic Data Results Robustness - cont.** **A** Model performance under increasing levels of noise, along with random initializations, over the model training iterations. Color: increasing levels of missing samples. Left: Jaccard index between the recovered $\boldsymbol{A}$ and the ground true $\boldsymbol{A}$. Middle: Correlation between the recovered $\boldsymbol{A}$ and the ground truth $\boldsymbol{A}$. Right: Correlation between the recovered $\boldsymbol{\Phi}$ and the ground true $\boldsymbol{\Phi}$. **B** Model performance under random initializations (no noise), over the model training iterations. The blue curve is the median over all repeats. **C** Model performance under increasing levels of noise only (fixed initializations). **D** Model performance under increasing levels of missing samples, over the model training iterations. **E** SiBBlInGSPerformance under increasing noise levels. **F** Comparison to other relevant methods, for each state individually (SiBBlInGSin blue, other methods in pink to red colors).

Table 2. Assumptions and capabilities comparison between SiBBlInGS and other methods.

Method	SiBBlInGS	mTDR	PCA	Fast ICA	NMF	GPFA	SRM	HOSVD	PARAFAC
Do not force orthogonality?	V	X	X	V	V	V	X	X	V
Sparse?	V	X	X	X	X	X	X	X	X
Flexible in time across states?	V	X	X	X	X	X	X	V	V
Support variations in BB across states?	V	X	X	X	X	X	X	X	X
Used for condition variability?	V	V	X	X	X	X	X	V	V
Works on tensors?	V	V	X	X	X	X	V	V	V
Consider both within & between states variability?	V	V	na	na	na	na	X	X	X
Supports state-specific emerging components?	V	V	na	na	na	na	V	X	X
Works on non-consistent data duration or sampling rates?	V	X	na	na	na	na	X	X	X
Can prior knowledge (labels) control state similarity?	V	V	na	na	na	na	V	X	X
Ability to define both state-specific and background components?	V	V	na	na	na	na	X	X	X
Supports non-negative decomposition?	V	X	X	X	V	X	X	X	V

B.2. Data-Driven $\boldsymbol{P}$

When prior knowledge about state similarity is uncertain or unavailable, SiBBlInGS also provides an unsupervised, data-driven approach to calculate $\boldsymbol{P}$ based on the distance between data points across states. Here we discuss the four options for constructing the matrix $\boldsymbol{P}$ in a data-driven manner, depending on the structure of the observations.

B.2.1. Single-trial per-state with equal-length across states:

This case refers to the scenario of a single trial for each state ($M_d = 1, \quad \forall d = 1 \dots D$), where all cross-state trials have the same length ($T_1^d = T \quad \forall d = 1 \dots D$). Here, the similarity graph $\boldsymbol{P}$ is constructed as

$$\boldsymbol{P}_{d,d'} = \exp\left(-||\boldsymbol{Y}_1^d - \boldsymbol{Y}_1^{d'}||_F^2/\sigma_{\boldsymbol{P}}^2\right), \tag{5}$$

where $\sigma_{\boldsymbol{P}}$ controls the bandwidth of the kernel.

B.2.2. Multiple trials per state, same trial duration

In the case where all trials have the same temporal duration, the similarity matrix $\boldsymbol{P}$ is computed by evaluating the distance between the values of each pair of states, considering all trials within each state. For this, we first find the transformation $\boldsymbol{\psi}_n^{ij} \in \mathbb{R}^{M_j \times M_i}$ between the observations of state i to the observation of state j, by solving the Orthogonal Procrustes problem (Golub & Van Loan, 2013; Gower, 2004). For this, let $\boldsymbol{Y}^{i*} \in \mathbb{R}^{M_i \times (T \times N)}$ be the matrix obtained by vertically concatenating the flattened observations from each trial ($m = 1 \dots M_i$) of state i. Then, the optimal transformation from the observations of state i ($\boldsymbol{Y}^{i*} \in \mathbb{R}^{M_i \times (T \times N)}$) to the observations of state j ($\boldsymbol{Y}^{j*} \in \mathbb{R}^{M_j \times (T \times N)}$) will be

$$\widehat{\boldsymbol{\psi}}^{ij} = \underset{\boldsymbol{\psi}^{ij}}{\arg\min} \|\boldsymbol{\psi}^{ij}\boldsymbol{Y}^{i*} - \boldsymbol{Y}^{j*}\|_F^2, \tag{6}$$

where this mapping projects the multiple trials of state i into the same space as of state j, via $\widetilde{\boldsymbol{Y}}^{i*} = \widehat{\boldsymbol{\psi}}^{ij}\boldsymbol{Y}^{i*}$. The state similarity matrix will thus be

$$\boldsymbol{P}_{ij} = \exp\left(-\|\widetilde{\boldsymbol{Y}}^{i*} - \boldsymbol{Y}^{j*}\|_F^2/{\sigma_p}^2\right), \tag{7}$$

for all states $i \neq j = 1 \dots D$, where σ_p controls the kernel bandwidth.

B.2.3. SINGLE-TRIAL PER STATE, SAME DURATION

Further generalization of the state similarity computation requires addressing the case of trials with varying duration. When the observations correspond to the same process and their alignment using dynamic time warping is justifiable, we can replace the Gaussian kernel measure with the Dynamic Time Warping (DTW) distance metric (Berndt & Clifford, 1994). In the case of a single trial for each state, the similarity metric becomes the average DTW distances over all channels,

$$\boldsymbol{P}_{ij} = \exp\left(-\frac{1}{N}\sum_{n=1}^{N} DTW(\boldsymbol{Y}^i_{n:}, \boldsymbol{Y}^j_{n:})\right). \tag{8}$$

B.2.4. MULTIPLE TRIALS PER STATE, DIFFERENT DURATION

Similarly, for the multi-trial case we have

$$\boldsymbol{P}_{ij} = \exp\left(-\frac{1}{N}\sum_{n=1}^{N}\left(\frac{1}{M_i}\frac{1}{M_j}\sum_{m_j=1}^{M_j}\sum_{m_i=1}^{M_i}\text{DTW}\Big(\Big(\boldsymbol{Y}^j_{m_j}\Big)_{n:} - \big(\boldsymbol{Y}^i_{m_i}\big)_{n:}\Big)\right)\right), \tag{9}$$

where, DTW, is, as before, the Dynamic Time Wrapping (Berndt & Clifford, 1994) operator, applied on the activity of the n-th channel in both states. It is crucial to note that this approach operates under the assumption that the trials being compared depict similar processes, and hence aligning them using DTW is a valid assumption.

C. Channel-similarity kernel ($\boldsymbol{H}$)—generation and processing

The kernel post-processing involves several steps. First, we construct the kernel $\widetilde{\boldsymbol{H}}^d$ for each state $d = 1 \dots D$, as described in the main text. To incorporate similarities between each possible pair of states $d' \neq d$, where $d, d' = 1 \dots D$, we perform a weighted average of each $\widetilde{\boldsymbol{H}}^d$ with the kernels of all other states, using $\boldsymbol{P}_{d:}$ for the weights, as it quantifies the similarity between state d and all other states: $\boldsymbol{H}^d = \sum_{d'=1}^{D} \boldsymbol{P}_{dd'}\widetilde{\boldsymbol{H}}^{d'}$. Then, to promote a more robust algorithm, we only retain the k highest values (i.e., k-Nearest Neighbors; kNN) in each row, while the rest are set to zero. The value of k is a model hyperparameter, and depends on the desired BB size. We then symmetrize each state's kernel by calculating $\boldsymbol{H}^d \leftarrow \frac{1}{2}\left(\boldsymbol{H}^d + (\boldsymbol{H}^d)^T\right)$ for all $d = 1 \dots D$. Finally, the kernel is row-normalized so that each row sums to one, as follows: Let Λ^d be a diagonal matrix with elements representing the row sums of $\boldsymbol{H}^d$, i.e., $\boldsymbol{\Lambda}^d_{ii} = \sum_{n=1}^{N} \boldsymbol{H}^d_{i,n}$. The final normalized channel similarity kernel is obtained as $\boldsymbol{H}^d_{\text{final}} = (\boldsymbol{\Lambda}^d)^{-1}\boldsymbol{H}^d$.

D. Solving Φ in practice

In Section 4, the model updates the temporal traces dictionary $\boldsymbol{\phi} = \boldsymbol{\Phi}^d_m$ for all $m = 1 \dots M_d$, $d = 1 \dots D$ using an extended least squares for each time point t, i.e.,

$$\widetilde{\boldsymbol{\phi}}_{[t,:]} = \arg\min_{\boldsymbol{\phi}_{[t,:]}} \|\widetilde{\boldsymbol{Y}}^d_{m_{[:,t]}} - \widetilde{\boldsymbol{M}}\boldsymbol{\phi}_{[t,:]}\|_2^2, \tag{10}$$

where $\boldsymbol{\phi}_{[t,:]} \in \mathbb{R}^p$ is the dictionary at time t,

$$\widetilde{\boldsymbol{Y}}^d_{m_{[:,t]}} = \begin{bmatrix} \boldsymbol{Y}^d_{m_{[:,t]}} \\ [\boldsymbol{0}]_{p\times 1} \\ \gamma_2\boldsymbol{\phi}^{(iter-1)}_{[t,:]} + \gamma_3\boldsymbol{\phi}_{[(t-1),:]} \end{bmatrix}, \quad \text{and} \quad \widetilde{\boldsymbol{M}} = \begin{bmatrix} \boldsymbol{A}^d \\ \gamma_4(\frac{[1]_{p\times p}}{p} - \boldsymbol{I}_{p\times p}) \circ \sqrt{D} \\ (\gamma_1+\gamma_2+\gamma_3)\boldsymbol{I}_{p\times p} \end{bmatrix},$$

with all parameters being the same as those defined in Section 4 of the main text and $\circ$ denotes element-wise multiplication. Here, $[0]_{p\times 1} \in \mathbb{R}^{p\times 1}$ represents a column vector of zeros, $[1]_{p\times p}$ represents a square matrix of ones with dimensions $p \times p$, and $\boldsymbol{Y}^d_{m_{[:,t]}} \in \mathbb{R}^N$ denotes the measurement in the m-th trial of state d at time t.

E. Model Complexity

SiBBIInGS relies on 4 main computational steps:

Channel Graph Construction: This operation, performed once for all N channels of every state $d = 1 \dots D$, generates a channel graph $\boldsymbol{H}^d \in \mathbb{R}^{N \times N}$ for each state $d \in [1, D]$ by concatenating within-state trials $1 \dots M_d$ horizontally, resulting in a $N \times \sum_{m=1}^{M_d} T_m^d$ matrix. For simplicity, let $\widetilde{T} = \sum_{m=1}^{M_d} T_m^d$. The computational complexity of calculating the pairwise similarities of this concatenated matrix for all D states is thus $\mathcal{O}\left(D\widetilde{T}^2 N(N-1)\right)$.

For the k-threshold step (B.2.1), that involves keeping only the k largest values in each row while setting the other values to zero—the complexity will be $\mathcal{O}\left(\widetilde{T}\log k\right)$ per row for a total computational complexity of $\mathcal{O}\left(DN\widetilde{T}\log k\right)$ for N rows and D states.

State Graph Construction: This is a one-time operation that involves calculating the pairwise similarities between each pair of states. For simplicity, if we assume the case of user-defined scalar labels, and as in this case there are D states (and accordingly D labels), the computation includes $D(D-1))/2$ pairwise distances for $\mathcal{O}\left(D^2\right)$.

BB Inference (Eq. (1)): This iterative step involves per-channel re-weighted ℓ_1 optimization. If the computational complexity of a weighted ℓ_1 is denoted as $\mathcal{C}$, then the computational complexity of the re-Weighted ℓ_1 Graph Filtering is $NL\mathcal{C} + LNk$, where N is the number of channels, L is the number of iterations for the RWLF procedure, and k is the number of nearest neighbors in the graph. For the last term in Eq. (1), there are p^2 multiplicative operations involving the vector ν and the difference in BBs, arising from the ℓ_2^2 norm. Additionally, there is an additional multiplication step involving $\boldsymbol{P}_{dd'}$. For each state d, this calculation repeats itself $D-1$ times (for all $d' \neq d$). This process is carried out for every $d = 1 \dots D$. In total, these multiplicative operations sum up to $\left(p^2+1\right) D(D-1)$, resulting in a computational complexity of $\mathcal{O}\left(D^2p^2\right)$.

Optimization for $\boldsymbol{\phi}$: This step refers to the least-squares problem presented in Eq. (10) in Appendix D. If a non-negative constraint is applied, SiBBlInGS uses scipy's "nnls" for solving $\widetilde{\boldsymbol{\phi}}_{[t,:]} = \arg\min_{\boldsymbol{\phi}_{[t,:]}} \|\widetilde{\boldsymbol{Y}}^d_{m_{[:,t]}} - \widetilde{\boldsymbol{M}}\boldsymbol{\phi}_{[t,:]}\|_2^2$, where $\widetilde{\boldsymbol{Y}}_m^d \in R^{(N+2p)\times T_m^d}$ and $\boldsymbol{M} \in R^{(N+2p)\times p}$. This results in complexity of $\mathcal{O}\left(p(N+2p)^2 F \max\left(T_m^d\right)\right)$, where F is the number of nnls iterations. Without non-negativity constraint, this problem is a least squares problem with a complexity of $\mathcal{O}\left(\max\left(T_m^d\right)p^2(2p+N)\right)\right)$. Potential complexity reduction options include parallelizing RWL1 optimizations per channel, using efficient kNN or approximate kNN search for constructing kNN graphs instead of full graphs, and employing dimensionality reduction techniques to expedite nearest neighbor searches.

F. Data and Code Availability

The code employed in this study will be made publicly available on GitHub upon publication and is included in the supplementary material of the submission. The data used in this study are publicly available and cited within the paper.

Algorithm 1 The SiBBlInGS Model (short version)

Inputs
$\{\boldsymbol{Y}_m^1\}_{m=1}^{M_1}, \dots \{\boldsymbol{Y}_m^D\}_{m=1}^{M_D}$ {Observations}
$\beta, \epsilon, \gamma_1, \gamma_2, \gamma_3, \gamma_4, w_{graph}, \sigma_p, \sigma_H$ {Hyperparameters}
Initialization and pre-Calculations
$\{\boldsymbol{A}^d\}, \{\boldsymbol{\Phi}_m^d\}_{m=1}^{M_d}, \forall d = 1 \dots D$ {Initialize BBs&traces}
$\boldsymbol{P} \in \mathbb{R}^{D\times D}$ {Calculate state-similarity graph}
$\boldsymbol{H}^d \in \mathbb{R}^{N\times N}, \forall d = 1 \dots D$ {Calculate channels graphs}
repeat
  for $d = 1 \dots D$ **do**
    Select a random batch of trials from state d
    Update $\boldsymbol{A}^d$ and $\boldsymbol{\lambda}^d$ {via Eq. (1) and Eq. (2)}
    for $m = 1 \dots M_d$ **do**
      Update $\boldsymbol{\Phi}_m^d$ {via Eq. (3)}
    end for
  end for
until BBs and traces of all states converged

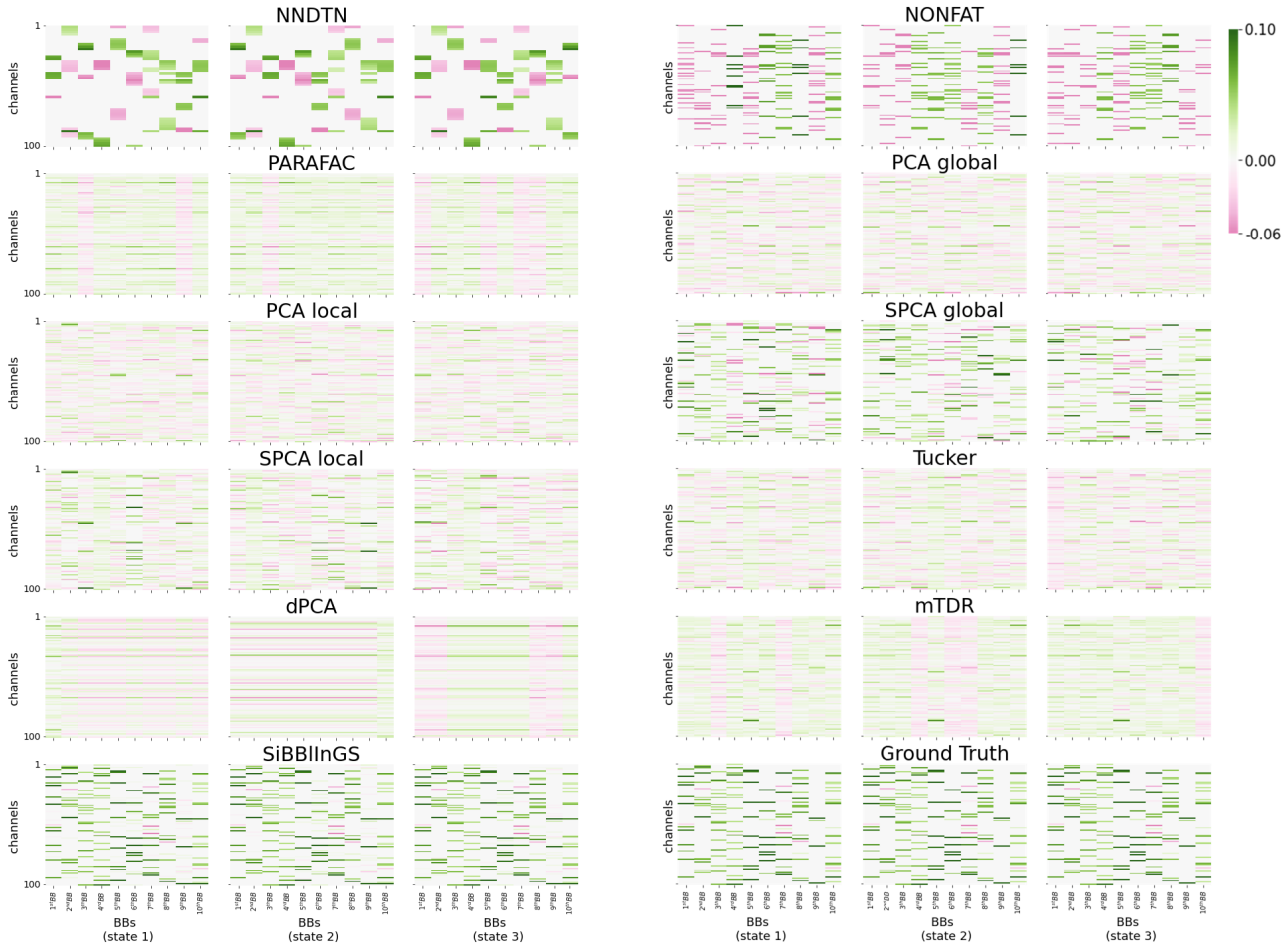

Figure 7. **BBs identified by different methods**. BBs identified by SiBBlInGS are compared with those from other methods, including PARAFAC, Tucker, PCA (global and local), Sparse PCA (global and local), demixed PCA (Kobak et al., 2016), mTDR (Aoi & Pillow, 2018), and Gaussian-process-based methods (Wang & Zhe, 2022b). The identified BBs were reordered to best match the ground truth BBs' temporal traces through maximum correlations. A subsequent hard-thresholding step was applied to achieve sparsity, aligning with the sparsity level with of the ground truth components. The BBs were normalized to have an absolute sum of 1 each for visualization purposes.

G. General Experimental Details

All experiments and code were developed and executed using Python version 3.10.4 and are compatible with standard desktop machines.

H. Synthetic Data—Additional Information

H.1. Synthetic Generation Details

We initiated the synthetic data generation process by setting the number of channels to $N = 100$ and the maximum number of BBs to $p = 10$. We further defined the number of states as $D = 3$ and determined the number of time points in each observation to be $T^d = 300$, where d represents the state index (here $d \in \{1, 2, 3\}$). We defined the number of trials for each state as one, i.e., $M_d = 1$ for $d = 1, 2, 3$.

We first initialized a "general" BB matrix ($\boldsymbol{A}$) as the initial structure, which will later undergo minor modifications for each state.

For each state d, we generated the time-traces $\boldsymbol{\Phi}^d$ via a linear combination of 15 trigonometric signals, such that the

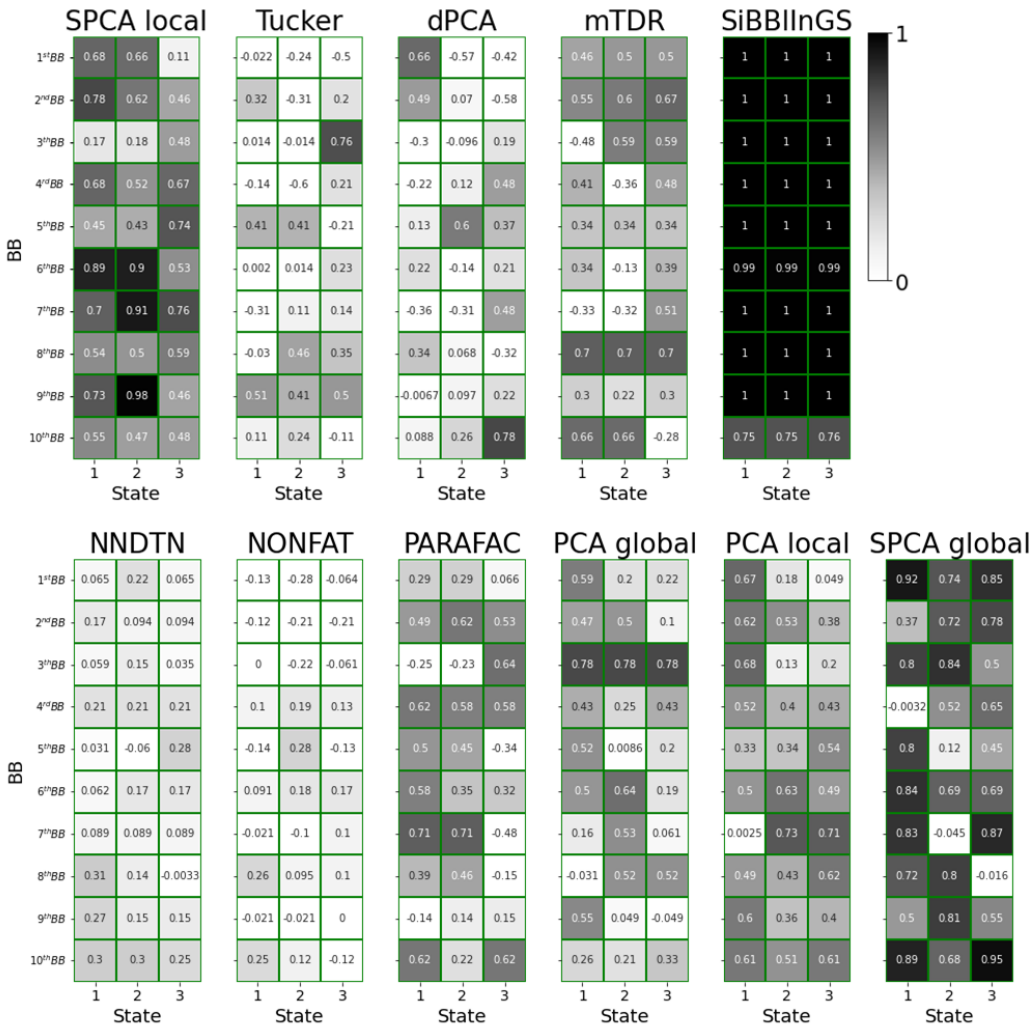

Figure 8. Correlations between BBs identified by different methods and ground truth BBs for each state and BB number.

temporal trace of the j-th BB was defined as $\mathbf{\Phi}^d_{:j} = \sum_{i=1}^{15} c_i f_i(\text{freq}_i * x)$ where x is an array of $T = 300$ time points ($x = [1, \ldots, 300]$), freq_i is a random scaling factor sampled uniformly on $[0, 5]$, f refers to a random choice between the sine and cosine functions (with probability 1/2 for each), and the sign (c_i) was flipped ($+1$ or -1) with a probability of 1/2.

During the data generation process, we incorporated checks and updates to $\boldsymbol{A}$ and $\mathbf{\Phi}$ to ensure that the BBs and their corresponding time traces are neither overly correlated nor orthogonal, are not a simple function of the states labels, and that different BBs exhibit comparable levels of contributions. This iterative process involving the checks persisted until no further modifications were required.

The first check aimed to ensure that the temporal traces of at least two BBs across all states were not strongly correlated with the state label vector ($[1, 2, 3]$) at each time point. Specifically, we examined whether the temporal traces of a j-th BB across all states ($\Phi^1_{tj}, \Phi^2_{tj}, \Phi^3_{tj}$) exhibited high correlation with the state label vector at each time point. This check was important to avoid an oversimplification of the problem by preventing the temporal traces from being solely influenced by the state labels. To perform this check, we calculated the average correlation between the temporal traces and the state labels ($[1, 2, 3]$) at each time point. If the average correlation over time exceeded a predetermined threshold of 0.6, we introduced additional variability in the time traces of the BB that exhibited a high correlation with the labels. This was achieved by adding five randomly generated trigonometric functions to the corresponding BB. These additional functions were generated in the same manner as the original data (with $\mathbf{\Phi}^d_{:j} = \sum_{i=1}^{5} c_i \boldsymbol{f}_i(\text{freq}_i \cdot x)$).

The second check ensured that the time traces were not highly correlated with each other and effectively represented separate functions. If the correlation coefficient between any pair of temporal traces of different BBs exceeded a threshold of $\rho = 0.6$, the correlated traces were perturbed by adding zero-mean Gaussian noise with a standard deviation of $\sigma = 0.02$.

Next, we ensured that the BBs represented distinct components by verifying that they were not highly correlated with each other. Specifically, if the correlation coefficient between a pair of BBs ($\boldsymbol{A}_{:j}$, $\boldsymbol{A}_{:i}$ for $j, i = 1 \dots 10$) within a state exceeded the threshold $\rho = 0.6$, each BB in the highly-correlated pair was randomly permuted to ensure their distinctiveness.

To prevent any hierarchical distinction or disparity in BB contributions and distinguish our approach from order-based methods like PCA or SVD, we evaluated each BB's contribution by measuring the increase in error when exclusively using that BB for reconstruction. Specifically, for the j-th BB, we calculated its contribution as $\text{contribution}_j = -\|\boldsymbol{Y} - \boldsymbol{A}_{:j} \otimes \boldsymbol{\Phi}_{:j}\|_F$, where $\otimes$ denotes the outer product. Then, we compared the contributions between every pair of BBs within the same state. If the contribution difference between any pair of BBs exceeded a predetermined threshold of 10, both BBs in the pair were perturbed with random normal noise. Subsequently, a hard-thresholding operation was applied to ensure that the desired cardinality was maintained.

To introduce slight variability also in the BBs' structures across states (in addition to the temporal variability), the general basis matrix $\boldsymbol{A}$ underwent modifications for each of the states. In each state and for each BB, a random selection of 0 to 2 non-zero elements from the corresponding BB in the original $\boldsymbol{A}$ matrix were set to zero, effectively introducing missing channels as differences between states, such that $\boldsymbol{A}^d$ is the updated $\boldsymbol{A}$ modified for state d. Finally, the data was reconstructed using $\boldsymbol{Y}^d = \boldsymbol{A}^d(\boldsymbol{\Phi}^d)^T$ for each state $d = 1, 2, 3$.

H.2. Experimental details to the Synthetic data

We applied SiBBLInGS to the synthetic data with $p = 10$ components and a maximum number of 10^3 iterations, while in practice about 50 iterations were enough to converge (see Fig. 6). The parameters for the λ update in Eq. (2) were $\epsilon = 0.01$, $\beta = 0.09$, and $w_{\text{graph}} = 1$. For the regularization of Φ (Eq. (3)), the parameters used were $\gamma_1 = 0.1$, $\gamma_2 = 0.1$, $\gamma_3 = 0$, and $\gamma_4 = 0.0001$. ν was set to be a vector of ones with length $p = 10$. The number of repeats for the update of $\boldsymbol{A}$ within an iteration, for each state, was set to 2. The number of neighbors used in the channel graph reconstruction ($\boldsymbol{H}^d$) was $k = 25$. The python scikit-learn's (Pedregosa et al., 2011) LASSO solver was used for updating $\boldsymbol{A}$ in each iteration. This synthetic demonstration used the supervised case for building $\boldsymbol{P}$, where $\boldsymbol{P}$ was defined assuming similar similarity levels between each pair of states, by defining $\boldsymbol{P} = \mathbf{1} \otimes \mathbf{1}^T \in \mathbb{R}^{3\times 3}$ (the case described in App. B.1.1, with $c = 1$).

H.3. Jaccard index calculation

In Figure 1C, we computed the Jaccard similarity index between the identified BBs by SiBBlInGS and the ground truth BBs. To obtain this measure, we first rearranged the BBs based on the correlation of their temporal traces with the ground truth traces (since the method is invariant to the order of the BBs). Then, we nullified the 15 lower percentiles of the $\widehat{\boldsymbol{A}}$ matrix, which correspond to values very close to zero. Finally, we compared the modified identified BBs to the ground truth BBs using the "jaccard_score" function from the sklearn library (Pedregosa et al., 2011).

I. Google Trends—Additional Information

I.1. Trends data acquisition and pre-processing

The acquisition and pre-processing of Google Trends data involved manually downloading the data from April 1, 2010, to November 27, 2022, for each of the selected states: California (CA), Maryland (MD), Michigan (MI), New York (NY), Illinois (IL), Louisiana (LA), Florida (FL), and Washington (WA), directly from the Google Trends platform (Google Trends, Accessed 11 November 2022). The full list of terms (including their clusters as SiBBlInGS recovered) is presented in Figure 9. To ensure coverage of search patterns, the data was downloaded by examining each query in all capitalization formats, including uppercase, lowercase, and mixed case.

The data (in CSV format) was processed using the "pandas" library in Python (pandas development team, 2020; Wes McKinney, 2010) and keeping only the relevant information from January 2011 to October 2022, inclusively. We conducted a verification to ensure the absence of NaN (null) values for each term in every selected state. To pre-process each term, we implemented a two-step normalization procedure. First, the values within the chosen date range were scaled to a maximum value of 100. This step ensured that the magnitude of each term's fluctuations remained within a consistent range. Next,

	CA	FL	IL	LA	MD	MI	NY	WA
BB 1	Berkeley, Campus, College, Harvard, Phd, Princeton	Admissions, Campus, College, Harvard, Phd, Princeton	Admissions, Campus, College, Harvard, Phd, Princeton	Admissions, Campus, College, Harvard, Phd, Princeton	Admissions, Campus, College, Harvard, Phd, Princeton	Admissions, Campus, College, Harvard, Phd, Princeton	Admissions, Berkeley, Campus, College, Harvard, Princeton	Admissions, Campus, College, Harvard, Phd, Princeton
BB 2	Afikomen, Chametz, Charoset, Haggadah, Pesach, Seder	Afikomen, Chametz, Charoset, Haggadah, Passover, Pesach	Chametz, Charoset, Haggadah, Passover, Pesach, Seder	Chametz, Charoset, Haggadah, Passover, Pesach, Seder	Afikomen, Chametz, Charoset, Haggadah, Passover, Pesach	Chametz, Charoset, Haggadah, Passover, Pesach, Seder	Afikomen, Chametz, Charoset, Haggadah, Pesach, Seder	Chametz, Charoset, Haggadah, Passover, Pesach, Seder
BB 3	Auld lang syne, Chicken soup, Decorations, Depression, Gpa, Sweets	Auld lang syne, Champagne, Chicken soup, Depression, Gpa, Sweets	Auld lang syne, Chicken soup, Decorations, Depression, Gpa, Sweets	Auld lang syne, Champagne, Chicken soup, Decorations, Gpa, Sweets	Auld lang syne, Champagne, Chicken soup, Decorations, Gpa, Sweets	Auld lang syne, Champagne, Chicken soup, Decorations, Gpa, Sweets	Auld lang syne, Champagne, Chicken soup, Countdown, Decorations, Sweets	Auld lang syne, Champagne, Chicken soup, Decorations, Depression, Sweets
BB 4	Elf, Gift, New years eve, Poinsettia, Ugly sweater	Elf, Gift, New years eve, Poinsettia, Ugly sweater	Elf, Gift, New years eve, Poinsettia, Ugly sweater	Elf, Gift, New years eve, Poinsettia, Ugly sweater	Elf, Gift, New years eve, Poinsettia, Ugly sweater	Elf, Gift, New years eve, Poinsettia, Ugly sweater	Elf, Gift, New years eve, Poinsettia, Ugly sweater	Elf, Gift, New years eve, Poinsettia, Ugly sweater
BB 5	Cdc, Hopkins, Kippur, N95, Quarantine, Zoom	Cdc, Hopkins, Mit, N95, Quarantine, Zoom	Cdc, Hopkins, N95, Quarantine, Zoom	Cdc, Hopkins, Mit, N95, Quarantine, Zoom	Cdc, Hopkins, N95, Quarantine, Zoom	Cdc, Hopkins, N95, Quarantine, Zoom	Cdc, Hopkins, N95, Quarantine, Zoom	Cdc, Hopkins, Mit, N95, Quarantine, Zoom

Figure 9. **Table of clustered words for the Google Trends experiment**

	SiBBlInGS	PCA Local	PCA Global	PARAFAC	Tucker
BB 1	Cdc, Hopkins, N95, Quarantine, Zoom	Hopkins, N95, Quarantine, Zoom	Hopkins, N95, New years eve, Quarantine, Ugly sweater, Zoom		
BB 2	Afikomen, Chametz, Charoset, Haggadah, Pesach, Seder	Afikomen, Ball drop, Charoset, Elf, Gift, Haggadah, Memorial N95, Pesach, Ugly sweater, Zoom	Afikomen, Ball drop, Berkeley, Chametz, Depression, Gpa, Haggadah, Harvard, Memorial, N95, New years eve, Passover, Pesach, Seder, Spirit	Admissions, Afikomen, Berkeley, Cdc, Chametz, Charoset, Decorations, Depression, Gpa, Haggadah, Harvard, Instacart, Labor, Matzo ball, Passover, Pesach, Princeton, Seder, Spirit	Elf, Hopkins, N95, New years eve, Poinsettia, Quarantine, Santa Ugly sweater, Zoom
BB 3	Auld lang syne, Chicken soup, Decorations, Depression, Gpa, Sweets	Charoset, Elf, Memorial, New years eve, Ugly sweater	Cdc, Chametz, Charoset, Haggadah, N95, Passover, Pesach, Quarantine, Seder, Zoom	Admissions, Ball drop, Cdc, Countdown, Hopkins, Instacart, N95, New years eve, Quarantine, Zoom	Cdc, Chametz, Charoset, Haggadah, N95, Passover, Pesach, Quarantine, Seder, Zoom
BB 4	Elf, Gift, New years eve, Poinsettia, Ugly sweater	Cdc, N95, Quarantine, Zoom	Afikomen, Auld lang syne, Ball drop, N95, New years eve, Ugly sweater	Auld lang syne, Champagne, Decorations, Elf, Gift, Hopkins, Labor, Memorial, N95, New years eve, Poinsettia, Santa, Ugly sweater	Campus, Charoset, Elf, Labor, Memorial, New years eve, Spirit, Ugly sweater
BB 5	Berkeley, Campus, College, Harvard, Phd, Princeton	Afikomen, Charoset, Memorial, Zoom	Charoset, Elf, Labor, New years eve, Ugly sweater	Auld lang syne, Ball drop, N95, New years eve	Auld lang syne, Ball drop, Labor, N95, Memorial, New years eve, Ugly sweater

Figure 10. **Comparison of The Google Trends Results to Other Methods with 5 BBs for CA:** Comparison to other methods, each applied with $p = 5$ BBs, yielded less interpretable BBs. For example, SiBBlInGS discerns theme-specific BBs (e.g., 'Covid' and 'University'), while other methods produce more blended compositions. Empty cells for PARAFAC and Tucker indicate that those BBs remained empty.

the values for each term were divided by the sum of values across all dates and then multiplied by 100, resulting in an adjusted scale where the area under the curve for each term equaled 100. This normalization procedure accounted for potential variations in the frequency and magnitude of term occurrences, enabling fair comparisons across different terms. By applying these pre-processing steps, we aimed to mitigate the influence of isolated spikes or localized peaks that could distort the overall patterns and trends observed in the data. Since the focus of this processing was on assessing the relative contribution of a term within a BB rather than comparing the overall amplitude and mean of the term across states, factors such as population size and other characteristics of each state were not taken into consideration.

I.2. Experimental details for Google trends

We ran the Trends experiment with $p = 5$ BBs, and applied non-negativity constraints to both the BB components and their temporal traces. The $\boldsymbol{\lambda}$'s parameters in Equation (1) included $\epsilon = 9.2$, $\beta = 0.01$, and $w_{\text{graph}} = 35$. For the regularization of $\boldsymbol{\Phi}$ in Equation (3), we used the parameters $\gamma_1 = 0$, $\gamma_2 = 0$, $\gamma_3 = 0.05$, $\gamma_4 = 0.55$. The trends example used the data-driven version for studying $\boldsymbol{P}$, and we set $\boldsymbol{\nu}$ to be a vector of ones with length $p = 5$.

During each iteration, $\boldsymbol{A}$ underwent two updates within each state d. The number of neighbors we used in the channel graph reconstruction was $k = 4$. We used the PyLops package in Python, along with the SPGL1 solver (Ravasi & Vasconcelos, 2020) to update A in each iteration. With respect to SPGL1 parameters (as described in (Ravasi & Vasconcelos, 2020)), we set the initial value of the parameter τ to 0.12, and a multiplicative decay factor of 0.999 was applied to it at each iteration. We note here that SPGL1 solves a Lagrangian variation of the original Lasso problem, where, i.e., it bounds the ℓ_1 norm of the selected BB to be smaller than τ, rather than adding the ℓ_1 regularization to the cost (van den Berg & Friedlander, 2008; Ravasi & Vasconcelos, 2020).

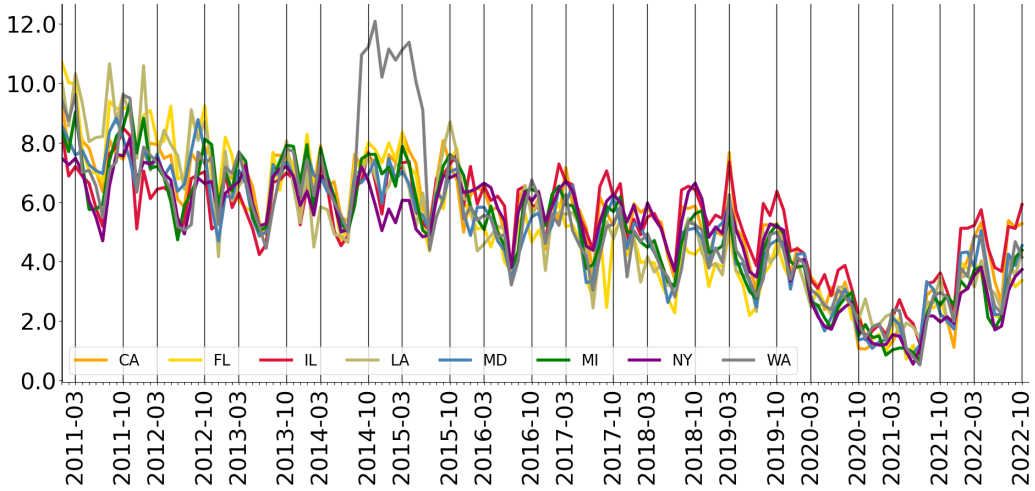

Figure 11. Temporal traces of one of the BBs SiBBlInGS identified, which includes college admission terms, show bi-yearly peaks around March and October, aligning with key milestones in the US college admissions process. Additionally, a decrease in online interest in the college BB is observed during the COVID-19 pandemic.

I.3. Temporal traces of college BB

The temporal traces of the BBs that relate to college admission, as identified by SiBBlInGS, exhibit distinct bi-yearly peaks, with notable increases in activity around March and October, along with a clear decrease between March to next October (Fig. 11). These peaks align with key periods in the US college admissions cycle, including application submission and admission decision releases. Particularly, around the end of March, many colleges and universities in the US release their regular admission decisions, prompting increased population interest. Similarly, October marks the time when prospective students typically start showing increased interest in applying to colleges, as many colleges have early application deadlines that fall in late October or early November. The bi-yearly peaks pattern in March and October thus reflects the concentrated periods of activity and anticipation within the college admissions process. External factors such as the COVID-19 pandemic can also influence the timing and dynamics of the college admissions process, as we observe by the decrease in the college BB activity during the pandemic period (Fig. 11).

I.4. Temporal traces of "Passover" BB

SiBBInGS identified a "Passover" BB, characterized by temporal traces that show a clear alignment with the timing of Passover, which usually occurs around April. The time traces demonstrate a prominent peak in states with higher Jewish population percentages, like CA, FL, and NY, as presented by the average peak value in Figure 12B plotted for the different states. The peaks detection (in Fig. 12) was done using scipy's (Virtanen et al., 2020) "find_peaks" function with a threshold of 4.

J. Neural Data—Additional Information

J.1. Neural Data Pre-Processing

In this experiment we used the neural data collected from Brodmann's area 2 of the somatosensory cortex in a monkey performing a reaching-out movement experiment from Chowdhury et al. (Chowdhury & Miller, 2022; Chowdhury et al., 2020). While the original dataset includes data both under perturbed and unperturbed conditions, here, for simplicity, we used only unperturbed trials. We followed the processing instructions provided by Neural Latents Benchmark (Pei et al., 2021) to extract the neural information and align the trials. The original neural data consisted of spike indicators per neuron, which we further processed to approximate spike rates by convolving them with a Gaussian kernel.

For each of the 8 angles, we randomly selected 18 trials, resulting in a total of 144 data matrices. The states were defined as the angles, and for learning the supervised $\boldsymbol{P}$, we used as labels the x-y coordinates of each angle in a circle with a radius of 1 (i.e., sine and cosine projections).

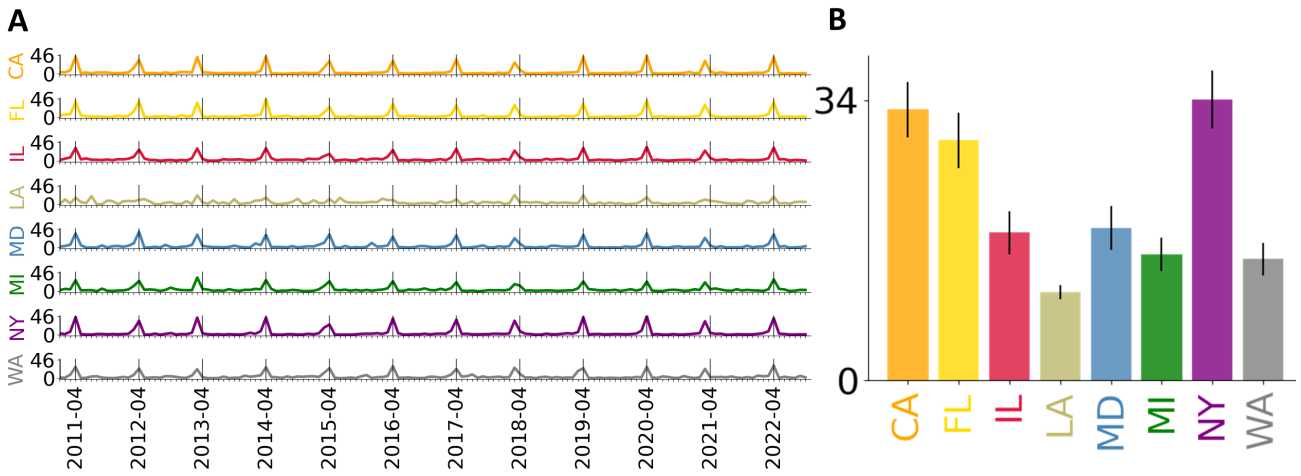

Figure 12. **Temporal trace of Passover BB**. The Passover's BB patterns, as SiBBlInGS found, show an alignment with the percentage of Jewish population in different states. **A** Temporal traces of the Passover BB for each state. Vertical black lines indicate the month of April, when Passover is usually celebrated. **B** The mean and standard error of peak values for each state.

J.2. Experimental details for the neural data experiment

We ran SiBBLInGS on the reaching-out dataset with $p = 4$ BBs. The $\boldsymbol{\lambda}$'s parameters used were $\epsilon = 2.1$, $\beta = 0.03$, and $w_{\text{graph}} = 10.1$. For the regularization of $\boldsymbol{\Phi}$ we used: $\gamma_1 = 0.001$, $\gamma_2 = 0.001$, $\gamma_3 = 0.1$, and $\gamma_4 = 0.3$ and we set $\boldsymbol{\nu}$ to be a vector of length $p = 4$ with $\nu_1 = 0.8$ (to allow more flexibility in the first BB), and $\nu_k = 1$ for $k = 2, 3, 4$. For the neural data, we used the supervised version of $\boldsymbol{P}$, where the $x - y$ coordinates are used as the labels for calculating $\boldsymbol{P}$. During each iteration, $\boldsymbol{A}$ underwent two updates within each state. We chose $k = 7$ neighbors for the channel graph reconstruction, and used Python scikit-learn's (Pedregosa et al., 2011) LASSO solver for the update of $\boldsymbol{A}$.

J.3. State prediction using temporal traces

We used the identified temporal traces $\boldsymbol{\Phi}$ to predict the state (hand direction). The dimensionality of each state's temporal activity $\boldsymbol{\Phi}^d$ was reduced to a vector of length $p \times 4 = 16$ using PCA with 4 components. A k-fold cross-validation classification approach with $k = 4$ folds was used in a multi-class logistic regression model with multinomial loss (trained on 3 folds and used to predict the labels of the remaining fold). This process was repeated for each fold, and the results were averaged. The confusion matrix and accuracy scores for each state (angle), are shown in Figure 4C and in Figure 13F.

J.4. Computation of $\rho_{\text{within/between}}$

To compute the correlation for the "within" state case, a random bootstrap approach was used, such that, for each state, we randomly selected 100 combinations of temporal trace pairs corresponding to the same BB from different random trials within the same state. We computed the correlations between these temporal trace pairs, and averaged the result over all 100 bootstrapped pairs to obtain the average correlation.

Similarly, for the "between" states case, we repeated this procedure for pairs of trials from distinct states. Particularly, we selected 100 random bootstrapped combinations of pairs of the same BB from trials of different states. In Figure 13C, the average correlations are shown for each BB. The ratio depicted in Figure 4E represents the ratio between the averages of the "within" and "between" state correlations.

K. Epilepsy—Additional Information

K.1. Data Characteristics and Pre-processing for SiBBlInGS Analysis

The Epilepsy EEG experiment in this paper is based on data kindly shared publicly in (Handa et al., 2021).

The data consist of EEG recordings obtained from six patients diagnosed with focal epilepsy, who were undergoing

presurgical evaluation. As part of this evaluation, patients temporarily discontinued their anti-seizure medications to facilitate the recording of habitual seizures. The data collection period spanned from January 2014 to July 2015. These seizures manifest different patients, seizure types, ictal onset zones, and durations.

The EEG data, as described by (Handa et al., 2021), were recorded using a standard 21 scalp electrodes setup, following the 10-20 electrode system, with signals sampled at a rate of 500 Hz. To enhance data quality, all channels underwent bandpass filtering, with a frequency ranging from 1/1.6 Hz to 70 Hz. Certain channels, including Cz and Pz, were excluded from some recordings due to artifact constraints.

Here, we focused on the EEG data from an 8-year-old male patient. This patient experienced five recorded complex partial seizures (CPS) in the vicinity of electrode F8. The EEG data for this patient includes both an interictal segment during which no seizures are recorded and 5 ictal segments representing seizures. ,

To prepare the data for compatibility with the input structure of SiBBlInGS, we divided the epileptic seizure data into non-overlapping batches, with a maximum of 8 batches extracted from each seizure. Each batch had a duration of 2000 time points, equivalent to 4 seconds. This process resulted in 4 seizures with 8 batches each and one seizure with 7 batches due to its shorter duration.

For each seizure, we also included data from the 8 seconds preceding the marked clinical identification of the seizure. This amounted to 2 additional 2000-long batches (each corresponding to 4 seconds) before each seizure event.

Regarding normal activity data, we randomly selected 40 batches, each spanning 4 seconds (2000 time points), from various time intervals that did not overlap with any seizure activity or the 8-second pre-seizure period.

In total, we had 40 batches of normal activity, 39 batches of seizure activity, and 10 batches of pre-seizure data.

We ran SiBBlInGS on this data with $p = 7$ BBs. For the state-similarity graph ($\boldsymbol{P}$), we adopted a supervised approach to distinguish between seizure and non-seizure states, as detailed in the categorical case in Appendix B.1.1, where we assigned a strong similarity value constraint to same-state trials and lower similarity values to different-state trials.

We also leverage this example to underscore the significance of the parameter $\boldsymbol{\nu}$ in the model's ability to discover networks that emerge specifically under certain states as opposed to background networks. In this context, we defined here $\boldsymbol{\nu} = [1, 1, 1, 1, 1, 1, 0]$ such that the similarity levels of the 1st to 6th BBs are determined by the relevant values in $\boldsymbol{P}$, while the last BB's similarity is allowed to vary between states.

During the training of SiBBlInGS on this data, we adopted a training strategy where 8 random batches were selected in each iteration to ensure that the model was exposed to an equal number of trials from each state during each iteration and enhancing its robustness.

L. Additional Details about the Baselines

Initial Extraction of BBs from each method: To compare SiBBlInGS with other methods (as presented in Fig. 10, 2, 5), we took the following approach. For PCA global, we applied PCA on the entire dataset after horizontally concatenating the time axis using the sklearn (Pedregosa et al., 2011) implementation, specifying the number of Principal Components (PCs) to match the number of BBs in the ground truth data ($p = 10$). These PCs were then treated as the BBs. In the case of PCA local, we followed a similar procedure. However, we ran PCA individually for each state.

For Sparse PCA global (SPCA global) and Sparse PCA local (SPCA local) we used its sklearn implementation, while tuning the sparsity level on the BBs (the α parameter) to match the sparsity level of the ground truth data. Similar to the regular PCA, SPCA global refers to applying a single SPCA on the entire dataset with $p = 10$ components, while and Sparse PCA local (SPCA local) refers to applying SPCA to the observations of each state.

For dPCA, we used the Python implementation offered here (Kobak et al., 2016) with $k = 3$ states and $p = 10$ components. We chose to protect the time axes within each trial against shuffling ("dpca.protect = ['t']") and extracted the temporal traces using the stimulus component ("s" key) of the trained model.

For Tucker and PARAFAC, we utilized the Tensorly library (Kossaifi et al., 2021) with a rank set to $p = 10$ (the number of BBs allowed by SiBBlInGS). We interpreted the BBs as the first factor (factors[0] in Tensorly output), and we considered the temporal traces as the second factor (factors[1] in Tensorly output) while multiplying them by the corresponding weights from the state factor (third factor, factors[2]) to enable cross-state flexibility to these temporal traces.

For mTDR (Aoi & Pillow, 2018; Aoi et al., 2020), we first note that this method focuses on a slightly different problem than SiBBlInGS, specifically tailored for cases where multiple conditions influence each trial simultaneously. Hence, in our comparative analysis, we first adapted mTDR to be comparable with SiBBlInGS by applying the following processing steps. Given that the synthetic example (Fig.2) involves categorical rather than ordinary sequential states, we changed the categorical states to dummy variables using one-hot encoding before running mTDR. We then ran mTDR on the concatenation of all trials to obtain the temporal basis matrices ($\boldsymbol{S}$ as denoted in (Aoi & Pillow, 2018)) and their neuron-specific weights $\boldsymbol{W}$. We recalculated the optimal coefficients based on the identified $\boldsymbol{S}$ and $\boldsymbol{W}$ to minimize the Mean Squared Error (MSE) in reconstructing each trial, and then obtained the state-specific temporal activity through optimal re-weighting of $\boldsymbol{S}$. We used the reweighed optimal $\boldsymbol{S}$ to compare it with $\boldsymbol{\Phi}$ from our paper's notation, while mTDR's $\boldsymbol{W}$ served as the structural matrices for comparison with $\boldsymbol{A}$ in our notation.

For NONFAT (Wang & Zhe, 2022b), we used the code shared by the authors at (Wang & Zhe, 2022a). The model was executed with the same parameters as specified in (Wang & Zhe, 2022b), but with rank set to 10 to align with the desired BBs. The algorithm was trained for 500 epochs across 10 folds. BBs were extracted from the two views of the "Z_{arr}" matrix during the last epoch. The first view was reweighted using the weights obtained from the second view of "Z_{arr}" for each state and BB. Temporal traces were then extracted from the "U_{arr}" matrix to calculate the trace of each BB under each state.

For NNDTN (discrete-time NN decomposition with nonlinear dynamics, as implemented by (Wang & Zhe, 2022a)), we concatenated individual components of "$\boldsymbol{v}_{i_n}$" over states over the number of BBs across all time points with re-weighting by "U_{vec}". The traces were then obtained by optimizing the BBs' activity to minimize the distance between the reconstruction and the original tensor.

Post-processing steps applied to baselines' BBs and traces to align them with the ground truth:

- *Synthetic Data*:

  To assess and compare the results of these alternative methods against the ground truth BBs and traces, we initially normalized the BBs to fit the range of the ground truth BBs, applied sparsity using hard-thresholding such that the identified BBs from each method will present similar sparsity level to that of the ground truth, and then reordered the BBs to maximize the correlations of their temporal traces with the ground truth traces. This alignment was necessary since SiBBlInGS is insensitive to the ordering of BBs. For the correlation comparisons ($\rho(\boldsymbol{A}, \widehat{\boldsymbol{A}})$), we examined the correlation between the BBs, as well as their temporal traces, in comparison to the ground truth. Recognizing that correlation might not be the most suitable metric for sparse BBs comparison, we further evaluated the performance using the Jaccard index as well.

- *EEG and Trends Experiments:*

  Similar to the synthetic data scenario, in the EEG and Trends experiments, we compared the identified components with outcomes generated by different tensor and matrix factorization methods. However, since these experiments are built on real-world data, in these case, no ground truth exists for the ensembles, as the 'real' ensembles are unknown and hence the evaluation of the identified structures is qualitative. Specifically, after applying the baselines to the EEG/Trends data, we extracted the BBs ($\boldsymbol{A}$) by the following: In the cases of global and local PCA, these BBs were treated as the Principal Components (PCs). In the PARAFAC and Tucker tensor-factorization methods, they were considered the first factor (factors[0] from the tensorly output), weighted by the relevant components from the third factor (the states axis, factors[2]). We then performed the following steps: 1) Normalized the matrices to ensure that each BB had a similar absolute sum of its columns, resulting in BBs of comparable magnitudes for state comparison, and 2) Introduced artificial sparsity into the matrices through hard thresholding, aiming to achieve the same level of sparsity observed in SiBBlInGS for each state. As seen in Figure 5, in the EEG experiment, the baselines failed to detect the emergence of BBs around electrode F8, resulting in widespread non-specific clusters; As seen in Figure 10, in the Trneds experiment, these methods produced less meaningful BBs than SiBBlInGS.

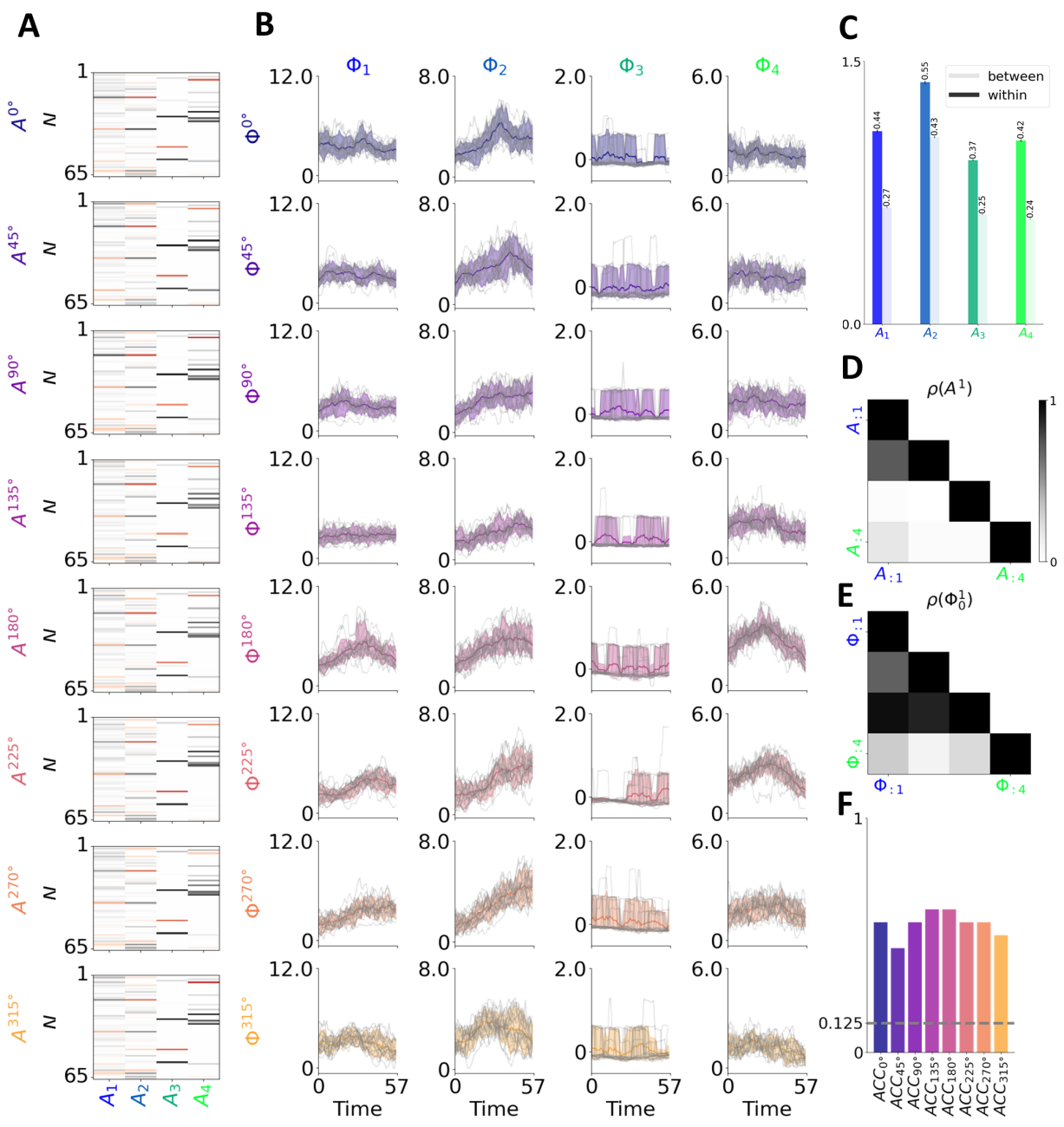

Figure 13. **Additional Figures for the Neural Recordings Experiment**. **A** The identified BBs for the different states. While there is clear consistency, slight modifications can be observed across states, capturing the natural variability in neural ensembles corresponding to different tasks. **B** Temporal traces of the identified BBs, shown with a 90% confidence interval (background color), and all trials are plotted in light gray. The color corresponds to the state color used in Figure 4. We observe adaptation over the states as well as differences between the temporal traces of BBs within a given state. The third BB exhibits significantly lower activity compared to the others (see also Figure 4), suggesting that it might capture general background trends or noise. **C** Within and between temporal trace correlations (averaged over 100 bootstrapped examples) with standard error, colored according to the BB color (as used in Fig. 4), and transparency representing the strength of the between (opaque) and within (less opaque) correlations. **D** Example of the correlations between each pair of BBs within the 1-st state (0°). This shows that while some BBs are orthogonal, others are not. **E** Example of within-state correlations between each pair of temporal traces of the BBs within the 1-st trial of the 1-st state (0°), showing that the temporal traces are neither orthogonal nor overly correlated. **F** Accuracy in predicting the state using only the temporal traces of that state as input (colored by the state color). While the random accuracy would be $1/\text{length(labels)} = \frac{1}{8} = 0.125$, the achieved accuracies are significantly higher for all states.