POISSON SPARSE COMPONENT ANALYSIS: A METHOD TO PARSE SINGLE-TRIAL NEURAL POPULATION ACTIVITY INTO FACTORS REFLECTING DISTINCT NEURAL PROCESSES

Anonymous authors
Paper under double-blind review

ABSTRACT

A central aim in systems neuroscience is discovering the distinct neural processes instantiated within the spiking activity of neural populations. Doing so within single trials or unconstrained behaviors presents dual challenges: an analysis method must disentangle the distinct signals mixed within individual neurons, while simultaneously denoising that activity from the stochastic spiking within large populations. To tackle these challenges, we present Poisson Sparse Component Analysis (pSCA), an unsupervised approach for discovering latent factors reflecting distinct neural processes in spiking datasets. pSCA seeks latents that occur sparsely in time and reside in orthogonal subspaces, enabling the decomposition of population activity into signals reflective of distinct neural processes without prior knowledge of these processes' identity or timing. Through a convolutional autoencoder architecture and masked training procedure, pSCA recovers denoised latent factors that align with identifiable task events and behavioral variables in multiple datasets across mouse and monkey cortex. Without supervision, pSCA finds a decomposition of spiking activity in which simpler components combine to produce more complex behaviors, exposing compositional structures alternative approaches do not yield.

1 INTRODUCTION AND RELATED WORK

Neural processes can often be understood at the level of latent factors, signals shared across groups of neurons, which are found using dimensionality reduction techniques (Vyas et al., 2020; Duncker & Sahani, 2021; Cunningham & Yu, 2014; Ruff et al., 2018; Khona & Fiete, 2022; Williamson et al., 2019). Commonly, some form of supervision is employed when using these techniques to discover particular neural processes of interest. For example, when discovering that movement preparation and execution were distinct processes in motor cortex belonging in orthogonal subspaces, researchers fit principal components analysis (PCA) to known time periods during which these processes occur, and compared subspaces during these periods (Wang et al., 2018; Panichello & Buschman, 2021; Elsayed et al., 2016). Researchers also commonly use explicitly supervised methods like demixed PCA (Kobak et al., 2016) or targeted dimensionality reduction (Mante et al., 2013; Aoi et al., 2020) to find latents reflecting neural processes related to specific experimental variables of interest. However, there are many situations in which researchers may want to discover latents belonging to distinct neural processes in an unsupervised manner. For instance, a researcher may be interested in identifying how flexible behaviors are composed of smaller distinct neural processes, without explicitly knowing which neural processes are involved or how they are arranged (i.e. compositionality (Tafazoli et al., 2026; Driscoll et al., 2024; Lazzari & Saxena, 2025; Kashefi et al., 2025)). Unsupervised methods are especially important in more complex naturalistic datasets that lack trial structure, when it may be nearly impossible to know, a priori, how to parse the behavioral variables that are relevant for neural activity.

However, most common unsupervised techniques, like PCA, mix distinct processes across dimensions (as do individual neurons). For instance, returning to the example of movement preparation and execution, when fitting PCA across an entire reaching dataset in an unsupervised manner, indi-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

vidual PCs reflect both preparation and execution, preventing scientific discovery of these as distinct processes. Recently, Zimnik et al. (2026) developed Sparse Component Analysis (SCA), which demonstrated an unsupervised approach for demixing neural processes across latents. By encouraging sparsity of the latents, and orthogonality of the latent-neuron mappings, SCA was able to separate distinct processes into distinct latents - for instance, separating movement preparation and execution. However, this approach (as in PCA and standard dimensionality reduction approaches), assumes that a user is seeking to model trial-averaged or pre-smoothed data rather than binned spiking activity. It is not designed to capture single-trial variability or naturalistic datasets that cannot be averaged - regimes in which unsupervised approaches could be especially valuable.

As there has been a growing interest in understanding single trials and naturalistic experiments without trial structures (Krakauer et al., 2017; Datta et al., 2019; Williams & Linderman, 2021), there have been several approaches developed to predict binned spiking activity (Pandarinath et al., 2018; Yu et al., 2009; Macke et al., 2011; Petreska et al., 2011; Kim et al., 2021; Jensen et al., 2021). These have consistently shown great benefit versus the common approach of presmoothing data. For instance, Gaussian Process Factor Analysis (GPFA) (Yu et al., 2009), one of the first and most commonly used approaches, includes a Gaussian-process prior with learnable GP timescales on the latents so that smooth single-trial latents are inferred. A more recent approach that leads to state-of-the-art predictions of neurons' firing rates is Latent Factors Analysis via Dynamical Systems (LFADS; Pandarinath et al., 2018) and its extensions, which model nonlinear dynamical systems occurring within the latent space in order to produce denoised latents. However, these methods prioritize denoising binned spiking activity and do not seek interpretable latent factors. GPFA and LFADS latents are sometimes projected post hoc onto subspaces tied to particular behaviors or task variables (e.g. Oby et al. (2025)), but like other supervised approaches, this requires knowing details about those variables *a priori*, which is especially difficult in naturalistic behavioral paradigms.

To address this, we introduce Poisson Sparse Component Analysis (pSCA), an unsupervised method for learning smooth, temporally dissociated latent representations directly from noisy spiking activity. In single-trial neural recordings from monkey motor cortex during hand-cycling, pSCA recovers latent factors that cleanly separate distinct behaviorally relevant neural processes, such as those related to preparation, movement, and stopping, where the corresponding factors from GPFA and LFADS remain entangled. This separation comes at virtually no cost to predictive power in this dataset: pSCA matches LFADS and exceeds GPFA when predicting held-out neurons on held-out trials, while training in a fraction of the time. When applied to Neuropixels recordings from a mouse, pSCA uncovers the composable neural building blocks that are shared across, and unique to, individual behaviors, along with facilitating targeted hypotheses about how the cortex and striatum differentially encode naturalistic behaviors. pSCA thus provides an efficient, accessible method that recovers generalizable, demixed latent representations of neural population activity.

2 METHODS

2.1 BACKGROUND: SPARSE COMPONENT ANALYSIS

Recent work introduced Sparse Component Analysis (SCA), an unsupervised linear autoencoder framework that demixes neural computations by encouraging sparsity on the latent factors and orthogonality on the 'loadings' (latent-neuron mappings) (Zimnik et al., 2026). Why do sparsity and orthogonality help demix underlying neural processes? In many systems, distinct neural processes—such as motor preparation and execution—are temporally flexible, occurring at partially or completely non-overlapping times, and reside in nearly orthogonal subspaces (Lara et al., 2018; Remington et al., 2018; Panichello & Buschman, 2021; Kobak et al., 2016; Kirk et al., 2023; Warriner et al., 2022; Inagaki et al., 2022; Tafazoli et al., 2026). If two processes always occurred at completely overlapping times in overlapping subspaces, we likely wouldn't consider them distinct. Thus, inferring latents that are temporally distinct and occupy distinct subspaces can help to discover distinct neural processes.

We can find distinct latent factors by applying a sparsifying ℓ_1 penalty to the K-dimensional inferred latent time series $\mathbf{Z} \in \mathbb{R}^{T \times K}$. Encouraging $\mathbf{Z}$ to be as sparse as possible while reconstructing neural activity means as few latents as possible are active at any given time, so each latent tends to capture a distinct neural process active only at specific times (e.g., movement preparation or execution). We further encourage each latent's loading to be orthogonal to those of the other latents, reflecting

the assumption that distinct neural processes occupy orthogonal subspaces to avoid interfering with one another. For more of a mathematical intuition on why sparsity and orthogonality help to demix neural processes in time, see §A.1. It is important to note that decomposing neural activity into a product of two matrices, where one is encouraged to be sparse, is a well-established technique. Both SCA (Zimnik et al., 2026) and our method (pSCA) are specific instances of what is called dictionary learning. We detail how pSCA fits into this wider literature in Section A.4.6.

2.2 Poisson Sparse Component Analysis Overview

While the original SCA framework yields demixed factors on trial-averaged datasets, if applied to single-trial or non-trial-structured data, it would rely on arbitrary, fixed pre-smoothing of the neural activity. Such pre-smoothing can obscure intrinsic neural timescales, remove important high-temporal-resolution signals, and actively interfere with the demixing of relevant computations (as we show in Fig. 2C column 4 and Appendix Fig. A.3).

We introduce **P**oisson **S**parse **C**omponent **A**nalysis (pSCA), an unsupervised method for learning smooth, temporally dissociated latent representations directly from noisy binned spiking activity. As in SCA, pSCA leverages an autoencoder framework and encourages sparse latents plus orthogonal loadings, and expands upon this framework with the addition of four key features: (1) a Poisson observation model; (2) learnable per-latent convolutional filters and (3) coordinated dropout, which together provide a new form of latent denoising (without coordinated dropout, the filters collapse to delta functions; §A.3.3); and (4) an adaptive sparsity penalty that enables out-of-the-box training without hyperparameter tuning, unlike standard approaches. We outline each in more detail below.

2.3 Observation model

We begin by describing the mapping from latent factors to observed neural activity. Let $\mathbf{y}_t \in \mathbb{Z}_{\geq 0}^N$ be a vector representing the binned spiking activity of N neurons at time t. Let $\mathbf{z}_t \in \mathbb{R}^K$ denote the learned latent factors. We model observed spike counts with a Poisson observation model whose rate is a nonlinear function f of a linear readout from the latents given by $\mathbf{y}_t \sim \mathrm{Poisson}(f(\mathbf{V}\mathbf{z}_t + \mathbf{b}))$, where $\mathbf{V} \in \mathbb{R}^{N \times K}$ and $\mathbf{b} \in \mathbb{R}^N$ parameterize the linear mapping and f is a point-wise nonlinearity. We choose $f = \exp(\cdot)$ adopting the standard nonlinearity utilized in Generalized Linear Models (GLMs) of neural spiking. Finally, to prevent the encoder from shrinking the latents arbitrarily while the decoder compensates — a degeneracy that would trivially minimize the sparsity penalty — we constrain the columns of $\mathbf{V}$ to unit norm. Importantly, including a nonlinear mapping from the latents to predicted firing-rates does not affect the geometric intuition developed in the SCA framework and §A.1 for how sparsity helps to demix distinct neural processes (§A.1.1).

2.4 Encoding model

To learn latents, we first apply a two-layer MLP pointwise in time, $\mathbf{z}_{\mathrm{enc},t} = \phi(\mathbf{y}_t)$, with a ReLU nonlinearity on the hidden layer and a hidden dimension equal to the number of neurons N. The output of the MLP is $\mathbf{Z}_{\mathrm{enc}} \in \mathbb{R}^{T \times K}$, whose k-th column $\mathbf{z}_{\mathrm{enc}}^k \in \mathbb{R}^T$ represents the trajectory of latent factor k over all T timepoints. To give the encoding model the capacity to denoise and smooth the spiking data, we then convolve each column with a learned temporal filter $\mathbf{g}^k \in \mathbb{R}^L$ of width L, giving $\mathbf{z}^k = \mathbf{g}^k * \mathbf{z}_{\mathrm{enc}}^k$. Each latent $\mathbf{z}_t^k$ is thus a weighted sum of $\mathbf{z}_{\mathrm{enc}}^k$ over a local temporal window of length L around t, with the weights given by $\mathbf{g}^k$. The filter weights are learned independently at each window position and undergo sign-preserving ℓ_1 normalization, $\mathbf{c}_k \leftarrow \mathbf{c}_k / \|\mathbf{c}_k\|_1$, which fixes the filter's overall scale without constraining individual weights in sign. While the filters are unconstrained, we initialize them to be Gaussian in shape. The latent time-series resulting from the convolution $\mathbf{Z} \in \mathbb{R}^{T \times K}$ is passed to the decoder described in §2.3, and serves as the primary point of interpretation for all analyses of the datasets in the experiments described below. Importantly, because each filter is learned independently per latent, the model adapts to the distinct timescales of different neural processes rather than imposing a fixed smoothness prior.

2.5 Loss function

We train pSCA by minimizing a loss function combining reconstruction, sparsity, and orthogonality terms. Reconstruction is measured by the Poisson negative log-likelihood, $\mathcal{L}_{\mathrm{recon}} = \hat{\mathbf{y}} - \mathbf{y} \ln \hat{\mathbf{y}}$,

where $\hat{\mathbf{y}} = f(\mathbf{V}\mathbf{z}_t + \mathbf{b})$ is the predicted firing rate. Rather than directly reconstructing the inputs to the decoder, we leverage a training procedure called coordinated dropout (Keshtkaran et al., 2022) explained in §2.6. To encourage the model to learn latent factors that represent neural processes that are temporally distinct, we encourage sparsity using an ℓ_1 penalty on the post-convolution latents (as described above), $\lambda_{\text{sparse}}\|\mathbf{Z}\|_1$. As past work has shown distinct neural processes commonly use orthogonal subspaces to avoid interfering with one another, and more generally to encourage the latent factors to represent distinct neural processes, we additionally penalize off-diagonal correlations between decoder columns via $\lambda_{\text{orth}}\|\mathbf{V}^\top\mathbf{V} - \mathbf{I}\|_F^2$, where λ_{orth} is a hyperparameter. This makes the full pSCA objective: $\mathcal{L} = (\hat{\mathbf{y}} - \mathbf{y}\ln\hat{\mathbf{y}}) + \lambda_{\text{sparse}}\|\mathbf{Z}\|_1 + \lambda_{\text{orth}}\|\mathbf{V}^\top\mathbf{V} - \mathbf{I}\|_F^2$. Rather than use a fixed λ_{sparse}, we use an adaptive penalty detailed in §2.7, so this hyperparameter does not need to be tuned. We set λ_{orth} to be 0.1% of the initial reconstruction loss in all experiments, which we have consistently found promotes approximately orthonormal loadings without harming reconstruction.

2.6 Coordinated dropout

Coordinated dropout, introduced by (Keshtkaran et al., 2022), applies a binary mask σ to the input matrix $\mathbf{Y} \in \mathbb{Z}_{\geq 0}^{T\times N}$ during training and evaluates the reconstruction loss on the complementary mask $(\mathbf{1} - \sigma)$ applied to $\mathbf{Y}$. The masked entries are thus encoded but not reconstructed, while the held-out entries are reconstructed but not encoded, preventing the network from learning a trivial identity mapping. In our setting, this scheme serves an additional purpose: denoising the latent variables. Because the spike-count noise at each entry of $\mathbf{Y}$ is approximately independent across neurons and timepoints, encoding such noise into the latents provides no benefit for predicting the held-out entries. The encoder is therefore pressured to discard single-entry noise, and the temporal filters on $\mathbf{Z}$ further smooth away residual fluctuations that survive the encoder. We use a default dropout rate of 50% for all experiments described below. Additionally, we note that a new binary dropout mask σ is sampled for every forward pass through the model, preventing the model from overfitting to one particular mask.

2.7 Auto-sparsity penalty

Rather than fix the sparsity coefficient λ_s in advance, we tune it dynamically over the course of training so that the sparsity penalty exerts a controlled amount of pressure relative to the reconstruction loss. At each step, we compute the gradient norms of both the sparsity and reconstruction terms with respect to the parameters of the encoder's final layer, and adjust λ_s so that the magnitude of the sparsity gradient equals that of the reconstruction gradient. This can be viewed as a variant of GradNorm (Chen et al., 2018): rather than pinning each loss term's gradient to the multi-task average, we pin the sparsity gradient directly to the reconstruction gradient, treating reconstruction as the reference signal. Importantly, this adaptive mechanism circumvents the cumbersome process of sweeping over fixed sparsity levels. By automatically balancing these gradients, the model reliably discovers the sparsest valid latent representation without compromising its ability to predict held-out data (see §A.4.7 for more details, including validation of this approach on our experimental dataset.)

3 Experiments

3.1 Synthetic data

To validate the capacity of pSCA to recover ground-truth, temporally-distinct latents from noisy spiking datasets, we first simulated population activity driven by known latent time-series. The underlying latent factors consisted of five windowed sine waves with 50% temporal overlap, shown in Fig. 1B, column 1. Simulation details are in §A.3.1.

pSCA successfully recovers and demixes the overlapping ground-truth latents into separate dimensions (Fig. 1C). Importantly, the sparsity penalty does not lead to artificial fragmentation by splitting the latents into extra dimensions (the latent in the sixth row has no activity), as the orthogonality penalty in the loss function prevents learning an extra latent collinear with the ground-truth. While Gaussian Process Factor Analysis (GPFA) learns an appropriate degree of temporal smoothing, it fails to correctly demix the latent trajectories, instead blending the latents across time. LFADS similarly mixed latents (Fig. A.2). 'Vanilla' SCA's ability to demix was variable, and caused unique

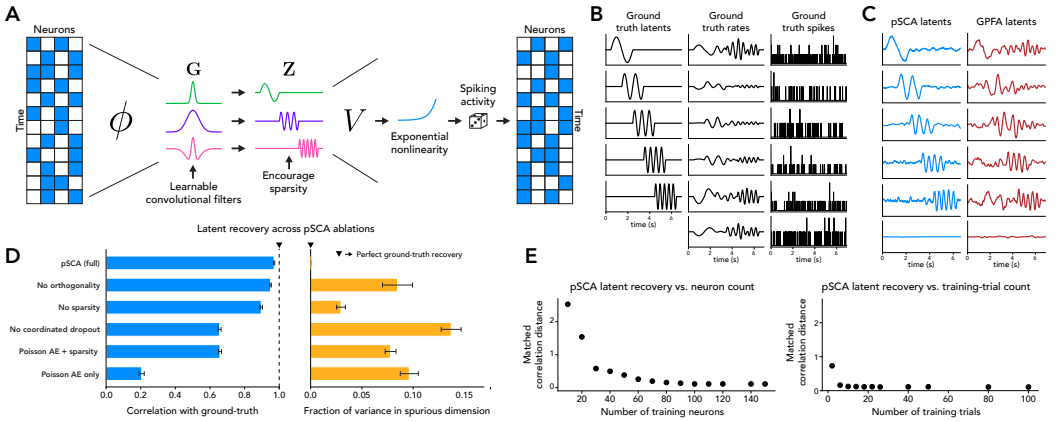

Figure 1: **pSCA accurately demixes latent structure in synthetic spiking data. (A)** pSCA leverages a convolutional autoencoder architecture and masked training procedure to infer smooth, temporally disentangled representations of latent neural processes. **(B)** We generated a set of $K = 5$ partially-temporally-overlapping ground-truth latent variables (col 1), mapped them to a high-dimensional firing-rate space (col 2; see §A.3.1), and used the result to simulate spiking activity binned at 10 ms (col 3), with an average firing-rate of 10 Hz. **(C)** Inferred latent trajectories: pSCA successfully demixes the distinct latent processes (left). In contrast, GPFA (right) learns an appropriate level of smoothing but temporally mixes the processes. **(D)** Selectively ablating any of pSCA's features results in degradations of demixing performance (left: correlation of the 5 most correlated latents with the ground-truth) and/or causes the model to populate a spurious 6th dimension (right: variance of the 6th latent that is least correlated with the ground truth - smaller is better). Error bars are s.d. over trials. **(E)** pSCA is data efficient: we computed the matched correlation distance §A.3.4 between recovered and ground-truth latents as a function of the number of training neurons (left; fixed at 100 trials) and training trials (right; fixed at 100 neurons). pSCA consistently recovers the true latents with realistic data constraints (e.g., $\geq$ 80 neurons or $\geq$ 10 trials).

concerns depending on the amount of presmoothing applied prior to running SCA (Fig. A.3). Small presmoothing left large amounts of noise, while large presmoothing removed the high-frequency ground-truth latent, demonstrating the value of pSCA's learned per-latent timescales.

In order to verify the importance of each of pSCA's features, we ablated each and computed recovery of the ground-truth latents (Figs. 1D and A.4 for viewing latents following ablations). As shown, the full pSCA architecture outperforms all ablations, highlighting the necessity of each of pSCA's features working in unison. See §A.3.3 for a more detailed discussion of the role of individual ablations, and §A.4.6 for the relation of ablated models to prior methods.

pSCA had successful latent recovery under realistic data constraints. When training with 100 6.5-second trials (≈11 minutes of data), pSCA consistently recovers the ground truth given a population of 80 or more neurons (Figure 1E, left). Conversely, for a fixed population of 100 neurons, accurate recovery is achieved with as few as 10 trials (≈1 minute of data; Figure 1E, right).

Although these simulations largely follow pSCA's assumptions, the ground-truth loadings were random rather than orthogonal, showing that pSCA tolerates mild assumption violations. pSCA also recovers the ground truth under stronger violations, including over- or under-dispersed activity, a different link function, and a non-sparse latent (§A.3.5, Fig. A.5).

3.2 Macaque motor cortex during hand-cycling

To rigorously validate pSCA, we first applied the method to a dataset in which there is a moderately mature scientific understanding - a task that has been studied using supervised approaches (Russo et al., 2018; 2020), and applying SCA to condition-averaged activity (Zimnik et al., 2026). Here, motor cortical activity was recorded while a macaque monkey turned a cycle with one hand to get to a virtual platform (Fig. 2. A, B; (Perkins et al., 2025)). The monkey was cued which direction they

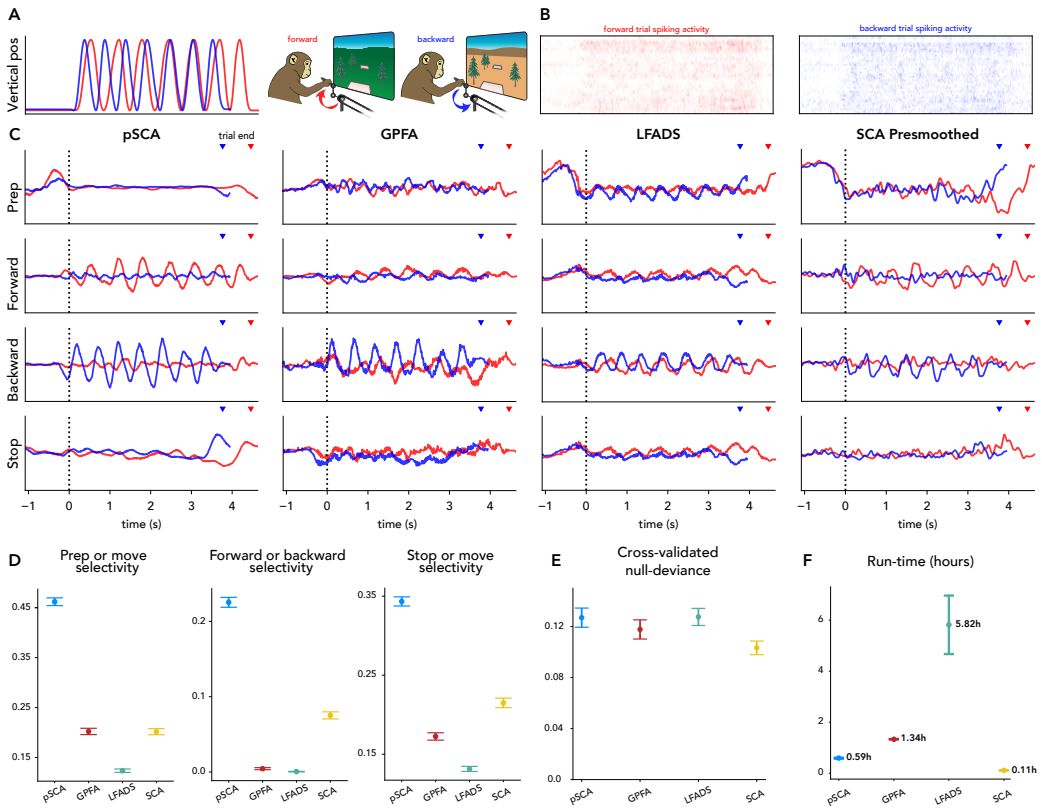

Figure 2: **pSCA uncovers separable computations during hand-cycling.** **(A)** *Right*: Monkeys turn a unimanual cycle to get from one virtual platform to the next. They are cued which direction they have to cycle (forward or backward). *Left*: Vertical position of the manipulandum on two example 7-cycle trials (the longest condition) - one forward (red), and one backward (blue). Note that the monkey will complete each individual trial with a slightly different trajectory and end-time. **(B)** Noisy spiking activity binned at 10 ms (the input to pSCA) for each example trial from panel A. **(C)** We fit four different models to this dataset: pSCA, GPFA, LFADS (AutoLFADS), and presmoothing (Gaussian kernel with 30-ms s.d.) of the data followed by SCA, using 20 latent dimensions for each model. For each row, we selected the latent from each model that was most preferentially active during the trial epoch or condition corresponding to the listed process on the left (see §A.4.1 for details). pSCA, unlike alternative approaches, consistently found latents that distinctly mapped to each process. **(D)** For each method, latents' selectivity for the time epoch of preparation versus movement (left), forward versus backward cycles (middle), or stopping versus ongoing movement (right). **(E)** Reconstruction accuracy via 5-fold bi-cross-validation, holding out both neurons and time (see §A.4.2). Error bars are standard deviations bootstrapped over predictions on held-out data. **(F)** Model run-times: mean +/- s.d. across cross-validation folds.

have to cycle (forward or backward; red and blue in Fig. 2). On different trials, the monkey had to complete a different number of cycles - 1, 2, 4, or 7.

We applied pSCA to 10-ms-binned spike counts from 112 neurons across 263 trials, fitting the model with 20 latents (results are robust for differing dimensionality - see Fig. A.13). While distinct neural processes weren't obvious at the level of individual neurons (Fig. 2B, Fig. A.12), pSCA found latents corresponding to distinct processes in an unsupervised manner. For instance, it uncovered latents corresponding to: **(1)** movement preparation (Fig. 2C, top) - latents solely active prior to movement that modulate their activity depending on the upcoming movement (forward or backward); **(2)** movement execution latents that are solely active during the ongoing movement, prior to the stop, with many of these latents preferentially active during either forward cycling (Fig. 2C, 2nd row) or backward cycling (3rd row); **(3)** latents active specifically during the final cycle of movement - putative 'stopping' latents. Crucially, the composition of neural processes found by

pSCA corresponds to results found previously when using pre-smoothed condition-averaged neural activity (Zimnik et al., 2026) (further discussed in §A.4.5), validating pSCA's single-trial findings.

Latents reflecting distinct neural processes were not found using GPFA or LFADS - visually (Fig. 2C; also see Fig. A.14 for all 20 latents from all methods) or quantitatively when determining how selective latents' activities are during one of the above behavioral epochs (e.g. preparation versus movement; Fig. 2D; §A.4.1). In general, the latents from these approaches mixed separate neural processes, making it challenging to understand the composition of processes underlying this behavior. Additionally, pre-smoothing the neurons' activities, followed by running SCA on that smoothed activity, did not lead to latents that were as demixed as pSCA (Fig. 2C,D). This was particularly apparent with regards to movement preparation, which this stepwise approach was unable to isolate as well, likely due to a limited signal-to-noise ratio without proper denoising.

Importantly, pSCA's interpretability did not come at the cost of accuracy - pSCA's cross-validated reconstruction performance was comparable to LFADS, slightly outperformed GPFA, and outperformed the stepwise presmoothing followed by SCA approach (Fig. 2E). As an additional benefit versus GPFA and LFADS (though not SCA), model run-times were substantially faster (Fig. 2F).

Beyond validating previous findings found in condition-averaged data, pSCA enables single-trial and higher-temporal-resolution insights. We highlight two such insights, in particular related to understanding movement preparation on a single-trial level. First, the monkeys performed trials in a block structure, with alternating blocks of forward and backward trials. Interestingly, some preparatory latents' activity changed depending on the trial location within a block, being attenuated on the first trial of each new block (shown in Fig. A.6).

Second, it was unclear in past work using condition-averaged responses (which can blur and average temporal information) whether 'stopping' latents were related to the preparation or execution of the final cycle (Zimnik et al., 2026). Here, it is clearer to hypothesize that one of the 'stopping' latents relates to the preparation to stop (based on its early and somewhat variable timing relative to the final cycle; Fig. A.7). Past approaches towards identifying preparation during ongoing movements relied on preparation occurring in the same subspace as an initial delay period (Lara et al., 2018; Zimnik & Churchland, 2021), while here, this putative stopping preparation is in fact approximately orthogonal to the initial movement preparation (latent shown in top row of Fig. 2, orthogonality shown in Appendix Fig. A.11). Thus, pSCA uniquely enables generating a novel hypothesis about this preparatory process - preparation to stop is distinct from the preparation to begin a movement. Reaching this hypothesis with a supervised approach would have required already suspecting that a distinct stopping-preparation process exists (which is not obvious *a priori*) and knowing precisely when it occurs. In contrast, pSCA identifies latents reflecting such distinct processes without requiring prior expectations; the researcher need only recognize these processes once pSCA finds them.

3.3 Mouse motor cortex and striatum in a naturalistic arena paradigm

After validating pSCA above on a more-established task, we next sought to demonstrate how pSCA could be valuable for scientific discovery and hypothesis generation with a less explored dataset without trial structure (Xing et al., 2024). Here, mice engaged in a naturalistic paradigm designed to promote both stereotyped, species-typical behaviors like eating and grooming, and those that challenge agility and dexterity, like climbing and walking across an irregular grid. In total, there were 10 human labeled behaviors (Fig. 3A). Neuropixels recordings were made from forelimb primary motor cortex (fM1) and dorsolateral striatum (DLS).

Applying pSCA to cortex uncovered many latents that related to individual behaviors or subsets of behaviors, enabling scientific discovery and hypotheses about the compositionality of neural coding of behaviors. To illustrate this, in Fig. 3B we computed a behavioral heatmap (§A.5.2)indicating how the activity of a given latent (columns) was distributed across different behaviors (rows); full latent time-series for selected behavioral bouts are shown in Fig. A.15. Looking at several latents, we can see that some are predominantly active within individual behaviors (e.g. latents 0 and 1 are active uniquely during eating and grooming), suggesting that neural coding of these behaviors may be fairly specialized. Importantly though, the ability to decompose activity across many latents allows us to add important nuance to this understanding, rather than simply saying these behaviors are fairly neurally-unique (which would have been possible with a supervised comparison (Xing et al., 2024)). Here, we can see that there are other latents, which are shared across all behaviors

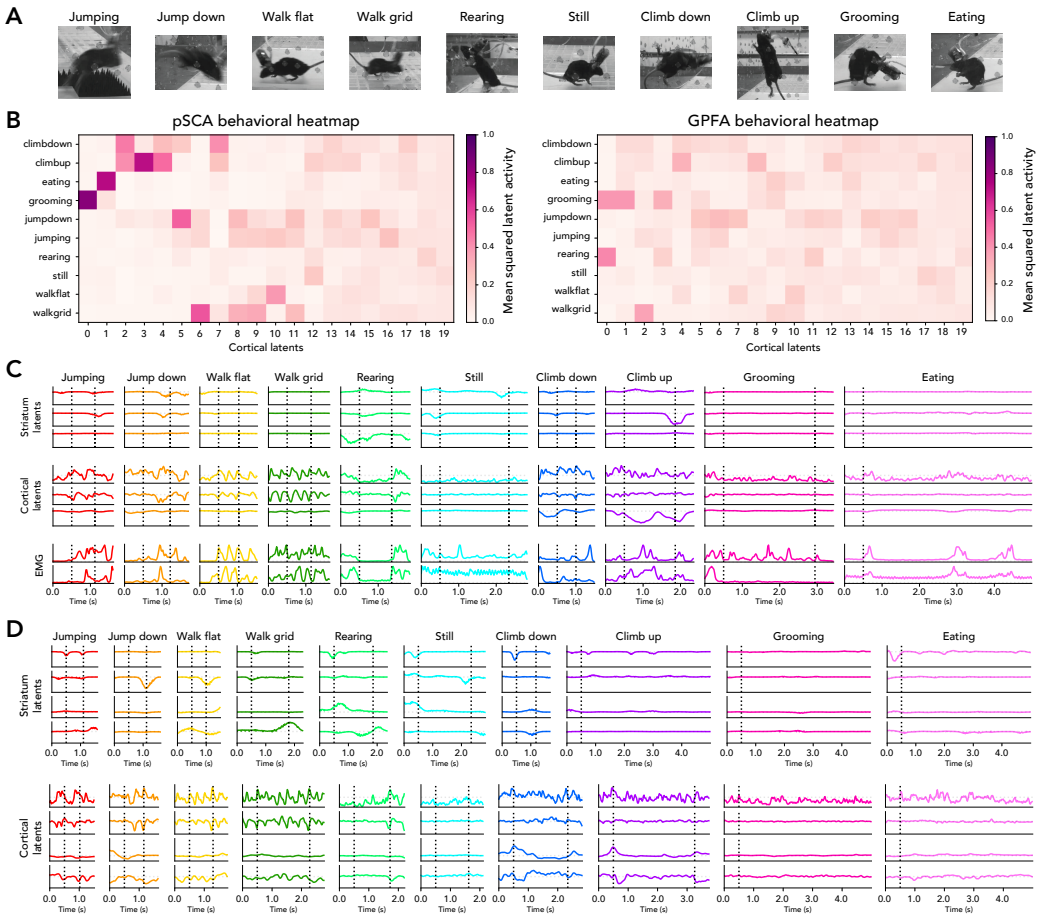

Figure 3: **pSCA latents correspond to subsets of naturalistic behaviors. (A)** Mice behaved in a naturalistic 'arena' designed to promote a variety of behaviors. **(B)** Latents' behavioral specificity (see §A.5.2). Dimensions inferred by pSCA represented patterns of neural activity mostly unique to subsets of behaviors. Latents found by GPFA were mostly spread across all behaviors. **(C)** *Top 3 rows:* Striatal latents with the 3 highest correlations with EMGs (see A.5.3). *Next 3 rows:* Cortical latents with highest EMG correlations. *Final 2 rows:* Representative EMG signals from the same behavioral bouts. Visually, the first two cortical latents appear to represent muscle activity. The ability to predict these latents from EMGs is on average $R^2 = 0.36$ for cortex and $R^2 = 0.05$ for striatum. **(D)** Latents within each region ranked according to their transition selectivity (§A.5.4). All four of the top transition-selective latents in the striatum are active at boundaries between behaviors. The latents most selective for transition points in cortex were broadly active throughout other times, although the last two latents appeared to selectively peak at climbing onset.

including grooming and eating - e.g. latent 18 in Fig. 3B, which has the highest correlation with muscle EMGs (Fig. A.15). pSCA can thus enable us to hypothesize that within fM1, eating and grooming contain largely unique neural codes, with the exception of shared muscle-related activities (a hypothesis that we could subsequently explore with targeted supervised approaches).

pSCA also provides insight into the compositional nature of neural coding of behavior. For instance, many latents are active both during climbing up and climbing down, suggesting similar neural encoding of these behaviors. The same is true for jumping across and jumping down (Fig. 3B, Fig. A.15). While encoded similarly, investigating additional latents demonstrates that their neural coding is not identical. For example, latent number 5 in Fig. 3B is active during jumping down and climbing down, but not jumping or climbing up. We can thus hypothesize that climbing down can be composed of (predominantly) neural activity patterns that are broadly active during climbing, combined with aspects that are unique for downward movements.

Related hypotheses can be generated about walking on a flat surface versus a grid - behaviors for which there is controversy about motor cortex's role (Kirk et al., 2025; Xing et al., 2022) (grid-walking is a more dexterous behavior that may be cortex-dependent, while flat-walking may not require cortex). While some are latents are co-active during both types of walking (latent 9 in Fig. 3B) - there are several latents that are much more active during grid-walking (e.g. Fig 3B, latents 6,8,11). Intriguingly, those latents are often active during jumping behaviors, which may similarly be more cortex-dependent. This again highlights how pSCA can elucidate hypotheses about the compositional nature of the neural code - grid-walking may be a mix of rhythmic locomotor (see rows 0 and 11 Fig A.15), and cortically-driven, components.

Lastly, we also applied pSCA to the striatal recordings and observed whether there were noticeable differences between regions. Two readily appeared within the pSCA latents. First, cortical latents had a much closer relation to ongoing muscle activity than did striatal latents (Fig. 3C), as found in (Xing et al., 2024). Second, striatal latents appeared more active around the time of behavioral transitions than did cortical latents (Fig. 3D). This aligns with past ideas about striatum being involved in action selection (Markowitz et al., 2018), while also allowing us to generate new hypotheses - there exist distinct transition signals localized to subpopulations of neurons in DLS, active during specific subsets of naturalistic behaviors.

Broadly, the above types of discoveries and hypotheses are not readily apparent using alternative unsupervised approaches. The vast majority of GPFA's cortical latents are co-active during all behaviors (Fig. 3B), obscuring how the neural codes for individual behaviors are composed from components shared across multiple behaviors and those unique to individual behaviors. Trying to discover such compositionality with a posthoc supervised approach on GPFA's latents would require naming a combinatorically large number of groups of behaviors (pairs, triplets, etc.) of interest, and then applying a method such as contrastive PCA (Abid et al., 2018) to find a projection of GPFA's latent space that emphasizes each one of those groupings. pSCA, by contrast, recovers this structure from a single fit, without requiring that any behavioral groupings be specified beforehand.

4 Discussion

Across single-trial and naturalistic spiking datasets, pSCA found denoised latents corresponding to distinct neural processes. This unsupervised discovery facilitated scientific hypothesis generation about how behaviors are compositionally created from separable neural processes.

To discover distinct neural processes in an unsupervised manner, pSCA makes two key assumptions: distinct neural processes occur at largely non-overlapping times, and they occupy approximately orthogonal subspaces. These assumptions hold on many real datasets (see §2.1), and their combination keeps pSCA from artificially fragmenting a single latent process into multiple highly collinear dimensions (see §A.1 and §A.3.3). Not every dataset will satisfy both, however, and pSCA may fail to demix temporally dissociable processes that occupy highly collinear subspaces. Even then, the failure is interpretable: if pSCA cannot recover sparse latents, this suggests either that (1) temporally distinct processes occupy highly collinear dimensions, or (2) processes in orthogonal subspaces occur at almost completely overlapping times. Both suggest concrete directions for further scientific exploration.

While pSCA helps to facilitate the grouping of latents into putative computational roles, it is important to note that this interpretation is generated by the method's user, rather than the method itself. As with any unsupervised method, interpreting latents is up to the user and should be further validated, such as experimentally. For example, the putative stopping-preparation latent (Fig. A.7) could be confirmed to relate to preparation rather than execution with a task where the target occasionally jumps on the final cycle to see whether stopping preparation still occurs without the execution.

Our architecture's modularity makes it straightforward to extend to broader lines of scientific inquiry. For instance, restricting the convolutional filters to be causal could enable more precise investigation into the timing of latent processes following stimuli. Furthermore, rather than modeling distinct brain regions independently and comparing them post-hoc (as done in Fig. 3C, D), the framework could be extended to jointly model inter-regional interactions. Finally, substituting pSCA's shallow readout for a highly nonlinear embedding model could adapt the framework for brain regions with more nonlinear representational geometries (e.g. hippocampus), enabling even broader use.

AI USE STATEMENT

Large language models (LLMs) were used to aid and polish the writing of this manuscript, such as refining wording and phrasing. LLMs were also used to help generate and clean up plots of our results in a few figures; all plotted data come from the authors' own analyses. LLMs were not used to formulate hypotheses, design methods or experiments, implement the method, process data, or interpret results. The authors reviewed all LLM-assisted content and take full responsibility for the manuscript.

ETHICS STATEMENT

No new data were collected from human or animal subjects as part of this work. All neural and behavioral datasets analyzed here were collected in prior studies, which are cited in the paper, under the animal-care approvals described in those original publications. As this work involves only secondary analysis of existing animal data, no additional IRB or IACUC approval was required.

REPRODUCIBILITY STATEMENT

We note that upon acceptance, we will make our code package publicly available on github and include jupyter notebooks demonstrating how to reproduce results from the paper and apply pSCA to new datasets.

REFERENCES

Abubakar Abid, Martin J. Zhang, Vivek K. Bagaria, and James Zou. Exploring patterns enriched in a dataset with contrastive principal component analysis. *Nature Communications*, 9(1):2134, 2018. doi: 10.1038/s41467-018-04608-8.

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

Chenglong Bao, Jian-Feng Cai, and Hui Ji. Fast sparsity-based orthogonal dictionary learning for image restoration. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3384–3391. IEEE, 2013.

A.S. Charles, N. Cermak, R.O. Affan, B.B. Scott, J. Schiller, and G. Mishne. Graft: Graph filtered temporal dictionary learning for functional neural imaging. *IEEE Transactions on Image Processing*, 31:3509–3524, 2022. doi: 10.1109/TIP.2022.3171414.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International Conference on Machine Learning*, pp. 794–803. PMLR, 2018.

J. P. Cunningham and B. M. Yu. Dimensionality reduction for large-scale neural recordings. *Nat. Neurosci.*, 17:1500–1509, 2014.

S. R. Datta, D. J. Anderson, K. Branson, P. Perona, and A. Leifer. Computational neuroethology: a call to action. *Neuron*, 104:11–24, 2019.

F. Diego, S. Reichinnek, M. Both, and F.A. Hamprecht. Automated identification of neuronal activity from calcium imaging by sparse dictionary learning. In *2013 IEEE 10th International Symposium on Biomedical Imaging*. IEEE, 2013.

Laura N. Driscoll, Krishna Shenoy, and David Sussillo. Flexible multitask computation in recurrent networks utilizes shared dynamical motifs. *Nature Neuroscience*, 27(7):1349–1363, 2024.

L. Duncker and M. Sahani. Dynamics on the manifold: Identifying computational dynamical activity from neural population recordings. *Curr. Opin. Neurobiol.*, 70:163–170, 2021.

G. F. Elsayed, A. H. Lara, M. T. Kaufman, M. M. Churchland, and J. P. Cunningham. Reorganization between preparatory and movement population responses in motor cortex. *Nat. Commun.*, 7: 13239, 2016.

A. Giovannucci, J. Friedrich, M. Kaufman, A. Churchland, D. Chklovskii, L. Paninski, and E.A. Pnevmatikakis. Onacid: Online analysis of calcium imaging data in real time. In *Advances in Neural Information Processing Systems*, 2017.

Raja Giryes and Michael Elad. Sparsity-based poisson denoising with dictionary learning. *IEEE Transactions on Image Processing*, 23(12):5057–5069, 2014. doi: 10.1109/TIP.2014.2362057.

H. K. Inagaki, S. Chen, M. C. Ridder, P. Sah, N. Li, Z. Yang, H. Hasanbegovic, Z. Gao, C. R. Gerfen, and K. Svoboda. A midbrain-thalamus-cortex circuit reorganizes cortical dynamics to initiate movement. *Cell*, 185:1065–1081.e23, 2022.

K. Jensen, T.-C. Kao, J. Stone, and G. Hennequin. Scalable bayesian gpfa with automatic relevance determination and discrete noise models. In *Advances in Neural Information Processing Systems*, volume 34, pp. 10613–10626, 2021.

Mehrdad Kashefi, Jonathan A. Michaels, R. Kersten, J. Lau, Jörn Diedrichsen, and J. Andrew Pruszynski. Compositional neural dynamics during reaching. *bioRxiv*, 2025.

Mohammad Reza Keshtkaran, Andrew R. Sedler, Raeed H. Chowdhury, Raghav Tandon, Diya Basrai, Sarah L. Nguyen, Hansem Sohn, Mehrdad Jazayeri, Lee E. Miller, and Chethan Pandarinath. A large-scale neural network training framework for generalized estimation of single-trial population dynamics. *Nature methods*, 19:1572–1577, 2022.

M. Khona and I. R. Fiete. Attractor and integrator networks in the brain. *Nat. Rev. Neurosci.*, 23: 744–766, 2022.

Timothy D. Kim, Thomas Z. Luo, Jonathan W. Pillow, and Carlos D. Brody. Inferring latent dynamics underlying neural population activity via neural differential equations. In *International Conference on Machine Learning*, pp. 5551–5561. PMLR, 2021.

E. A. Kirk, K. T. Hope, S. J. Sober, and B. A. Sauerbrei. An output-null signature of inertial load in motor cortex. *bioRxiv*, 2023. doi: 10.1101/2023.11.06.565869.

Eric A. Kirk, Kangjia Cai, and Britton A. Sauerbrei. Cortical sculpting of a rhythmic motor program. *bioRxiv*, 2025. doi: 10.1101/2025.06.20.660772.

D. Kobak, W. Brendel, C. Constantinidis, C. E. Feierstein, A. Kepecs, Z. F. Mainen, X. L. Qi, R. Romo, N. Uchida, and C. K. Machens. Demixed principal component analysis of neural population data. *eLife*, 5, 2016. doi: 10.7554/eLife.10989.

J. W. Krakauer, A. A. Ghazanfar, A. Gomez-Marin, M. A. MacIver, and D. Poeppel. Neuroscience needs behavior: correcting a reductionist bias. *Neuron*, 93:480–490, 2017.

A. H. Lara, G. F. Elsayed, A. J. Zimnik, J. Cunningham, and M. M. Churchland. Conservation of preparatory neural events in monkey motor cortex regardless of how movement is initiated. *eLife*, 7, 2018. doi: 10.7554/eLife.31826.

John Lazzari and Shreya Saxena. Multitasking recurrent networks utilize compositional strategies for control of movement. *bioRxiv*, 2025.

Jakob H. Macke, Lars Buesing, John P. Cunningham, Byron M. Yu, Krishna V. Shenoy, and Maneesh Sahani. Empirical models of spiking in neural populations. In *Advances in Neural Information Processing Systems*, volume 24, pp. 1350–1358, 2011.

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

Jeffrey E. Markowitz, Winthrop F. Gillis, Celia C. Beron, Shay Q. Neufeld, Keiramarie Robertson, Neha D. Bhagat, Ralph E. Peterson, Emalee Peterson, Minsuk Hyun, Scott W. Linderman, Bernardo L. Sabatini, and Sandeep Robert Datta. The striatum organizes 3d behavior via moment-to-moment action selection. *Cell*, 174(1):44–58, 2018.

Noga Mudrik, Gal Mishne, and Adam S. Charles. Sibblings: Similarity-driven building-block inference using graphs across states. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 36504–36530. PMLR, 2024.

Emily R. Oby, Alan D. Degenhart, Erinn M. Grigsby, Asma Motiwala, Nicole T. McClain, Patrick J. Marino, Byron M. Yu, et al. Dynamical constraints on neural population activity. *Nature Neuroscience*, 2025. doi: 10.1038/s41593-024-01845-7.

Bruno A Olshausen and David J Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, 1996. doi: 10.1038/381607a0.

C. Pandarinath et al. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature Methods*, 15:805–815, 2018.

M. F. Panichello and T. J. Buschman. Shared mechanisms underlie the control of working memory and attention. *Nature*, 592:601–605, 2021.

Sean M Perkins, Elom A Amematsro, John Cunningham, Qi Wang, and Mark M Churchland. An emerging view of neural geometry in motor cortex supports high-performance decoding. *eLife*, 12:RP89421, 2025. doi: 10.7554/eLife.89421.3. URL https://doi.org/10.7554/eLife.89421.3.

Biljana Petreska, Byron M. Yu, John P. Cunningham, Gopal Santhanam, Stephen I. Ryu, Krishna V. Shenoy, and Maneesh Sahani. Dynamical segmentation of single trials from population neural data. In *Advances in Neural Information Processing Systems*, volume 24, 2011.

E. D. Remington, D. Narain, E. A. Hosseini, and M. Jazayeri. Flexible sensorimotor computations through rapid reconfiguration of cortical dynamics. *Neuron*, 98:1005–1019.e5, 2018.

Ron Rubinstein, Alfred M Bruckstein, and Michael Elad. Dictionaries for sparse representation modeling. *Proceedings of the IEEE*, 98(6):1045–1057, 2010. doi: 10.1109/JPROC.2010.2040551.

D. A. Ruff, A. M. Ni, and M. R. Cohen. Cognition as a window into neuronal population space. *Annu. Rev. Neurosci.*, 41:77–97, 2018.

A. A. Russo, R. Khajeh, S. R. Bittner, S. M. Perkins, J. P. Cunningham, L. F. Abbott, and M. M. Churchland. Neural trajectories in the supplementary motor area and motor cortex exhibit distinct geometries, compatible with different classes of computation. *Neuron*, 107:745–758.e6, 2020.

Abigail A. Russo, Sean R. Bittner, Sean M. Perkins, Jeffrey S. Seely, Brian M. London, Antonio H. Lara, Andrew Miri, Najja J. Marshall, Adam Kohn, Thomas M. Jessell, Laurence F. Abbott, John P. Cunningham, and Mark M. Churchland. Motor cortex embeds muscle-like commands in an untangled population response. *Neuron*, 97(4):953–966.e8, 2018. doi: 10.1016/j.neuron.2018.01.004.

Andrew H. Song, Francisco Flores, and Demba Ba. Spike sorting by convolutional dictionary learning, 2018. URL https://arxiv.org/abs/1806.01979.

Ju Sun, Qing Qu, and John Wright. Complete dictionary recovery over the sphere. In *2015 International Conference on Sampling Theory and Applications (SampTA)*, pp. 407–410. IEEE, 2015.

Sina Tafazoli, Flora M. Bouchacourt, Adel Ardalan, Nikola T. Markov, Motoaki Uchimura, Marcelo G. Mattar, Nathaniel D. Daw, and Timothy J. Buschman. Building compositional tasks with shared neural subspaces. *Nature*, 650(8100):164–172, 2026.

S. Vyas, M. D. Golub, D. Sussillo, and K. Shenoy. Computation through neural population dynamics. *Annu. Rev. Neurosci.*, 2020.

J. Wang, D. Narain, E. A. Hosseini, and M. Jazayeri. Flexible timing by temporal scaling of cortical responses. *Nat. Neurosci.*, 21:102–110, 2018.

C. L. Warriner, S. Fageiry, S. Saxena, R. M. Costa, and A. Miri. Motor cortical influence relies on task-specific activity covariation. *Cell Rep.*, 40, 2022.

Alex H. Williams and Scott W. Linderman. Statistical neuroscience in the single trial limit. *Current Opinion in Neurobiology*, 70:193–205, 2021.

R. C. Williamson, B. Doiron, M. A. Smith, and M. Y. Byron. Bridging large-scale neuronal recordings and large-scale network models using dimensionality reduction. *Curr. Opin. Neurobiol.*, 55: 40–47, 2019.

David Xing, Wilson Truccolo, and David A. Borton. Emergence of distinct neural subspaces in motor cortical dynamics during volitional adjustments of ongoing locomotion. *Journal of Neuroscience*, 42(49):9142–9157, 2022. doi: 10.1523/JNEUROSCI.0746-22.2022.

David Xing, Joshua Glaser, and Andrew Miri. Neural dynamics hierarchy in motor cortex and striatum across naturalistic behaviors. *bioRxiv*, 2024.

Byron M. Yu, John P. Cunningham, Gopal Santhanam, Stephen I. Ryu, Krishna V. Shenoy, and Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. *J. Neurophysiol.*, 102:614–635, 2009. doi: 10.1152/jn.90941.2008.

Yuexiang Zhai, Zitong Yang, Zhenyu Liao, John Wright, and Yi Ma. Complete dictionary learning via l^4-norm maximization over the orthogonal group. *Journal of Machine Learning Research*, 21 (165):1–68, 2020.

A. J. Zimnik and M. M. Churchland. Independent generation of sequence elements by motor cortex. *Nat. Neurosci.*, 24:412–424, 2021.

Andrew J. Zimnik, Xinyue An, K. Cora Ames, A. Ulmer, A. H. Lara, A. A. Russo, et al. Sparse component analysis: A method that uncovers separable computations within neural population activity. *Neuron*, 2026. doi: 10.1016/j.neuron.2026.05.022.

A Appendix

A.1 Sparsity and orthogonality intuition

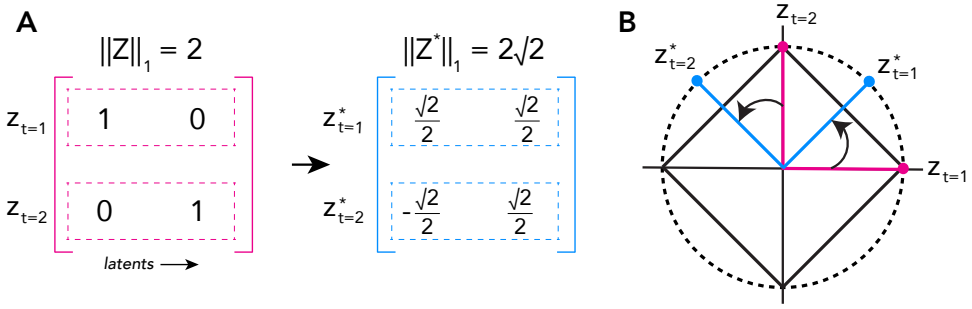

Figure A.1: **Latent sparsity demixing intuition (A)** Consider ground-truth latent processes z^1 and z^2 (left) that are preferentially active at different time points (columns represent latents, rows represent time). Inferring latents $z^\star_{t=1}$ and $z^\star_{t=2}$ (right) that are rotations away from the temporally dissociated ground-truth results in the ℓ_1 norm increasing. **(B)** The solid black diamond represents configurations of the latents that result in a constant ℓ_1 norm. Any rotation that results in latent factors $z^\star_{t=1}$ and $z^\star_{t=2}$ which includes dimensions that are co-active at the same t, will increase the ℓ_1 norm of the inferred latents.

We can build mathematical intuition for why encouraging sparsity enables us to find temporally distinct neural processes by considering a 2D latent space where the ground-truth latent dimensions, z^1 and z^2, are preferentially active during specific time points. Specifically, we can think of the

ground-truth 2D latent z_t at time points $t = 1$ and $t = 2$ as two different vectors, $z_{t=1}$ and $z_{t=2}$, each lying on the circumference of the unit-circle shown as the dotted line in Fig. A.1B. If we concatenate $z_{t=1}$ and $z_{t=2}$ across time into a matrix representing our entire latent time-series Z (as shown in the red matrix in Fig. A.1A) we see that $||Z||_1 = 2$.

Now consider inferred latent factors $Z^\star$, where $Z^\star$ is some rotation of the ground-truth latent factors in Z. As shown in blue in Fig. A.1A, if we consider a 45° rotation, that will result in $z^\star_{t=1} = [\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}]$ and $z^\star_{t=2} = [-\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}]$, making $||Z^\star||_1 = 2\sqrt{2}$. This shows that inferring $Z^\star$ for which both latent dimensions are active (non-zero) at $t = 1$ and $t = 2$ causes $||Z^\star||_1 > ||Z||_1$.

A graphical depiction of this effect can be seen in Fig. A.1B. The solid diamond represents configurations of the latent factors that produce a constant ℓ_1 norm. Rotations (lying on the dotted line) of the ground-truth latent factors (red) where both latents are active (blue) result in $||Z^\star||_1 > ||Z||_1$. Inferring $Z^\star$ for which $||Z^\star||_1$ is minimized will pull $Z^\star$ closer to the ground-truth latent factors (or reflections thereof) and thus demix latent processes in time.

We intentionally consider rotations in this toy example because we encourage the mapping from latents to observed neural activity (V in §2.3) to be orthogonal. If instead we considered only a unit-norm constraint, one could find a $Z^\star$ that minimizes $\|Z^\star\|_1$ but whose inferred latents are redundant (e.g., $z^\star_{t=1} = [1, 0]$ and $z^\star_{t=2} = [-1, 0]$), failing to recover the ground truth. Recovering the ground-truth latent processes therefore requires finding the ℓ_1-minimizing rotation subject to approximate orthogonality of the latent-neuron mappings. The importance of including orthogonality in addition to sparsity is further highlighted by our ablation tests (Figs. 1D, A.4) which illustrate that inferring latent dimensions with sparsity and no orthogonality can result in a single latent variable being fragmented across collinear dimensions, shown in rows 2 and 6 of the "no orthogonality" column in Fig. A.4.

Importantly, while this didactic example considered completely non-overlapping ground-truth processes (active either only at $t = 1$ or $t = 2$) the basic principle extends to latent processes that overlap up to $\approx 58\%$, as shown in Fig. 1 of (Zimnik et al., 2026). Therefore, if the underlying neural processes naturally occur at distinct times (whether or not they are highly sparse), imposing an ℓ_1 penalty encourages the model to demix and recover the true, temporally dissociated factors (rather than some rotation).

A.1.1 Demixing with sparsity and orthogonality in pSCA

This core demixing intuition continues to hold within the nonlinear architecture of pSCA. Because the ℓ_1 penalty is applied directly to the latent space prior to passing through the decoder, any rotation away from a set of sparse, ground-truth latents will strictly increase the penalty, exactly as it does under the linear observation model of the SCA framework. Furthermore, because we choose a f which is strictly monotonically increasing, it is injective, guaranteeing a one-to-one mapping between the linear readout and the predicted firing rates. If we were to use a non-injective function with a "dead zone" - such as a Rectified Linear Unit (ReLU) - the model could theoretically discover degenerate rotations that push $\mathbf{V}\mathbf{z}_t + \mathbf{b}$ into the negative domain of f, yielding identical reconstructions while arbitrarily altering the ℓ_1 norm. The injectivity of the exponential function prevents this, ensuring every rotation of the latent space uniquely impacts the output and preserving ability for the ℓ_1 penalty to demix non-synchronous latent processes.

A.2 Model fitting

Note: the following settings were used for all model comparisons in both the simulated dataset and the cycling dataset.

pSCA was fit directly to the raw datasets without preprocessing (aside from binning spikes). pSCA is fit using the PyTorch library, Adam optimizer, batch size of 256, learning rate of 1×10^{-3} and 10K epochs in this and all other experiments. For GPFA, we utilized the implementation provided in the `elephant` package. Because GPFA assumes a Gaussian observation model, this implementation inherently applies a square-root transform to the spike counts prior to fitting. For LFADS, we employed the PyTorch AutoLFADS framework (available on github), which leverages Population-Based Training (PBT) to automatically discover optimal hyperparameters rather than relying on manual tuning. To enable AutoLFADS' early-stopping functionality, we randomly allocated 20% of

the trials from each dataset as a validation set. Finally, because standard SCA assumes a Gaussian observation model, we pre-smoothed the binned spiking activity using a 30 ms Gaussian kernel prior to fitting - the same level of smoothing used for baseline comparisons in (Pandarinath et al., 2018). All model runtimes reported in Fig. 2F were evaluated on the same machine, utilizing an Intel Core Ultra 9 285K processor.

A.3 SIMULATED DATA

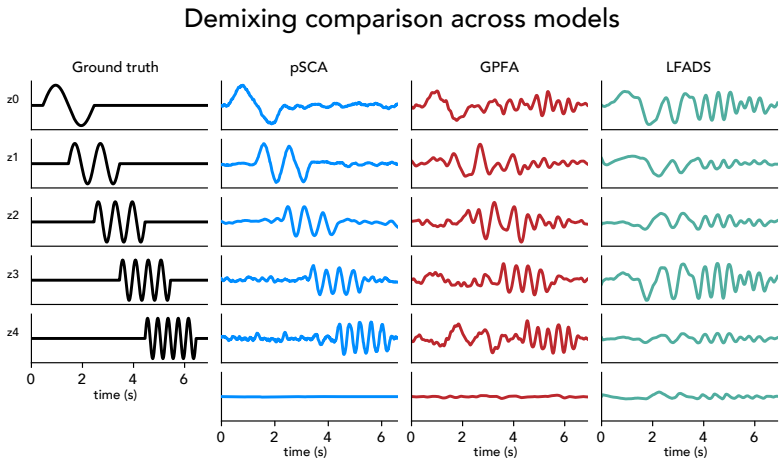

Figure A.2: **Extended comparison of latent recovery across modeling frameworks.** Reconstructed latent trajectories for five ground-truth latent dimensions across three models: pSCA, GPFA, and LFADS, each fit with six learnable latent dimensions. While pSCA successfully demixes the ground-truth signals into distinct dimensions, alternative methods fail to achieve clean separation. LFADS, like GPFA, captures the underlying oscillatory structure but temporally blends the processes across multiple latents.

A.3.1 SIMULATION DETAILS

To rigorously evaluate the latent recovery and demixing capabilities of our model, we generated a simulated dataset with known ground-truth latent processes. The simulation process was designed to emulate typical firing-rates observed in cortical spiking activity while maintaining explicit control over the underlying temporal structure.

We simulated $R = 5$ independent ground-truth latent dimensions. Each latent factor was defined as a windowed sine wave active for 2 seconds. To test each models' capacity to demix overlapping signals, the onset of each successive latent wave was shifted by 100 time bins, resulting in a 50% temporal overlap between adjacent latent processes. The simulated ground-truth latent variables can be seen in the first column of Fig. 1B.

The low-dimensional latent time-series were projected into a high-dimensional neural space via a random (non-orthogonal) projection matrix $V \in \mathbb{R}^{N \times K}$, where each entry was drawn from a standard normal distribution. For Fig. 1C and Appendix Fig. A.2, we chose $N = 150$ to illustrate that even in a data-rich regime, GPFA and LFADS fail to demix overlapping latent processes.

The resulting high-dimensional continuous rates (second column of Fig. 1B.) were passed through an exponential function to strictly enforce non-negativity. Finally, the population firing rates were globally scaled such that the average firing rate across all neurons and timepoints was 10 Hz, mirroring typical activity levels observed in cortical datasets.

Using these underlying continuous firing rates, we simulated 100 identical-length (6.5 second) trials. For each trial, discrete spike counts were generated by passing the scaled firing rates through a Poisson observation model, assuming a bin size of 10 ms ($\Delta t = 0.01$ s); example spikes shown in third column of Fig. 1B. We fit each model type with 6 learnable latent dimensions to evaluate

their capacity to infer the true underlying dimensionality ($K = 5$). Results illustrating each model's demixing capabilities are shown in Fig. A.3 and Fig. A.2.

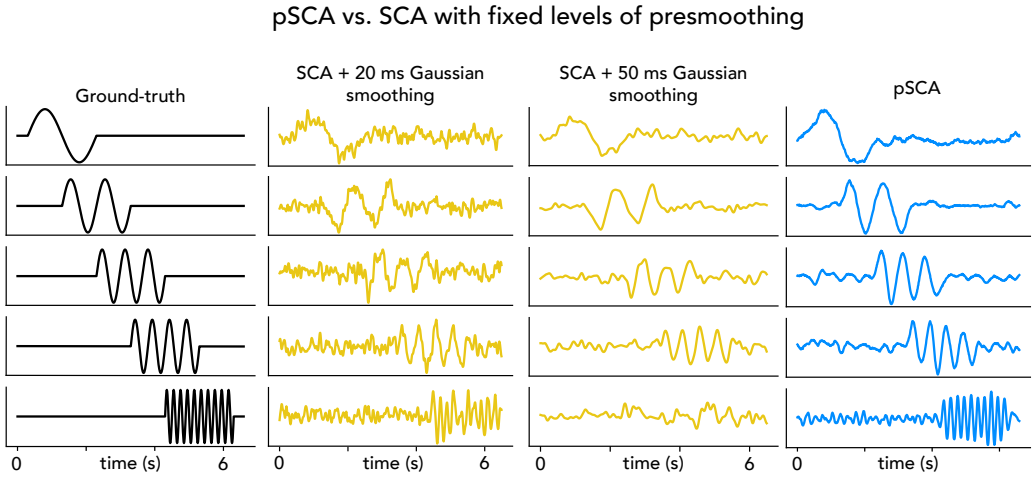

Figure A.3: **pSCA infers the correct amount of smoothing for each latent variable.** We simulated a dataset containing multiple underlying timescales, including a distinct high-frequency ground-truth latent variable (first column). Applying standard SCA requires rigid Gaussian pre-smoothing of the raw spiking data. Using a narrow 20 ms filter (second column) preserves the high-frequency dynamics but yields excessively noisy latent estimates. Conversely, applying a wider 50 ms filter (third column) denoises slower signals but completely obscures the high-frequency latent. Because pSCA automatically infers the optimal temporal filter for each latent variable directly from the data, it successfully recovers both fast and slow dynamics with high fidelity, circumventing the need for arbitrary pre-smoothing (fourth column).

A.3.2 pSCA vs. SCA with Gaussian smoothing

To illustrate the limitations of fixed pre-smoothing on single-trial data, we simulated an additional dataset containing a high-frequency ground-truth latent variable (Appendix Fig. A.3, bottom left). The original SCA framework requires the raw spiking data to be pre-smoothed with a fixed-width Gaussian filter (σ shown in figure), forcing an arbitrary temporal trade-off: a narrow filter preserves high-frequency dynamics but yields noisy latent estimates, while a wide filter successfully denoises slow timescales but causes SCA to fail to recover the high-frequency latent. Because pSCA automatically learns an optimal temporal filter for each latent variable independently, it bypasses this trade-off entirely.

A.3.3 pSCA ablation study

To isolate the contribution of each component of pSCA, we re-fit the model on the simulation data from Fig. 1 (five ground-truth latents, 50 trials, 15 Hz mean rate; A.3.1) with individual pSCA features removed while deliberately over-specifying the model with six latent dimensions. In Fig. 1D we report two measures of how well each model recovered the ground-truth **(1)** the mean cross-correlation between aligned inferred latents and the ground truth and **(2)** the variance of the least-correlated (sixth) dimension as a fraction of total latent variance, which should be equal to zero when the model recovers the true dimensionality of the simulated data. The full pSCA model performed best. Removing the orthogonality penalty cost little correlation of the 5 true latents, but caused the model to populate a spurious sixth dimension (elaborated on below); removing sparsity degraded temporal demixing; and removing coordinated dropout had the largest single effect, with the convolutional filters collapsing to delta functions rather than learning temporal smoothing. We also show demixing performance using only a Poisson autoencoder (AE), and a Poisson AE with a sparsity penalty. Neither of these models succeed in demixing the latents, further highlighting the necessity of each of pSCA's features. We touch on the connection between the Poisson AE with sparsity in §A.4.6. A qualitative depiction of each feature's contribution to finding the ground truth can be seen in Fig. A.4.

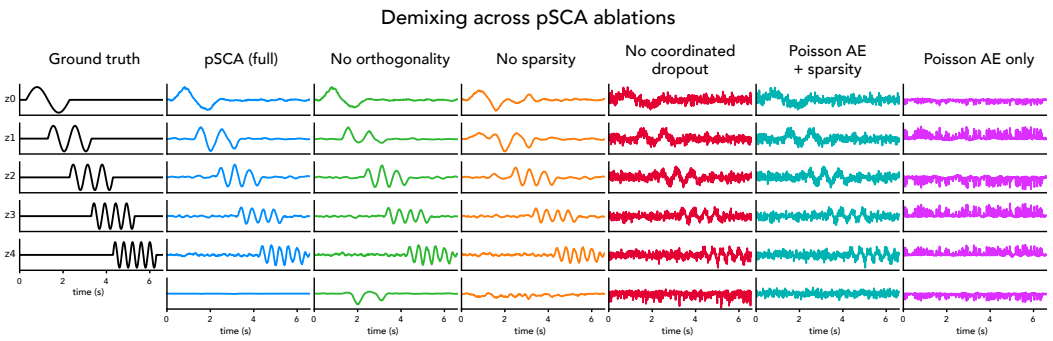

Figure A.4: **Qualitative effect of ablating individual pSCA components on latent recovery.** pSCA was fit to the Fig. 1 simulation (five ground-truth latents, 50 trials, 15 Hz) with six latent dimensions. Columns show the ground truth and inferred latents from a randomly selected trial for the full model and each ablation, aligned to the ground truth; rows are the five matched dimensions plus the unmatched sixth. The full model recovers all five latents and leaves the sixth empty. Removing orthogonality populates a spurious sixth dimension; removing sparsity degrades temporal demixing; removing coordinated dropout is most damaging; ablating all three leaves latents largely uncorrelated with the ground truth.

We would like to draw specific attention to the model fit only ablating orthogonality. We have seen empirically that without orthogonality, pSCA can find latent representations with the same level of ℓ_1 sparsity, but that *split* ground-truth latent processes into multiple dimensions. This can be seen in rows 2 and 6 of column 3. This highlights the need for orthogonality in the pSCA architecture, in order to prevent false positive detection of redundant, artificially fragmented latent neural processes.

To additionally test the contribution of the Poisson reconstruction loss, we generated simulated data at a reduced rate of 2.5 Hz, where the Gaussian approximation to the Poisson likelihood is poor. pSCA recovered the ground-truth latents moderately well with a correlation of 0.790 while an otherwise identical Gaussian model completely collapsed with correlation = 0.072.

A.3.4 pSCA DATA EFFICIENCY ANALYSIS

To test how well pSCA recovers ground-truth latent variables under realistic data constraints, we varied the number of both neurons (Fig. 1E left) and trials (Fig. 1E right) and measured the matched correlation distance between the inferred latents and the ground-truth. This was computed by first aligning the inferred and ground-truth latents via their cross-covariance and linear sum assignment and then calculating the squared difference of the resulting matched cross-correlation matrix from the identity matrix (where a distance of 0 indicates perfect recovery). We used 100 neurons when sweeping over trials and 100 trials when sweeping over neurons.

A.3.5 pSCA DEMIXING PERFORMANCE ACROSS ASSUMPTION VIOLATIONS

In order to test how robust pSCA is to datasets that violate its core assumptions, we created modified versions of the original simulated dataset shown in Fig 1.B (detailed in §A.3.1 such that each modified version of the dataset included conditions that break pSCA's assumptions. We then trained pSCA on these datasets to test pSCA's ability to recover the ground-truth latent variables.

The individual conditions considered were: **(1)** the ground-truth nonlinearity mapping latents to firing-rates is a softplus rather an exponential (what pSCA assumes), **(2)** the spiking activity is over-dispersed with an average Fano factor of 1.5, **(3)** the spiking activity is under-dispersed with a Fano factor of 0.8, **(4)** one of the ground-truth latents is a slow ramping signal (non-sparse), and **(5)** a combined dataset integrating conditions 1, 2, and 4. Additionally, all simulated datasets (including the original in Fig. 1) slightly differ from pSCA's orthogonality assumption by using a non-orthogonal, random mapping from the ground-truth latent factors to the observed rates. As in the Fig. 1 simulations, there were 5 ground-truth latent factors, but we intentionally over-specified the model by fitting pSCA with 6 dimensions to evaluate its ability to infer the true underlying dimensionality.

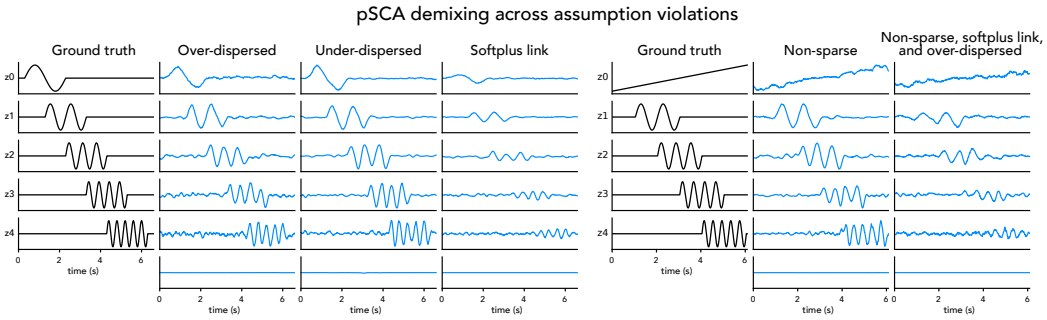

Figure A.5: **pSCA finds the ground-truth latents even when model assumptions are violated.** We generated simulated datasets that violated pSCA's assumptions of sparse latents, Poisson distributed data, and using an exponential link function. Each column represents either the ground-truth latents or the pSCA-inferred latents on a random single trial inferred using simulated data violated one or more of pSCA's assumptions. In each case, pSCA successfully recovered the ground-truth, even when multiple violations were present (last column).

We used two metrics to assess performance: **(1)** to assess accuracy of the inferred latents, is the cross-correlation between the aligned pSCA-inferred latents and the ground-truth latents, averaged across dimensions; and **(2)** to assess whether extra dimensions were inaccurately learned, was the variance of the dimension least-correlated with the ground-truth latents. Because there were only 5 ground-truth latents, a successful recovery by the over-specified model means this 6th (least-correlated) dimension should have a variance very close to zero. The results indicate that pSCA is broadly robust to deviations from its core assumptions. For all individual violations (columns 2-5), the correlation with the ground-truth latents remained above 0.91, and only dropped to a still-strong 0.833 when the dataset simultaneously violated 3 of the assumptions at once. In every scenario tested, the variance of the 6th dimension was 0.0, confirming the model successfully recovered the correct ground-truth dimensionality despite assumption violations. A qualitative depiction of pSCA's performance under these conditions is shown in Fig. A.5.

A.4 Macaque motor cortex during hand-cycling

Data formatting and preprocessing: We obtained the cycling dataset from Zenodo (DOI 10.5281/zenodo14769761) and binned the continuous neural activity into 10 ms intervals. To ensure behavioral consistency, we excluded trials exhibiting significant electromyography (EMG) activity prior to the "go" cue. This filtering resulted in a final dataset of 263 valid trials, downselected from the original 293. Because the trials naturally varied in duration depending on the monkey's execution speed and the specific number of cycles required, the resulting sequence lengths were unequal.

To accommodate the trial-to-trial temporal variability, we applied model-specific preprocessing. For pSCA and LFADS, we zero-padded the shorter trials to match the duration of the longest trial in the dataset. GPFA required no structural preprocessing as it naturally accommodates variable-length trials (the necessary square-root transform was handled internally by the `elephant` package as previously noted). Finally, for standard SCA with presmoothing, we smoothed each trial's neural activity with a 30 ms Gaussian kernel and concatenated the sequences across all trials to form a single continuous time series, circumventing the need for padding. For all analyses subsequent to model fitting, the latents were only evaluated using the non-padded portions of the trials.

A.4.1 Latent Sorting and Feature Selectivity for Cycling Task

In order to compare the unsupervised demixing capabilities across model architectures in panel C and panel D of Fig. 2 in the main text, we aligned the inferred latent trajectories to behavioral epochs and algorithmically identified dimensions corresponding to specific task features.

Trial Selection and Kinematic Alignment: Continuous angular velocity measurements were used to parse the temporal structure of each trial. We automatically identified movement onset as the

time point where angular velocity exceeded baseline resting levels (mean plus 10 standard deviations of the pre-cue period). Similarly, movement termination (stopping) was identified when angular velocity returned to baseline near the end of the trial. Trials in which movement was initiated prior to the "go" cue were excluded from the analysis. To ensure a fair comparison across models, latent trajectories from GPFA, LFADS, and presmoothed SCA were truncated by 150 ms at the beginning and end to match the inherent temporal truncation of the pSCA convolutional architecture.

Functional Latent Sorting: We compared how selective the latents produced by each method were for specific task-related neural processes. To do this, we utilized a variance-based sorting algorithm to identify the specific latent dimensions that most strongly correlated with distinct behavioral epochs (Preparation, Forward cycling, Backward cycling, and Stopping). For each model, we computed an "occupancy" metric for each dimension based on its mean squared activity within specific temporal windows.

- **Preparation:** Let $P_{k,i}$ and $M_{k,i}$ represent the mean squared activity of latent dimension k on trial i during the preparation ($\mathcal{T}_{\text{prep}}$) and movement ($\mathcal{T}_{\text{move}}$) windows, respectively. The preparation-selectivity score for dimension k, averaged across N valid trials, is defined as:

$$S_k^{\text{prep}} = \frac{1}{N} \sum_{i=1}^{N} \frac{P_{k,i}}{P_{k,i} + M_{k,i}}$$

  The representative preparation dimension shown in Fig.2C row 1 was then selected as the dimension maximizing this score: $\arg\max_k(S_k^{\text{prep}})$. For Fig.2D, we recomputed this ratio for each latent dimension over multiple bootstrapped samplings of trials, and then plotted the average and standard deviation across both dimensions and bootstrapped samples to identify which method produces the most preparation-selective latent dimensions.

- **Directionality (Forward/Backward):** To identify direction-specific dimensions, we isolated the active movement window ($\mathcal{T}_{\text{move}}$). Because a single trial consists of exclusively forward or backward cycling, we first computed the trial-averaged mean squared activity of latent dimension k across all forward-cycling trials (F_k) and backward-cycling trials (B_k). The forward- and backward-selectivity scores are defined as:

$$S_k^{\text{fwd}} = \frac{F_k}{F_k + B_k} \quad \text{and} \quad S_k^{\text{bwd}} = \frac{B_k}{F_k + B_k}$$

  The representative dimensions shown in Fig.2C rows 2 and 3 were selected by maximizing these scores: $\arg\max_k(S_k^{\text{fwd}})$ and $\arg\max_k(S_k^{\text{bwd}})$. As with preparation, these scores were recomputed over bootstrapped trial samples to identify the most direction-selective models for Fig.2D.

- **Stopping:** We identified the stopping dimension by comparing activity at the end of the trial to activity during steady-state movement. Let E_k and C_k represent the trial-averaged mean squared activity of latent dimension k during a window immediately preceding movement termination ($\mathcal{T}_{\text{stop}}$) and a steady-state movement window ($\mathcal{T}_{\text{steady}}$), respectively. The stopping-selectivity score is defined as:

$$S_k^{\text{stop}} = \frac{E_k}{E_k + C_k}$$

  The representative stopping dimension shown in Fig.2C row 4 was selected as the dimension maximizing this score: $\arg\max_k(S_k^{\text{stop}})$. We similarly evaluated stopping-selectivity robustness by bootstrapping these scores across trial samples for Fig.2D, using the exact same sampling procedure described for preparation.

Visualization: For Fig. 2, we plotted the single-trial latent trajectories for one representative forward trial and one representative backward trial. Traces were aligned to movement onset ($t = 0$). To visualize the relative amplitude of the signals without vertical offset artifacts, each plotted latent dimension was mean-centered across the visualized trials. Fig. A.14 displays the complete set of inferred latent dimensions for each method using the same trial alignment and centering procedures.

A.4.2 BI-CROSS-VALIDATION ANALYSIS

To quantitatively evaluate predictive performance, we employed a 5-fold bi-cross-validation strategy across both neurons and trials. For each fold, we fit the models using 80% of the neural population and subsequently inferred the latent trajectories across all trials using only these training neurons. We then trained a Poisson generalized linear model (GLM) to map from these inferred latents to the activity of the held-out neurons using a training subset of trials (80%). The trained GLM was then tested on its ability to predict the activity of the held-out neurons on the held-out trials. Performance was quantified using the Poisson deviance fraction explained, or pseudo-R^2:

$$D^2 = 1 - \frac{\text{NLL}_{\text{model}} - \text{NLL}_{\text{sat}}}{\text{NLL}_{\text{null}} - \text{NLL}_{\text{sat}}}$$

$\text{NLL}_{\text{model}}$ is the Poisson negative log-likelihood of the trained model (pSCA, GPFA, LFADS, SCA). NLL_{sat} is the Poisson negative log-likelihood assuming perfect predictions. NLL_{null} is the Poisson negative log-likelihood for a model that predicts the mean firing-rate for each neuron. For the results in Fig. 2E, we re-computed the pseudo-R^2 over bootstrapped samplings across trials and plotted the resulting standard deviations as error bars for each method.

A.4.3 SINGLE-TRIAL DIFFERENCES IN LATENT PREPARATORY ACTIVITY

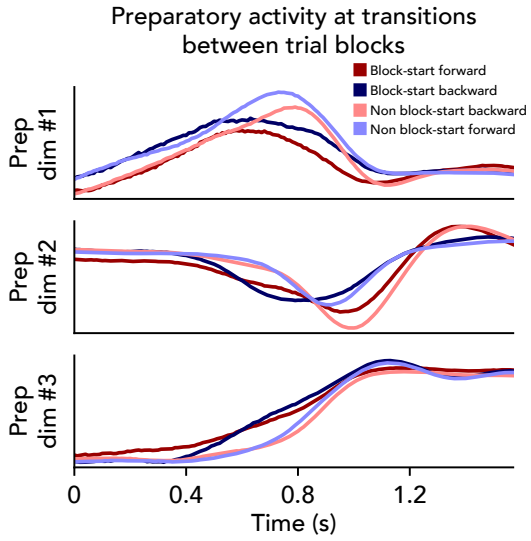

Figure A.6: **Differences in preparatory activity at the single-trial level:** We computed the average latent activity for the top 3 preparatory dimensions recovered by pSCA within the preparation window. For instance, the dark red line represents the preparatory-period average of the first dimension across the very first trial of every forward-cycling block, whereas the corresponding lighter line illustrates the average across all non-starting trials across the trial blocks. If we interpret the magnitude of the preparatory signals in rows 1 and 2 as reflective of "how prepared" the monkey was for the upcoming trial, this might indicate that the monkey was less prepared for the first trial within a forward block than it was for a subsequent trial within that same block.

To demonstrate the insights available from single-trial data, we investigated how the preparatory dimensions recovered by pSCA differ between the first trial of a block and the subsequent trials. We began by identifying the top three preparatory dimensions across all trials using the approach described in A.4.1. Next, we averaged the activity within the pre-movement preparatory window, grouping trials by their cycling direction and whether they were the initial trial of a block. Results are shown in Fig. A.6. The preparatory activity at the start of a new block (e.g. switching from forward trials to backward trials) was attenuated when compared to trial that did not occur at block switches.

A.4.4 SINGLE-TRIAL STOPPING ACTIVITY

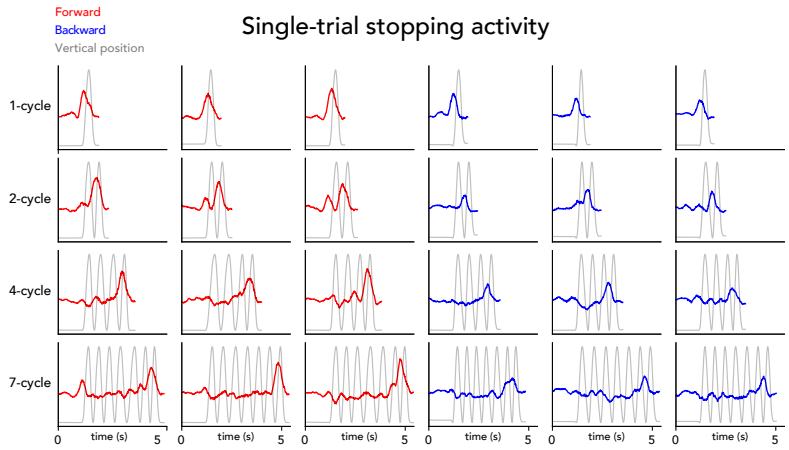

Figure A.7: **Single-trial stopping activity**. Latent trajectories (red: forward, blue: backward) plotted against vertical position (gray) across varying cycle lengths. Single-trial activity reveals a latent factor that activates during the cycle preceding the final stop.

pSCA found latent dimensions preferentially active prior to the final cycle on a single-trial basis. The latent shown in Fig. A.7 begins to increase its activity during the previous cycle (at least a few hundred ms before the final cycle start), more consistent with a preparatory role. Additionally, the exact timing of this latent varies relative to the actual movement on a trial-to-trial basis (compare columns) - again more consistent with a preparatory role. Beyond helping to suggest a more preparatory (versus execution) role, examining these un-averaged, single-trial fluctuations also provides a window into internal cognitive states, exposing occasional premature stopping preparation potentially caused by misestimation of the distance to the target.

A.4.5 RELATIONSHIP BETWEEN PSCA'S LATENTS AND PAST NEURAL PROCESSES FOUND IN CYCLING

Past work discovering distinct neural processes on trial-averaged cycling data found the four underlying categories of distinct neural processes to be **1.)** preparation, **2.)** movement prior to the final cycle (there referred to as 'steady state'), **3.)** stopping, and **4.)** posture. Here, pSCA recovered latents reflecting the first three processes, but not the fourth. The reason for this relates to differences in task design between the two cycling datasets analyzed. In the dataset analyzed in previous work, each trial began with the monkey holding the manipulandum at different starting positions (top or bottom), which resulted in different neural activity patterns corresponding to the different postures reflected in the inferred latent variables. In the single-trial dataset we analyzed here, there was not a variable start position, and therefore the latents related to variable postures were not discovered.

A.4.6 PSCA'S RELATIONSHIP TO DICTIONARY LEARNING

The notion of sparsity in both neuroscience and machine learning has a long history (Olshausen & Field, 1996; Rubinstein et al., 2010). By decomposing neural activity into the product of two matrices, and encouraging one to be sparse, both SCA (Zimnik et al., 2026) and pSCA can be considered types of what is called dictionary learning. While dictionary learning is a broad literature that has produced many useful tools for neuroscience (such as for extracting neural activity from calcium-imaging data (Giovannucci et al., 2017; Diego et al., 2013; Charles et al., 2022) or for spike sorting (Song et al., 2018)), it had not typically been used for finding latent factors of neural activity prior to SCA. Furthermore, pSCA includes a unique combination of features that distinguish it from other forms of dictionary learning and make it particularly useful for finding denoised and temporally dissociated latent factors.

Broadly, dictionary learning seeks to decompose a data matrix $X \in \mathbb{R}^{d \times n}$ into a dictionary matrix $D \in \mathbb{R}^{d \times k}$ and a sparse coefficient matrix $A \in \mathbb{R}^{k \times n}$, such that $X \approx DA$. This is typically



Figure A.8: **Latents learned by a Poisson autoencoder trained with a sparsity penalty on the latents.** To highlight the importance of the specific combination of pSCA's features on a real dataset, we fit a Poisson autoencoder with a sparsity penalty on the latent time-series to the single-trial cycling data. The raw latent time-series produced by this model are extremely noisy, making them difficult to interpret. Applying a modest amount of smoothing to the latents using a Gaussian filter does not substantially improve demixing performance. The failure of this model highlights that the Poisson loss and sparsity penalty are not enough to get strong demixing performance. Much better performance is achieved by using *all* of pSCA's features, as seen in Fig. A.10.

formalized as an optimization problem that minimizes a reconstruction error alongside a sparsity penalty on the coefficients:

$$\min_{D,A} \|X - DA\|_F^2 + \lambda_{\text{sparse}} \|A\|_1$$

The exact loss function and penalty can vary depending on the application. While these methods are effective at finding efficient representations of signals, prior methods are not suited to be directly applied to complex neural time-series data. As some examples (detailed more below), past dictionary learning variants are either typically optimized for Gaussian-distributed data, finding non-orthogonal coefficients, finding orthogonal coefficients with complete dictionaries ($k \gg n$), have denoising strategies suited only for specific applications (e.g. low light imaging), or some combination of these.

The key *combination* of features distinguishing pSCA from broader dictionary learning approaches are: **(1)** it handles the stochastic, Poisson-distributed nature of neural spiking; **(2)** it encourages inferred latent factors to reside in orthogonal subspaces; **(3)** the sparsity penalty is applied to the latent time-series, not the coefficients; **(4)** the combination of convolutional filters and coordinated dropout is specifically designed for denoising neural time-series; and **(5)** the use of an adaptive sparsity penalty, removing the need for time-consuming sweeps over sparsity penalty strengths (λ above).

While there are methods in the dictionary learning literature designed with some subset of these features, there are no methods that include all of them in a unified architecture. For instance, while some techniques are optimized for Poisson observations (Giryes & Elad, 2014), they lack orthogonal dictionary components and rely on averaging techniques tailored to static image patches, which translate poorly to neural time-series data. Conversely, methods that do find orthogonal dictionaries either fail to handle Poisson noise, fall under "complete dictionary learning" (which requires that $k = n$), or both (Sun et al., 2015; Zhai et al., 2020; Bao et al., 2013). More recent approaches which have been applied to neural data (Mudrik et al., 2024) require segmenting recording sessions into trials with predefined states. As discussed in the main text, this is impractical for many naturalistic datasets, which are rarely easily parsed into trials, let alone assigned discrete states. Moreover, none of the approaches mentioned above make use of an adaptive sparsity penalty that automatically selects the value of λ_{sparse}.

To further illustrate the benefits of including *all* of pSCA's features over existing approaches that might include a subset of those features, we trained a Poisson autoencoder (AE) with a sparsity penalty, which is similar to the Poisson dictionary learning technique presented in Giryes & Elad (2014). This model fails to demix the ground-truth latents, producing extremely noisy estimates. In our simulations, this is shown quantitatively in row 5 of Fig. 1D and qualitatively in column 6 in Fig. A.4. We also show how a more standard Poisson dictionary learning approach, or one with posthoc smoothing, fails to achieve demixed and denoised latents like pSCA does on our cycling dataset in Fig. A.8. Thus, by combining temporal convolutions, Poisson likelihood, an orthogonality penalty, the coordinated dropout denoising procedure, and an adaptive sparsity penalty into a single unified framework, pSCA is uniquely equipped, relative to other dictionary learning approaches, to simultaneously denoise and demix single-trial population activity.

A.4.7 AUTO-SPARSITY APPLIED TO THE CYCLING DATASET

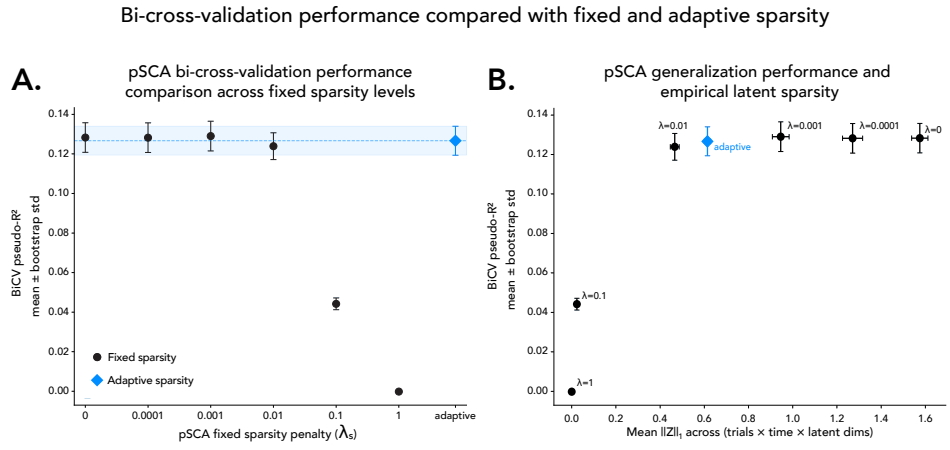

Figure A.9: **(A)** Generalization performance (BiCV pseudo-R^2) across fixed sparsity penalties (λ_s) compared to the adaptive method. The adaptive penalty generalizes as well as the best fixed settings, while bypassing the need for an extensive sweep over values of λ_s, reducing the required model fits from 6 to 1 per cross-validation fold. **(B)** The trade-off between predictive performance and empirical latent sparsity ($||\mathbf{Z}||_1$). The adaptive method automatically isolates the optimal regime, discovering a highly sparse representation without suffering the performance collapse observed at rigid penalties $\lambda_s \geq 0.1$. All analyses were performed on the cycling dataset.

In this work, we consider an optimal latent representation as one that maximizes temporal sparsity without sacrificing the ability to accurately predict held-out data. Traditionally, achieving this balance requires a hyperparameter sweep over the sparsity coefficient λ_s. For example, evaluating 6 fixed values of λ_s under the 5-fold cross-validation scheme described in §A.4.2 requires fitting 30 separate models to identify that $\lambda_s = 0.01$ yields the sparsest valid representation.The adaptive sparsity penalty circumvents this computational burden by dynamically adjusting the regularization pressure during training. By automatically seeking the sparsest representation that preserves generalization, the adaptive method requires fitting only 5 total models (one per fold). As shown in Fig. A.9A, the adaptive penalty achieves a BiCV pseudo-R^2 on par with the best-performing fixed sparsity levels. Furthermore, when evaluating the trade-off between predictive performance and empirical latent sparsity (Fig. A.9B), the adaptive method successfully isolates the optimal regime, yielding a highly sparse representation without the collapse in performance observed at fixed penalties $\lambda_s \geq 0.1$.

A.4.8 COUPLED COORDINATED DROPOUT

Standard coordinated dropout can still sometimes lead pSCA to produce uninterpretable, "spiky" latents from highly correlated neurons (e.g., due to electrode array crosstalk) by using one neuron to predict its highly-correlated partner. While it is still advantageous that pSCA isolates these spiky latents into specific dimensions rather than mixing them with others, it nonetheless could be sim-

plify interpretation by removing them entirely. To address this, we introduce a new extension of coordinated dropout where user-specified correlated pairs share identical temporal dropout masks, strictly preventing the model from reconstructing masked spikes using an unmasked partner. This modification successfully eliminates spiky latents (see appendix Fig. A.14 vs. Fig. A.10). We provide this feature as an option in our code package, however, all models presented in this work (except for the one in Fig. A.10) were fit with it disabled.

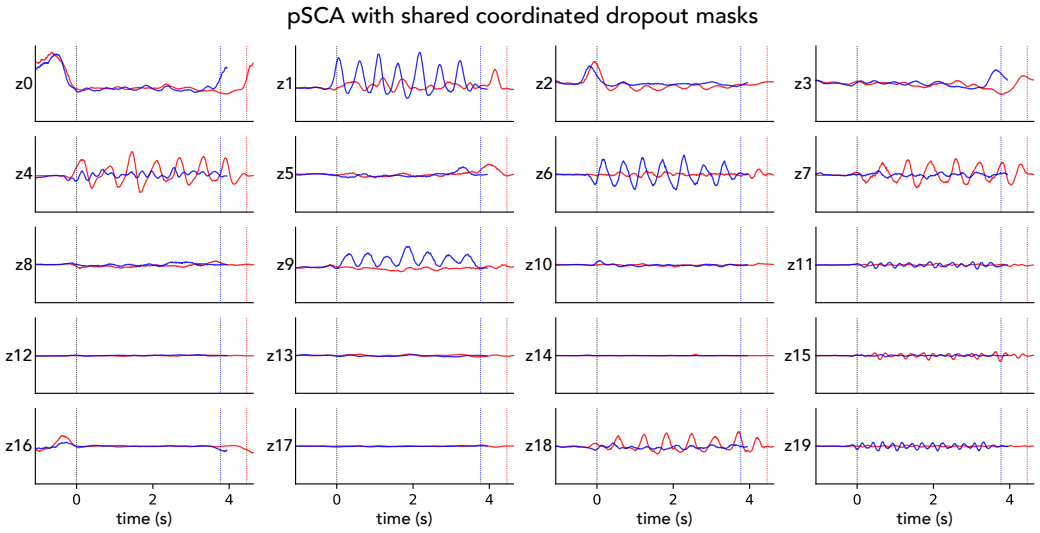

Figure A.10: **Removing "spiky" latents learned by pSCA**. By forcing pSCA to use the same temporal mask for highly correlated neurons, we can remove the "spiky" latents learned by pSCA observed in appendix Fig. A.14.

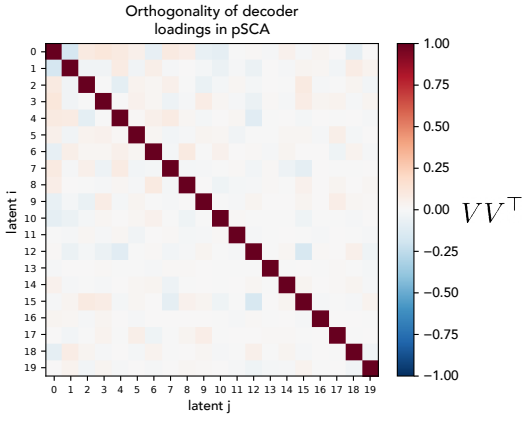

Figure A.11: **Orthogonality of pSCA's decoder loadings**. By computing $VV^\top$ with a trained pSCA model, we can evaluate the orthogonality of the learned latent representations. For the cycling dataset, pSCA learned an approximately orthogonal latent subspace for all latent representations.

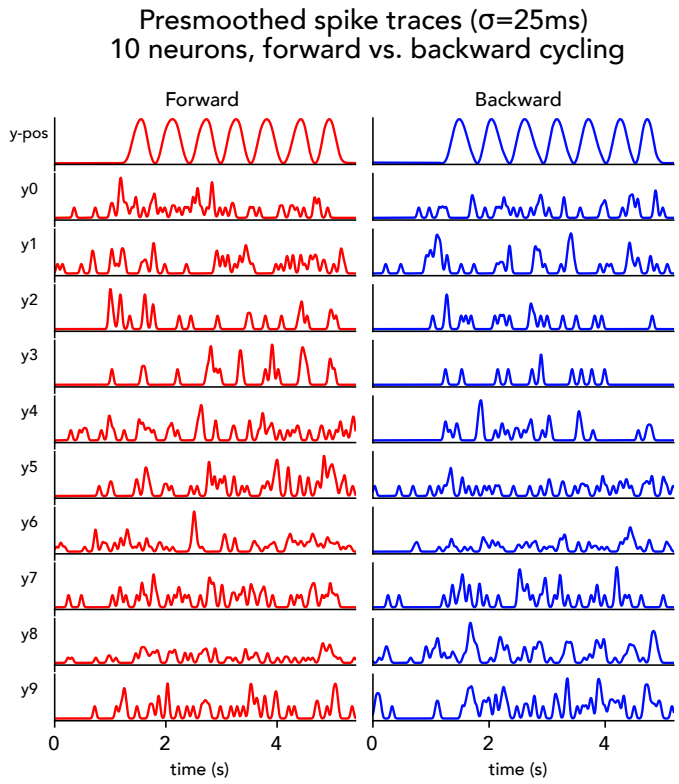

Figure A.12: **Smoothed spiking activity of 10 neurons in the primary motor cortex.** On the left is the activity of those neurons on a forward cycling trial, on the right is the activity of the same neurons on a backwards trial. Single-trial smoothed neural activity is not easily attributed to the underlying functional building blocks required to complete a task

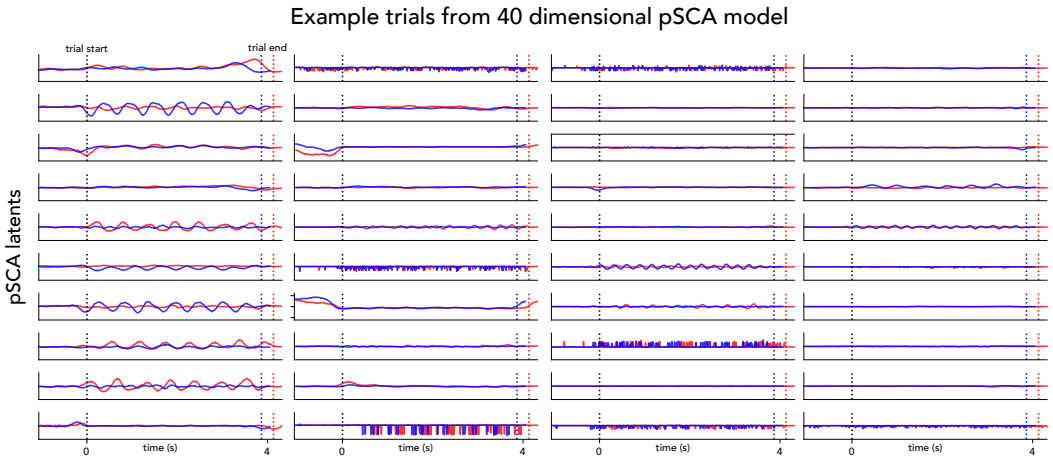

Figure A.13: **pSCA fit with 40 latent dimensions**: The results reported in the main text concerning the interpretability of the latents found by pSCA are robust to the number of dimensions selected. That is, pSCA still recovers dimensions uniquely active during forwards cycling, backwards cycling, movement preparation, and stopping. We note that we also include latents from pSCA fit with coupled coordinated dropout to remove the spiky latents (see Fig. A.10).

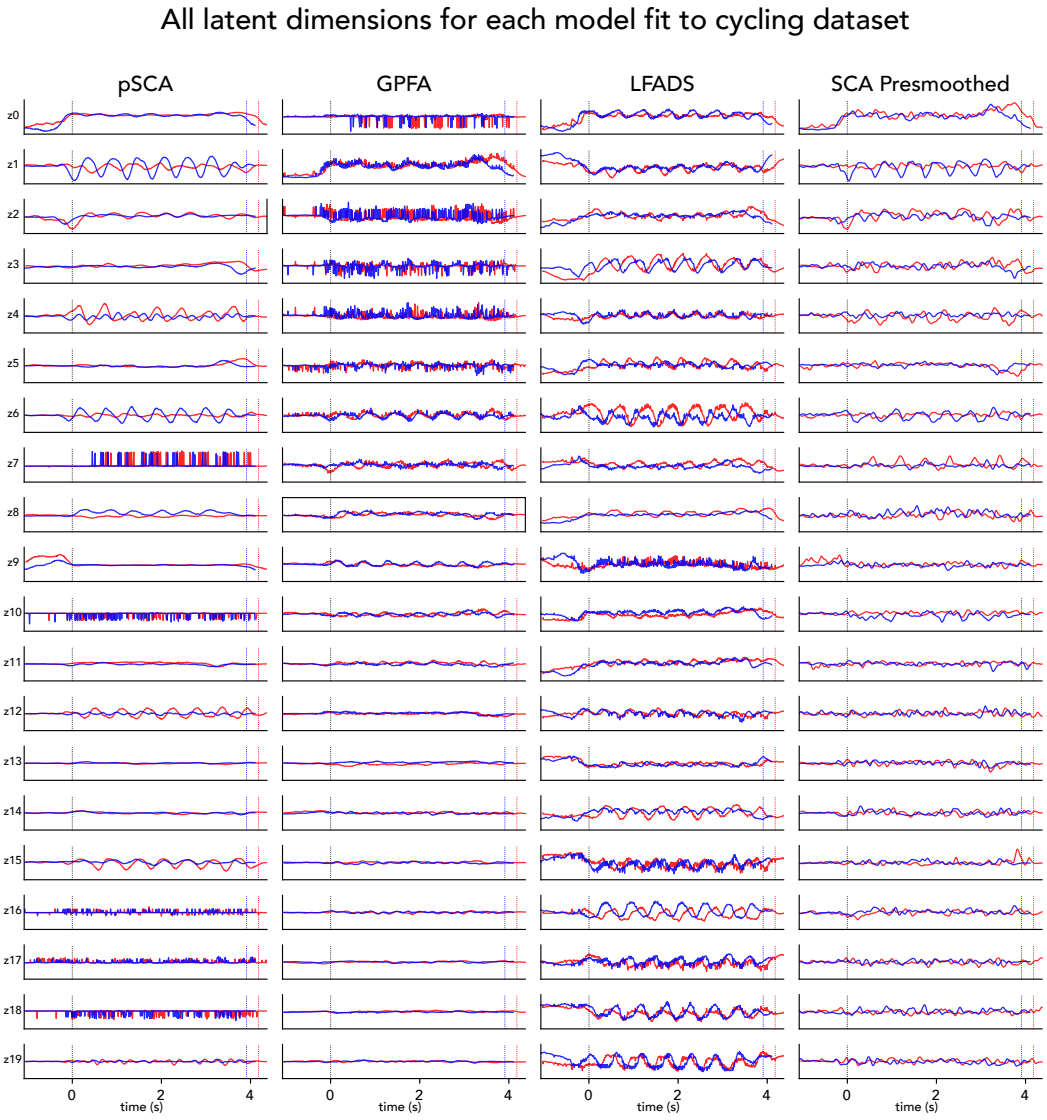

Figure A.14: **Latents inferred across models:** All 20 latents shown across models, illustrating the power of pSCA to demix distinct neural processes temporally, relative to other single-trial approaches. The largest magnitude latent for each method is shown in the first row and plotted on its own y-axis. All subsequent latents within a method share a common axis. We note that we also include latents from pSCA fit with coupled coordinated dropout to remove the spiky latents (see Fig. A.10).

A.5 MOUSE MOTOR CORTEX AND STRIATUM IN NATURALISTIC ARENA PARADIGM

A.5.1 ARENA DATASET PREPROCESSING

Continuous 1 ms-resolution single-unit firing rates (from both forelimb motor cortex (fM1) and dorsolateral striatum (DLS)), electromyography (EMG), and behavioral labels from a supervised classifier trained to predict human labeled behaviors from EMG signals were extracted from the raw recording. Contiguous behavioral epochs (bouts) lasting $\geq$ 500 ms were isolated and padded with 800 ms of the surrounding time-points.

The neural activity was binned at 20 ms. Within each bin, EMG was averaged and the behavioral label was taken as the mode within each 20 ms interval. Neurons exhibiting zero spikes across the entirety of the recording were removed from both populations. The result was 81 neurons in fM1, 148 neurons in DLS, 8 EMGs, for a total of approximately 36 minutes of data.

From this processed data, we generated two parallel datasets: a "full" dataset comprising the intact, variable-length padded bouts, and a "training" dataset where these bouts were segmented into uniform, non-overlapping windows defined by the duration of the shortest bout. The "training" dataset was used to train both pSCA and GPFA. Natural class imbalances across behavior types were preserved in both datasets without resampling.

A.5.2 EVALUATING BEHAVIORAL SPECIFICITY OF LATENT REPRESENTATIONS

For each latent dimension we compute its per-behavior power: the mean squared latent activity across all bouts of that behavior. The resulting $N_{\text{behaviors}} \times K$ matrix, is then column normalized, providing a probability distribution of each latent over all behaviors. To order the latents from most-specific to least-specific, we compute the Shannon entropy of each column and sort latents by ascending entropy, so columns on the left of the heatmap are dominated by a single behavior (highly selective) while columns on the right are spread across many behaviors (mixed).

We used this method to generate heatmaps describing the behavioral selectivity of the latents produced by both pSCA and GPFA, using data from both fM1, shown in Fig. 3B, and striatum, shown in Fig. A.17.

A.5.3 EVALUATING CORTICAL AND STRIATAL LATENTS FOR MUSCLE REPRESENTATIONS

To identify "EMG-related" pSCA components for both the fM1 and DLS regions, we evaluated the relationship between the inferred latent trajectories using pSCA and muscle activity. For each (latent, EMG-channel) pair within a behavioral bout, we computed the small-lag (± 5 bins) cross-correlogram of the mean-centered traces and extracted the squared peak. After aggregating these scores across all bouts and behaviors, we ranked the latents by their maximum correlation to any EMG channel and selected the top three factors from each brain region. Figure 3C visualizes the selected components for both brain regions during a characteristic bout for each behavior (columns). The plot displays the top fM1 latents in the 4-6 the top DLS latents in the subsequent rows 1-3, and two exemplary EMG traces in the bottom two rows, all plotted in the same exemplary behavioral bout. To facilitate direct comparison, column widths are scaled proportionally to bout duration, dotted vertical lines denote the 25-bin pre- and post-bout padding, and each row shares a common y-axis scale across all behavioral columns.

A.5.4 EVALUATING CORTICAL AND STRIATAL LATENTS FOR BEHAVIORAL-TRANSITION SELECTIVITY

To identify dimensions preferentially active during behavioral transitions, we utilized the known 500 ms (25-bin) padding boundaries marking behavior onset and offset. For the pSCA latent trajectory within each bout, we computed the mean squared activity within narrow ($\pm$ 2 bin; 40 ms windows centered on these boundaries. Latents were ranked by this "transition power" (summed across both onset and offset boundaries and averaged across all bouts), and the top four were selected as the primary transition latents. To visualize the transition-related dynamics for each brain region, we selected one representative bout per behavior, for each brain region. The selection was made by identifying the bouts on which the transition-related latent activity was maximal, computed as the ratio of the latent activity within the transition windows to the overall activity throughout the entire

bout. The resulting figure plots the top four transition-related latents across the 10 behaviors, with panel widths scaled proportionally to bout duration, dotted vertical lines denoting the transition boundaries, and globally shared y-axes for each latent.

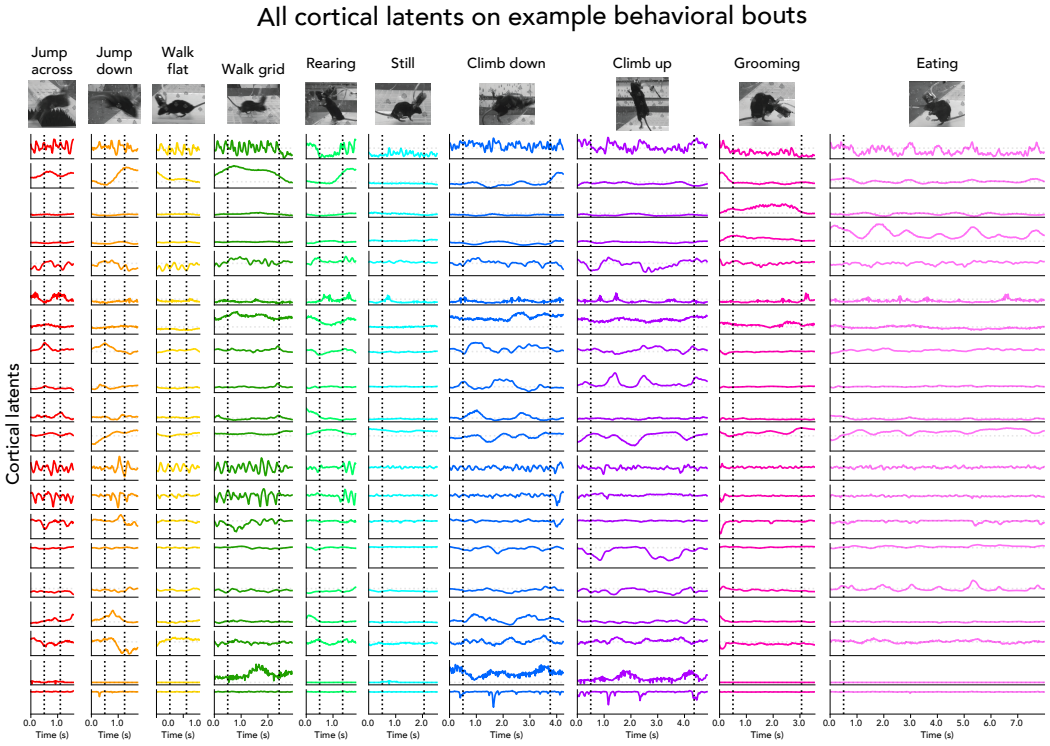

Figure A.15: **Latents inferred using pSCA on fM1:** All 20 latent dimensions from the fit pSCA model when applied to data gathered from fM1 during an unconstrained behavioral paradigm that promotes a diverse set of behaviors such as jumping, rearing, and climbing. Each row represents a different latent dimension and each column represents a different exemplary bout of the behavior shown at the top of the column. Dotted lines represent the onset and offset of the behavior indicated at the top of the column. In fM1, pSCA finds latent dimensions that vary in their information content, partially capturing different patterns of muscle activity, and differentially capturing neural activity patterns unique to and shared across different behaviors.

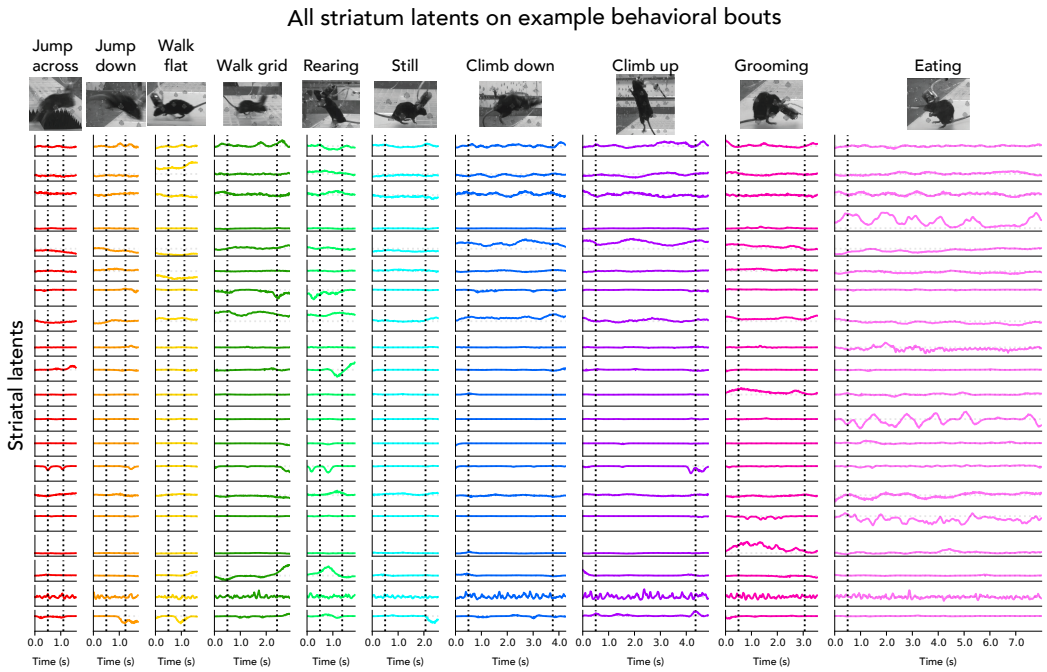

Figure A.16: **Full set of latents inferred using pSCA on DLS:** All 20 latent dimensions from the fit pSCA model when applied to data gathered from DLS during an unconstrained behavioral paradigm that promotes a diverse set of behaviors such as jumping, rearing, and climbing. Each row represents a different latent dimension and each column represents a different exemplary bout of the behavior shown at the top of the column.

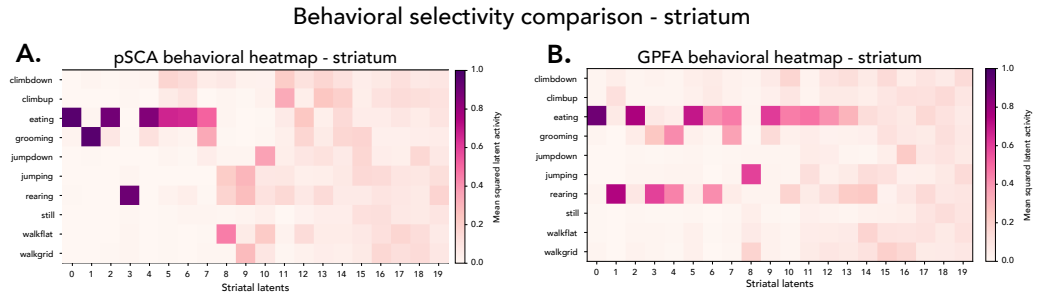

Figure A.17: **(A.)** Behavioral selectivity of pSCA's latent representations when fit to neural activity from striatum. pSCA finds multiple dimensions specific to eating, one dimension specific to grooming, one dimension specific to rearing, one shared between eating and grooming, and other dimensions with activity differentially shared across jumping, rearing, and different types of walking. **(B.)** GPFA fit to striatum behavioral heatmap: Behavioral selectivity of GPFA's latent representations when fit to neural activity from striatum. GPFA appears to find latents that are mostly active exclusively during eating, and rearing, missing shared structure found by pSCA for jumping and rearing, as well as different types of walking.