# Nonlinear, Scalable Sequential Component Analysis

**Arielle Rosinski**

Department of Engineering

University of Cambridge

This dissertation is submitted for the degree of
*Master of Philosophy* in
*Machine Learning and Machine Intelligence*

Pembroke College 15th of August, 2024

# Abstract

Neural computations such as those underlying motor control, perception, navigation, and decision-making have been proposed to be supported by a dynamical system defined in low-dimensional latent space. Dynamical systems typically produce state-space trajectories that are consistent with a lawful flow-field. The trajectories are thus temporally non-reversible, such that they do not have equal probability of occurring in reverse. Therefore, enhancing our understanding of cortical function could benefit from techniques that facilitate investigation of non-reversible structure in population-level data. This thesis presents sequential components analysis (SCA), a dimensionality reduction method seeking projections of the data that maximise non-reversibility. Specifically, SCA defines a second-order measure of non-reversibility, which is optimised in a stochastic gradient fashion. Importantly, the SCA optimisation objective can be restated to promote kernelisation of the algorithm. SCA and kernel SCA (kSCA) were validated on a range of synthetic datasets with known ground-truth non-reversible dynamics. Notably, kSCA, but not SCA, successfully retrieved non-reversible structure even when it was not linearly accessible. (k)SCA was then applied to recordings from the monkey motor (M1) and dorsal premotor (PMd) cortices. The resulting latents exhibited rotational dynamical structure, which appeared behaviourally relevant. Indeed, the (k)SCA-derived non-reversible components of neural data were more reflective of hand kinematics relative to principal components. Overall, the results highlight the value of non-reversibility as a learning goal for dynamic time series analysis, with potential applications extending beyond neuroscience.

# Contents

# Declaration of Originality

I, Arielle Rosinski of Pembroke College, being a candidate for the MPhil in Machine Learning and Machine Intelligence, hereby declare that this report and the work described in it are my own work, unaided except as may be specified below, and that the report does not contain material that has already been used to any substantial extent for a comparable purpose.

Arielle Rosinski, 15th of August, 2024

# Declaration of Resources

- **Code**: the code for this thesis was written in Python using libraries including Numpy (Harris et al., 2020), scikit-learn (Pedregosa et al., 2011), Jax (Bradbury et al., 2018), and Optax (DeepMind et al., 2020). It can be found at https://github.com/ArielleRosinski/SCA_project.
- **Datasets**: experiments in Section 4 relied on the `MC_Maze` dataset (DANDI 000128), consisting of motor cortex (M1) and dorsal premotor cortex (PMd) recordings. It was introduced as part of the Neural Latents Benchmark by Pei et al. (2021), and is available at https://neurallatents.github.io/datasets
- **Conceptualisation**: this thesis project builds upon the work of Virginia Rutten, Alberto Bernacchia, and Guillaume Hennequin, who devised a related, singular value decomposition-based form of the linear SCA algorithm (e.g., Rutten et al., 2020b). Additionally, these authors introduced the prototypes of the linear and kernel SCA algorithms used here.
- **Contribution**: Guillaume Hennequin provided the joint torques described more fully in Section 4.

# Word count

10,658

## Acknowledgements

I would like to express my gratitude to my supervisor Guillaume Hennequin. Guillaume has kindly accepted to take me on board, suggesting an area of research that is particularly exciting to me at the intersection between machine learning and neuroscience. He has taken the time to introduce me to various concepts, techniques, and papers, which have not only been useful for this project, but will also be beneficial for my research career ahead. I have appreciated his patience as well as his valuable help and guidance throughout this project.

Moreover, I am grateful to have had the privilege to be a part of the MPhil in Machine Learning and Machine Intelligence (MLMI) at the University of Cambridge this year. This has been a unique and enriching learning experience. I am also happy to have had the chance to form new friendships, both in MLMI and at Pembroke College.

Finally, I thank my family and friends for their precious support and encouragement during this journey.

# 1 Introduction

## 1.1 Motivation

The brain can be viewed as constituting a complex, high-dimensional coordinate system, whereby each coordinate axis represents the firing of a single neuron (Saxena & Cunningham, 2019). An increasingly popular view posits that the apparently high-dimensional biophysics of neural circuits implement a dynamical system confined to a lower-dimensional subspace, or manifold (Duncker & Sahani, 2021; Pandarinath et al., 2018b). This is compatible with the high recurrent connectivity of cortical networks (Mastrogiuseppe & Ostojic, 2018). Exploring the organisation of such population-level dynamics could contribute to enhancing our knowledge of how neuronal response patterns integrate into representations that ultimately support behaviour, a central goal in neuroscience (Gallego et al., 2017).

An important property of autonomous dynamical systems is that the state-space trajectories they generate are governed by a consistent flow field (Rutten et al., 2020a). This implies that the trajectories are temporally non-reversible, or equivalently, sequential. That is, they do not have equal probability of occurring in the reverse order. Accordingly, sequentiality appears to characterize population-level activity patterns across multiple neural systems. This is exemplified in the motor domain. Monkey primary motor (M1) and dorsal premotor (PMd) cortical activity during behaviours such as reaching and cycling exhibits low-dimensional sequential structure in the form of rotations of the neural state (Churchland et al., 2012; Lara et al., 2018; Russo et al., 2018; Rutten et al., 2020a). In the avian premotor nucleus HVC, which has been described as a potential analog of the mammalian motor cortex (Pfenning et al., 2014), neurons produce precisely timed, non-reversible sequences during birdsong vocalisations (Lynch et al., 2016; Picardo et al., 2016).

In the decision-making domain, accumulation of evidence, reflective of the formation of a decision, is represented by graded firing rates that ultimately result in a commitment to a choice (Gold & Shadlen, 2007). This can be demonstrated using a random-dot motion paradigm (Newsome et al., 1989), where subjects (e.g., monkeys, humans; Hanks et al., 2011; Roitman and Shadlen, 2002) are required to discriminate the direction of motion of dots, only a percentage of which move coherently towards one of two peripheral targets. Just before the decision is communicated by a saccadic eye movement, activity of neurons in the lateral intraparietal cortex (LIP) encodes the accumulation of sensory evidence (Roitman & Shadlen, 2002). This accumulation process unfolds until the neurons tuned to the chosen target reach a threshold firing rate. On the other hand, a decline in firing rate is observed when the alternative target is selected, consistent with mounting evidence against the target the neurons are tuned to (Roitman & Shadlen, 2002). Therefore, such accumulation preceding commitment is inherently non-reversible, since activity either increases or decreases without retracting.

In the spatial navigation domain, trajectories in the environment induce precisely ordered spike sequences of spatially-tuned cells (Drieu & Zugaro, 2019). These include place cells in the cornu ammonis 1 (CA1) region of the hippocampus, active when the animal traverses a specific location in the environment. Notably, place cells were found to be directional such that different heading directions elicit different response patterns (Acharya et al., 2016; Navratilova et al., 2012; Stachenfeld et al., 2017). Therefore, the statistical profile of hippocampal population activity is anticipated to be non-reversible even during back-and-forth navigation in a linear track environment. Moreover, the place cell firing sequences observed during wakefulness are also recapitulated during periods of rest, a phenomenon referred to as replay (Lee & Wilson, 2002; Nádasdy et al., 1999; Ólafsdóttir et al., 2015; Skaggs & McNaughton, 1996). Instances of pre-activation (i.e., preplay) of hippocampal cells, potentially encoding future planned trajectories, have also been reported (Ólafsdóttir et al., 2015).

## 1.2 Previous work and limitations

The above examples demonstrate the prevalence of non-reversibility in the brain. Consequently, understanding the computational algorithms driving cortical networks could possibly benefit from techniques compressing population-level recordings in a way that facilitates probing of sequentiality in data. Previous studies have already leveraged dimensionality reduction approaches to uncover latent trajectories that are shared across the population and behave in agreement with lawful flow fields. For instance, jPCA is a linear dimensionality reduction method that seeks projections (onto an orthonormal basis) capturing rotational structure in data (Churchland et al., 2012). While initially deployed to probe monkey motor cortex latent variables, jPCA-obtained rotations have since been studied in various brain areas and species (Kuzmina et al., 2024). However, the specific form of dynamical structure sought by jPCA (i.e., rotations wherever they occur) has been criticised (Lebedev et al., 2019). There is indeed no a priori reason to assume that purely rotational dynamics dominate neural data, and other types of dynamics (e.g., contractions, shear, expansions; Lara et al., 2018) might have been overlooked. Alternative dimensionality reduction algorithms alleviating these concerns have been introduced (Lara et al., 2018; Rutten et al., 2020a; see also Mackevicius et al., 2019, which directly extracts temporal sequences from high-dimensional data). These include GPFADS (Gaussian Process Factor Analysis with Dynamical Structure; Rutten et al., 2020a). GPFADS constructs a Gaussian Process kernel that promotes the extraction of latent factors with non-reversible temporal evolution. It is then incorporated in a Gaussian Process Factor Analysis (GPFA) framework to learn latent processes statistics. GPFADS successfully revealed non-reversible population-level dynamics in both synthetic and neuroscience datasets.

Whilst theoretically appealing, there are still potential limitations to GPFADS. First, even though the non-reversible GPFADS kernel admits a Kronecker factorization that helps increase scalability, it remains relatively difficult to scale the method to very large datasets. Yet, these are increasingly prevalent in neuroscience given the advent of powerful recording technologies with large volume coverage (Jun et al., 2017; Siegle et al., 2021). Moreover, GPFADS is restricted to the identification of 2-dimensional subspaces. Although individual planes can be combined (Rutten et al., 2020a), it would be useful to develop a more straightforward approach towards identifying higher-dimensional subspaces. This seems particularly necessary given that the dimensionality of neural computational manifolds often matches or exceeds the dimensionality of corresponding task-related variables, which can be greater than 2 (e.g., 3-dimensional location; Duncker and Sahani, 2021; Jeffery et al., 2015; see also Low et al., 2018; Nieh et al., 2021). Finally, GPFADS-extracted subspaces are linear. Facilitating identification of nonlinear subspaces might nonetheless be crucial in certain cases. As will be shown in this thesis, trajectories satisfying an intuitive notion of non-reversibility can sometimes be (spuriously) deemed fully reversible by linear methods. For instance, the inherent non-reversibility of pure rotations will be missed if these are generated in the polar coordinate system but are subsequently converted to the Cartesian domain. In addition to enhancing the robustness of the method, promoting nonlinear transformations might confer greater biological relevance. Nonlinearities are indeed ubiquitous in the brain (e.g., in the motor cortex; Paninski et al., 2004, in the hippocampus; Low et al., 2018; Nieh et al., 2021), alluding to the possibility that meaningful non-reversible trajectories that are not linearly accessible exist.

## 1.3 Thesis outline and contributions

This project builds on the prototype of sequential components analysis (SCA) introduced by Rutten et al. (2020b), which provides an alternative, more flexible approach to examining the spatio-temporal structure of sequentiality in data. SCA directly searches for the lower-dimensional (potentially non-linear) projection for which sequentiality is maximised. It does so by defining a measure of temporal non-reversibility, which serves as an objective to be optimised (e.g., in a stochastic gradient-based fashion). The dimensionality of the resulting SCA-defined subspaces is not limited to planes. Additionally, SCA can be readily kernelised to accommodate higher-dimensional (incl. infinite) feature spaces, thereby allowing for nonlinear dimensionality reduction. The mathematical foundations relevant to linear SCA and kernel SCA (kSCA) are described in further details in Sections 2.1 and 2.2, respectively. The method is then validated on a range of synthetic datasets with known ground-truth non-reversible structure in Section 3. Section 4 highlights the potential of SCA-obtained projections to address questions about neural representational and computational properties in the motor domain.

# 2 Linear and nonlinear sequential component analysis

## 2.1 Mathematical foundations of linear SCA

Consider a set of zero-mean spatio-temporal observations $\{X_k \in \mathbb{R}^{N\times T}\}_{k=1,\ldots,K}$, where $T$ denotes the number of time bins, $N$ the number of units, and $K$ the number of samples. For instance, in a neuroscience setting, $N$ could represent the recorded neurons and $K$ the individual trials, or conditions if trial-averaged data is used. $X_k$ will be deemed reversible if, and only if $p(X_{k_{ti}}, X_{k_{t'i'}}) = p(X_{k_{t'i}}, X_{k_{ti'}})$ for two different units $i, i'$ and time points $t, t'$. This definition is analogous to the concept of detailed balance, which applies to stationary Markov processes and implies that the probability of transitioning between any two states is invariant under time reversal, though it extends to non-stationary processes. An example of such time-reversibility is the statistical structure of noise. In contrast, for structured space-time trajectories, such as those observed in the motor cortex, LIP, and the hippocampus (see Introduction), the reversibility criterion is unlikely to be met. The trajectories are thus considered non-reversible, or sequential.

The organisation of sequentiality in data can be specifically probed by considering the space-time covariance matrix $C \in \mathbb{R}^{TN\times TN}$, given by

$$C = \mathbb{E}[\mathrm{vec}(X_k^\top)\mathrm{vec}(X_k^\top)^\top] \tag{1}$$

where vec is the vectorisation operator (vertically stacks the columns of $X_k$), and the expectation is taken over the spatio-temporal samples ($K$ in total). This yields $N \times N$ blocks of dimension $T \times T$, denoting the temporal covariance between pairs of units. On the diagonal, these would correspond to the temporal covariance of a given unit with itself i.e., its auto-covariance (e.g., $C_{11}$ in Figure 1). Off-diagonal blocks specify the cross covariance between pairs of units at different lags. $C$ can then be decomposed by adding or subtracting its time-transposed version $\sigma(C)$ i.e., the matrix obtained by separately transposing each $T \times T$ block

$$\begin{aligned} C^{(+)} &= C + \sigma(C) \\ C^{(-)} &= C - \sigma(C) \end{aligned} \tag{2}$$

This effectively replaces each $T\times T$ cross-covariance matrix by its symmetric $(\cdot^{(+)})$ or its skew-symmetric $(\cdot^{(-)})$ component (Figure 1). Notably, the structure of $C^{(+)}$ reflects that of a reversible process. Indeed, symmetric blocks in $C^{(+)}$ capture the even parts of the cross-correlograms between units. The opposite is true for $C^{(-)}$, which captures the odd parts of the cross-correlograms, and thus describes the spatio-temporal structure of sequentiality in the data at least up to second-order moments. The degree of sequentiality, or equivalently, the departure from reversibility, of a spatio-temporal process can then be quantified using

$$\zeta = \frac{\left\|C^{(-)}\right\|_{\mathrm{F}}^2}{\left\|C^{(+)}\right\|_{\mathrm{F}}^2} \tag{3}$$

with $\|\cdot\|_{\mathrm{F}}^2$ indicating the squared Frobenius norm, corresponding to the sum of squares of individual elements. Conveniently, this ratio is provably bounded between 0 (i.e., perfectly reversible process) and 1 (i.e., systematic ordering between units; Rutten et al., 2020a). Note already that, under the linear SCA metric, 1-dimensional dynamics will always be reversible, since $C^{(-)}$ is 0, which results from subtracting the transpose of symmetric matrices (see diagonals in Figure 1). This potentially undesirable property does not apply for kernel SCA, described in the next section.

In its linear form, SCA seeks a unitary matrix $U \in \mathbb{R}^{N\times d}$ that maximizes the sequentiality of the spatially-projected samples $\{Y_k = U^\top X_k\}$ (with space-time covariance $C_U$). The numerator in $\zeta$ (equation 3) can thus be treated as a sequentiality objective to be optimised, subject to orthonormality of $U$ (i.e., $U^\top U = I_d$). To this end, it can be re-expressed using a computationally efficient stochastic estimator,

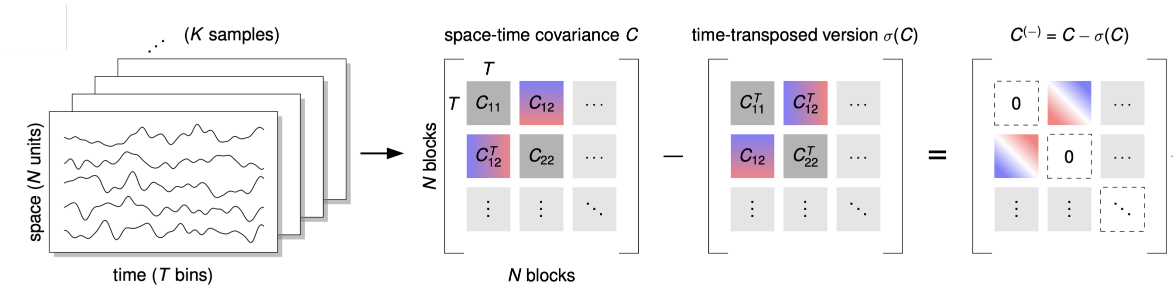

Figure 1: **Obtaining $C^{(-)}$ from a multivariate time series.** The space-time covariance of a spatio-temporal process (e.g., corresponding to the activity of $N$ neurons over $T$ time steps, for $K$ conditions) is computed. The matrix $\sigma(C)$, where each $T \times T$ block has been transposed, is subtracted, thereby retaining the skew-symmetric parts. Figure from Rutten et al. (2020b).

which will become particularly helpful for kernelisation purposes. First, Hutchinson's trace estimator identity is used to yield

$$\begin{aligned}\mathcal{S}(U) = \left\| C_U^{(-)} \right\|_{\mathrm{F}}^2 &= \mathrm{Tr}[C_U^{(-)} C_U^{(-)}] = \mathrm{Tr}\Big[C_U^{(-)} C_U^{(-)} \mathbb{E}_{v \sim \mathcal{N}(0,I)}[vv^\top]\Big] = \mathbb{E}\Big[\mathrm{Tr}[C_U^{(-)} C_U^{(-)} vv^\top]\Big] \\ &= \mathbb{E}\Big[\mathrm{Tr}[v^\top C_U^{(-)} C_U^{(-)} v]\Big] = \mathbb{E}\Big[v^\top C_U^{(-)} C_U^{(-)} v\Big] \qquad (4)\end{aligned}$$

which leverages the linear as well as the cyclic nature of the trace. Additionaly, the matrix-vector products $C_U^{(-)} v$ can be re-expressed as $\frac{1}{K}\sum_{k=1}^{K} \mathrm{vec}(\mathrm{Tr}[Y_k V^\top] Y_k - Y_k V^\top Y_k)$ (Rutten et al., 2020b). This results in

$$\begin{aligned}\mathcal{S}(U) &= \frac{1}{K^2} \sum_{k,k'} \mathbb{E}_{v \sim \mathcal{N}(0,I)} \left[ \mathrm{Tr}\left[ \left(\mathrm{Tr}[Y_k V^\top] Y_k - Y_k V^\top Y_k\right) \left(\mathrm{Tr}[Y_{k'} V^\top] Y_{k'} - Y_{k'} V^\top Y_{k'}\right)^\top \right] \right] \\ &= \frac{1}{K^2} \sum_{k,k'} \mathbb{E}\left[ \mathrm{Tr}\Big[ \mathrm{Tr}[Y_k V^\top] \mathrm{Tr}[Y_{k'} V^\top] Y_k Y_{k'}^\top - \mathrm{Tr}[Y_k V^\top] Y_k Y_{k'}^\top V Y_{k'}^\top - \mathrm{Tr}[Y_{k'} V^\top] Y_k V^\top Y_k Y_{k'}^\top + Y_k V^\top Y_k Y_{k'}^\top V Y_{k'}^\top \Big] \right] \\ &= \frac{2}{K^2} \sum_{k,k'} \Big[ \mathrm{Tr}^2[Y_k Y_{k'}^\top] - \mathrm{Tr}[Y_k Y_{k'}^\top Y_k Y_{k'}^\top] \Big] \qquad (5)\end{aligned}$$

where the sum is performed over all pairs $(k, k')$ of projected samples (see Appendix A for a full proof). The sequentiality objective $\mathcal{S}(U)$, written as an average, can be approximated by Monte-Carlo sampling using mini-batched pairs. SCA therefore lends itself to stochastic gradient-based optimisation of the subspace $U$. Note that optimising the numerator in $\zeta$, rather than the ratio, which no longer evaluates to an average, is necessary to help avoid biased estimates when computations are based on subsets of samples.

Additionally, an important consideration is that $U$ is parameterised as the Q factor in the QR decomposition of an otherwise unconstrained matrix $\tilde{U}$ of free parameters, thereby enforcing orthonormality of the basis while facilitating an unconstrained optimisation procedure.

## 2.2 Mathematical foundations of kSCA

Inspection of equation 5 reveals that the projected data only appears in inner products reducing the spatial dimension (though note that terms inside the trace could have been shifted, thus targeting the temporal dimension instead; see also Section 5). This formulation implies that the kernel trick (Berlinet

& Thomas-Agnan, 2004) may be used, such that the inner products are replaced by positive definite kernel functions $k(\cdot,\cdot)$. Their outputs are by definition equal to the implicitly computed inner products between nonlinear, potentially high- or even infinite-dimensional feature representations $\phi(\cdot)$ of the data. Nonlinear solutions are thus obtained only based on evaluations of the kernel, while alleviating the need to explicitly compute and store mappings to the feature space, an endeavour that could be infeasible in the case of infinite dimensional embeddings. Notably, multiple classical algorithms, including PCA and support vector machines (SVMs), have been kernelised in this fashion, allowing to address problems that could not be solved otherwise (e.g., SVM-facilitated classification of non-linearly separable data; Hofmann et al., 2008; Scholkopf and Smola, 2002).

Specifically focusing on the SCA setting, consider a feature map with potentially infinitely many coordinates $\phi(X_k) \in \mathbb{R}^{\infty \times T}$. The sequentiality objective is then restated as

$$\begin{aligned} \mathcal{S}[\phi(U)] = \frac{2}{K^2} \sum_{k,k'} & \mathrm{Tr}^2 \left[ \phi(U)^\top \tilde{\phi}(X_k) \tilde{\phi}(X_{k'})^\top \phi(U) \right] \\ & - \mathrm{Tr}\left[ \phi(U)^\top \tilde{\phi}(X_k) \tilde{\phi}(X_{k'})^\top \phi(U) \phi(U)^\top \tilde{\phi}(X_k) \tilde{\phi}(X_{k'})^\top \phi(U) \right] \end{aligned} \tag{6}$$

where $\tilde{\cdot}$ denotes the removal of the across-trial mean (i.e., yielding centred feature representations). To enforce orthonormality of the (to-be-optimised) $d$-dimensional subspace $\{\phi(U)\}$ in feature space, a minimizer to the Lagrangian $\mathcal{L} = S[\phi(U)] - \mathrm{Tr}[\Lambda \phi(U)^\top \phi(U)]$ is sought, where $\Lambda$ is a $d \times d$ symmetric matrix of Lagrange multipliers. Differentiating with respect to $\phi(U)$ reveals that the solution lies in the span of

$$\phi(U) = \tilde{\phi}(A)\alpha \tag{7}$$

where $A \in \mathbb{R}^{N \times KT}$ is the matrix of all spatial samples collected at all trials and times, and $\alpha \in \mathbb{R}^{KT \times d}$ is a matrix of parameters. Therefore, kernel SCA consists in optimising $\alpha$ (e.g., in a stochastic gradient manner) according to the revised objective

$$\mathcal{S}(\alpha) = \frac{2}{K^2} \sum_{k,k'} \left[ \mathrm{Tr}^2 \underbrace{\left[ \alpha^\top H \mathcal{K}(A, X_k) \mathcal{K}(X_{k'}, A) H \alpha \right]}_{Q} - \mathrm{Tr}[QQ] \right] \tag{8}$$

where $\mathcal{K}(A, X_k) = \langle \phi(A), \tilde{\phi}(X_k) \rangle$ is a $KT \times T$ Gram matrix, $\mathcal{K}(X_{k'}, A)$ is the equivalent with dimensions swapped, and $H$ a centring matrix given by

$$H = I_T \otimes \left( I_K - \frac{1}{K} \mathbb{1}_K \right) \tag{9}$$

where $\otimes$ is the Kronecker product, $I$ is the identity and $\mathbb{1}_K$ is a $K \times K$ matrix of ones, such that $H$ performs across-trial mean removal for each time-step individually. Importantly, in addition to the centring of $\phi(A)$, achieved by the multiplication by $H$, the mean across conditions $k$ also needs to be removed from $\mathcal{K}(A, X_k)$ and $\mathcal{K}(X_{k'}, A)$, in line with the centred feature representation of $X$ (see equation 6). Again, this is estimated in a Monte-Carlo fashion, with the same mini-batched samples as those used to estimate $\mathcal{S}(\alpha)$. Test samples $\phi(X_*)$ can then be spatially projected onto the learned basis $\phi(U)$ according to $\langle \phi(U), \phi(X_*) \rangle = \alpha^\top H \mathcal{K}(A, X_*)$.

Importantly, as for linear SCA, the projection vectors need to be orthonormal, although now in feature space

$$\langle \phi(U), \phi(U) \rangle = \alpha^\top H \mathcal{K} H \alpha = \alpha^\top \tilde{\mathcal{K}} \alpha = I_d \tag{10}$$

where $\mathcal{K} = \mathcal{K}(A, A) = \langle \phi(A), \phi(A) \rangle$ is the $KT \times KT$ Gram matrix. This is achieved through appropriate parametrisation of $\alpha$, discussed in Section 2.3.

## 2.3 Scaling up kSCA using inducing points

Computing the potentially large Gram matrices in Section 2.2 can prove computationally expensive. This could remain feasible for off-the-shelf kernel functions without learnable parameters, as these can be computed once, outside the optimization loop. Still, approaches for scaling up kSCA are warranted, particularly so for kernels with free parameters, which need re-computation at every iteration, and/or for Gram matrices that are too large to fit in memory.

Here, we capitalise upon inducing point methods, originally introduced to address the computational limitations associated with Gaussian processes (GP; Quinonero-Candela and Rasmussen, 2005; Rasmussen and Williams, 2006). Unless specified otherwise, kernel-based analyses included in this thesis were conducted using inducing points-based approximations, thereby promoting faster computations by alleviating the need to compute and store full Gram matrices. Sparse approximations of the kernel are formulated in terms of a matrix $C \in \mathbb{R}^{N\times c}$, for $c$ latent inducing points. Specifically, the subset of regressors (SoR; Silverman, 1985) approximation reads

$$\mathcal{K}(B, B') \approx \mathcal{K}_{\mathsf{SoR}}(B, B') \triangleq \mathcal{K}(B, C)\mathcal{K}(C, C)^{-1}\mathcal{K}(C, B') \tag{11}$$

for $B$, $B'$ matrices $\in \mathbb{R}^{N\times KT}$, where $\mathcal{K}(B, C) \in \mathbb{R}^{KT\times c}$, $\mathcal{K}(C, C) \in \mathbb{R}^{c\times c}$, and $\mathcal{K}(C, B') \in \mathbb{R}^{c\times KT}$ are obtained from the exact kernel. Here, $\mathcal{K}_{\mathsf{SoR}}(A, A)$ replaces $\mathcal{K}(A, A)$, and $\mathcal{K}_{\mathsf{SoR}}(A, X_k)$ replaces $\mathcal{K}(A, X_k)$. The SoR approximation method is advantageous since, apart from introducing approximate covariances, it does not otherwise alter the algorithmic pipeline (Quinonero-Candela & Rasmussen, 2005): we will simply use $\mathcal{K}_{\mathsf{SoR}}$ as a drop-in replacement for $\mathcal{K}$ everywhere. In addition, unlike alternative approximation techniques (e.g., random Fourier features; Rahimi and Recht, 2007), it is compatible with any valid kernel function. Note however that the SoR-conferred simplicity, which incidentally implies covariances matrices of rank at most equal to $c$, has been criticised and more faithful kernel interpolations have been proposed (Quinonero-Candela & Rasmussen, 2005; Wilson & Nickisch, 2015). Still, as shown below, approximated kernels were found to perform well for kSCA, hence justifying their appropriateness for the purposes of the present thesis.

To enforce orthonormality in this inducing point setting (i.e., $\alpha^T \tilde{\mathcal{K}}_{\mathsf{SoR}}\alpha$ equal to the identity; equation 10), we propose the following parametrisation based on the QR decomposition of $\mathcal{K}(A, C) = QR$ and the Cholesky decomposition of $\mathcal{K}(C, C) = LL^\top$:

$$\alpha = QR^{-\top}L\tilde{\alpha} \quad \text{with} \quad \tilde{\mathcal{K}}_{\mathsf{SoR}}(A, A) = HQRL^{-\top}L^{-1}R^\top Q^\top H \tag{12}$$

where $\tilde{\alpha} \in \mathbb{R}^{c\times d}$ is a unitary matrix obtained as the Q factor of a QR-decomposed, same-shaped matrix of free (unconstrained) parameters. Note that `jax` lets us seamlessly differentiate through all of these linear algebra operations, such that we can automatically obtain gradients of our sequentiality objective w.r.t. those free parameters. Note also that a Cholesky and QR-based approach is advantageous, for instance relative to SVD. SVD can indeed emphasise ill-conditioning-related issues, producing numerical errors for close-to-zero singular values. Moreover, for kernels with free parameters that need to be differentiated through, SVD appears unsuitable as the associated backpropagation results tend to be numerically unstable (Wang et al., 2019).

An important caveat is that orthonormality is only fulfilled in the feature space of the kernel with its inducing points. This becomes relevant, for example when computing the percentage of the total variance explained and comparing with kernel-, and hence inducing points-free approaches (see also Section 2.5).

Another element to consider concerns the choice of the inducing points. In the following analyses, these are initialised as a random subset of the $KT$ spatial vectors. Their locations $C$ are then optimised with respect to the sequentiality objective in Equation 8, an approach analogous to the GP realm, where $C$ is normally optimised w.r.t. the same log marginal likelihood objective as for the other GP kernel hyperparameters (Quinonero-Candela & Rasmussen, 2005).

## 2.4 kSCA kernel functions

Kernel SCA can be interpreted as running the linear SCA algorithm in the space of feature-mapped data. Therefore, when a linear kernel

$$k(z, z') = \langle z, z' \rangle \tag{13}$$

is used for spatial patterns $z, z' \in \mathbb{R}^N$, the implied feature map is the identity map, $\phi(z) = z$. Equations 5 and 6 are thus equivalent and linear SCA is retrieved (though solutions are not expected to be identical if inducing points are used; see Section 2.5).

To leverage nonlinear kSCA, the principal kernel function used in this thesis is the radial basis function (RBF) kernel (but note that in principle, alternative valid kernels that evaluate to dot products in feature space could be used). It is defined as

$$k(z, z') = \exp\left(-\frac{\|z - z'\|^2}{2\ell^2}\right) \tag{14}$$

with $\ell$ denoting the length-scale, separating the distances between $z$ and $z'$ that SCA will consider to constitute small versus large differences. Importantly, the RBF kernel function promotes nonlinear behaviour, as its outputs are provably equivalent to the result of a dot product after infinite expansion in terms of basis functions (Bishop & Nasrabadi, 2006; Rasmussen & Williams, 2006). This implies that kSCA can operate on higher-order moments.

Again, as for the inducing points, the $\ell$ free parameter of the RBF kernel used was optimized with respect to the sequentiality objective in equation 8. It was initialised at 0.1, though precautions were taken to yield sensible results. These were motivated by recognising that computing the Gram matrices in equations 8 and 10 involves aggregating over $N$. If each element of the spatial vectors have $\mathcal{O}(1)$ activity, then the sum of the squared differences will be $\mathcal{O}(N)$. The length-scale should thus be set in $\mathcal{O}(\sqrt{N})$ to achieve kernel outputs in $\mathcal{O}(1)$ for large $N$. Alternatively, averaging, instead of summing over squared differences, can be performed. This latter solution was used here.

## 2.5 Variance captured

To assess the quality of the fits, the fraction of total data variance explained by (k)SCA projections was calculated as 1 minus the normalised reconstruction error. For SCA, this is the same approach as that for PCA, yielding

$$\mathrm{Var} = \frac{\mathrm{Tr}[U^\top \Sigma U]}{\mathrm{Tr}[\Sigma]} \tag{15}$$

where $\Sigma$ is the $N \times N$ covariance matrix of the data (training or test set) with collapsed condition and time. For kSCA, the approach is analogous though the fraction of variance explained is now computed in feature space. The reconstruction error E is given by

$$\begin{aligned}
\mathrm{E} &= \mathrm{Tr}\Big[\Big(\tilde{\phi}(A_*) - \phi(U)\phi(U)^\top \tilde{\phi}(A_*)\Big)^\top \Big(\tilde{\phi}(A_*) - \phi(U)\phi(U)^\top \tilde{\phi}(A_*)\Big)\Big] \\
&= \mathrm{Tr}\Big[\tilde{\phi}(A_*)^\top \tilde{\phi}(A_*) - 2\tilde{\phi}(A_*)^\top \phi(U)\phi(U)^\top \tilde{\phi}(A_*) + \tilde{\phi}(A_*)^\top \phi(U) \underbrace{(\phi(U)^\top \phi(U))}_{I_d} \phi(U)^\top \tilde{\phi}(A_*)\Big] \\
&= \mathrm{Tr}\Big[\tilde{\phi}(A_*)^\top \tilde{\phi}(A_*)\Big] - \mathrm{Tr}\Big[\tilde{\phi}(A_*)^\top \underbrace{\phi(U)}_{\tilde{\phi}(A)\alpha} \phi(U)^\top \tilde{\phi}(A_*)\Big] \\
&= \mathrm{Tr}[\tilde{\mathcal{K}}(A_*, A_*)] - \mathrm{Tr}[\alpha^\top \tilde{\mathcal{K}}(A, A_*) \tilde{\mathcal{K}}(A_*, A)\alpha]
\end{aligned} \tag{16}$$

with $A_*$ indicating the $N \times K'T$ test set comprising $K'$ samples. Further dividing by $\mathrm{Tr}[\tilde{\mathcal{K}}(A_*, A_*)]$ gives the normalised reconstruction error. The fraction of variance captured is thus calculated as

$$\mathrm{Var} = \frac{\mathrm{Tr}[\alpha^\top \tilde{\mathcal{K}}(A, A_*)\tilde{\mathcal{K}}(A_*, A)\alpha]}{\mathrm{Tr}[\tilde{\mathcal{K}}(A_*, A_*)]} \tag{17}$$

When linear kernels are relied on, and when there are no sparse approximations of the Gram matrices (e.g., inducing points-based), equations 15 and 17 are provably equivalent. Yet, in the inducing point framework used throughout this thesis, $\alpha$ is parameterised such that $\phi(U)$ constitutes an orthonormal basis in the feature space implied by the approximated kernel, which potentially differs from that implied by the exact kernel. Consequently, the sensitivity to inducing points of the numerator in equation 17 prevents direct comparisons between the outputs of equations 15 and 17 even when a linear kernel is used.

## 2.6 Summary

Sequentiality is an anticipated property of most dynamical systems in which state-space trajectories obey a lawful flow-field, except if the system is strongly driven by external inputs (Rutten et al., 2020a). This sequentiality can be identified and its organisation can be studied using SCA, which performs dimensionality reduction through maximising a second-order measure of non-reversibility. Additionally, the ability to kernelise the SCA algorithm allows to include higher-order moments. As will be shown in this thesis, a kernel-based approach is sometimes crucial to recover non-linearly accessible sequentiality.

The (k)SCA method could be valuable, for instance to study neural computations suggested to be instantiated by a dynamical system (Churchland et al., 2012; Galgali et al., 2023). Still, possible applications extend beyond neuroscience. Non-reversibility as a learning goal could indeed serve to extract meaningful projections in various domains, such as bioinformatics and genomics, where developmental patterning trajectories characterise how fate specification emerges from initially transcriptionally equivalent cells (Mi et al., 2018; Shi et al., 2021).

# 3 Recovering synthetic sequential spatio-temporal processes

## 3.1 (k)SCA surpasses PCA in discovering noise-corrupted, linearly accessible latent dynamics

Linear SCA and its kernel counterpart were first validated on a range of synthetic datasets with known ground truth non-reversible dynamics. First, $K = 100$ pure rotations were generated from combined sine and cosine oscillations starting at varying phases and associated with different radii. Time ranged from 0 to $2\pi$, corresponding to a full rotation. These highly reversible processes were then projected onto a 50-dimensional orthonormal basis, and low-rank, temporally correlated noise was added. The low-dimensional noise subspace was generated as $B\epsilon_t$, where $B$ is a $N \times 3$ random orthogonal matrix. To obtain $\epsilon_t$, normal samples from a distribution $\mathcal{N}(0, \sigma^2)$, generated anew for each time step and condition, were multiplied by the lower-triangular Cholesky factor of a $T \times T$ RBF Gram matrix, whose length-scale was set to approximately match the dynamics of the rotational signal. In this setting, as $\sigma^2$ becomes larger, the plane where the rotational trajectories of interest unfold is expected to be discarded by PCA, which will instead capture the high-variance noise fluctuations. In contrast, corruption by large low-rank signals is not anticipated to be as detrimental for SCA and kSCA. This is because (k)SCA should not capture statistically reversible noise. In the examples that follow, $\sigma^2$ was selected to highlight this important difference between (k)SCA and PCA.

(k)SCA-based projections were learned according to the sequentiality objectives in Sections 2.1 and 2.2 on a training set consisting of 80 (out of 100) trajectories, with mini-batches of 100 pairs. The parameters for $U$ (SCA) and $\alpha$ (kSCA) were randomly initialised according to a standard normal distribution. For kSCA, $c = 30$ inducing points started as a randomly selected subset of the spatial vectors. The Adaptive Moment Estimation (Adam) optimizer with initial learning rate $10^{-3}$ was used. Convergence under these settings was verified by tracking the losses (i.e., $-\mathcal{S}(U)$ and $-\mathcal{S}(\alpha)$; see Appendix C for losses associated with all Figures in this Section).

Results are displayed in Figure 2 for the training set (left) and for 20 test set trajectories (right). 2-dimensional ground-truth, high-$\zeta$, trajectories (upper row) appeared to be better recovered by SCA (training: $\zeta = 0.84$, test: $\zeta = 0.63$) and by kSCA (training: $\zeta = 0.90$, test: $\zeta = 0.79$ with the RBF kernel) relative to PCA (training: $\zeta = 0.01$, test: $\zeta = 0.02$), which failed to de-mix rotational signals from the high-variance distractor.

Furthermore, PCA projections captured a greater percentage of the total data variance (training: 48.85%, test: 49.19%) relative to SCA (training: 36.24%, test: 31.22%) and kSCA (training: 22.01%, test: 21.43%). These results are in line with the variance-maximising learning goal of the PCA algorithm, which in this case does not constitute the most adapted inductive bias.

Similar findings were obtained for PCA, SCA, and kSCA, when evaluated on additional trajectories that also followed lawful flow fields: the Van der Pol oscillator, with determined derivatives $\frac{dx_1}{dt} = x_2$ and $\frac{dx_2}{dt} = (1 - x_1)^2 x_2 - x_1$ (Figure 3a), and the Duffing oscillator, given by $\frac{dx_1}{dt} = x_2$ and $\frac{dx_2}{dt} = x_1 - x_1^3$ (Figure 3b). The procedure to obtain the embedded, noise-corrupted trajectories, as well as to train the models was the same as for Figure 2. Again, in both cases, (k)SCA, but not PCA, retrieved the ground-truth structure, whereas explained variance was higher for PCA. Overall, these results demonstrate that (k)SCA performance is not contingent on a high signal-to-noise ratio.

## 3.2 Potential relevance of (k)SCA processing in neuroscience settings

This ability to extract sequential dynamics in spite of distractor noise processes appears particularly valuable for the analysis of noise-laden neural data. To illustrate this, we generated high-dimensional, noise-corrupted neuronal spike-like sequences in a system of $N = 50$ units. At each $k \in K = 100$ trial (split as before between 80 training and 20 test), a particle travels clockwise on a circle, starting from a random phase in $[0, 2\pi)$ and continuing until a full rotation is achieved. Each neuron $n \in N$ has a

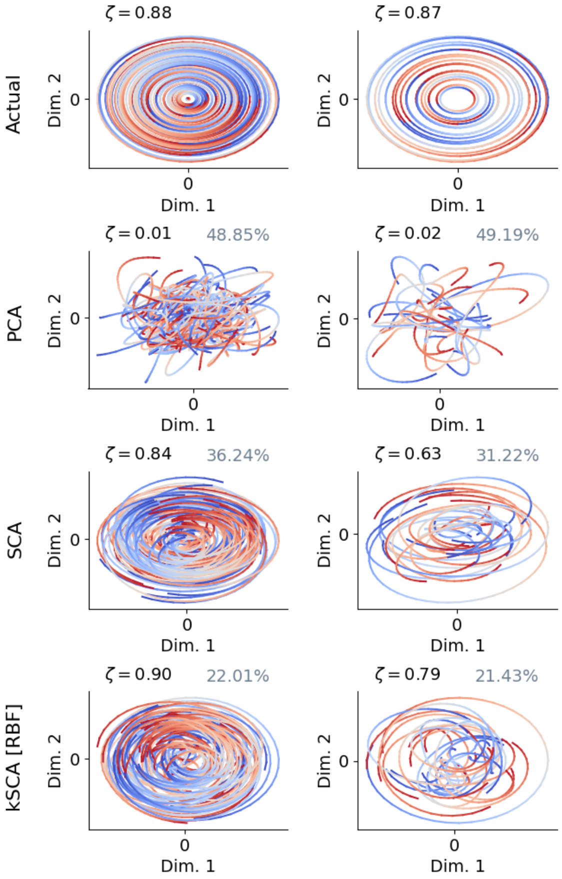

Figure 2: **SCA and kSCA recover embedded latent rotations**. Pure rotations (upper row) were projected in higher-dimensional space plus temporally correlated noise (drawn from a Gaussian Process with a smooth RBF kernel) of variance $\sigma^2 = 1$. The 2-dimensional rotational dynamics were recovered by SCA (3rd row) and kSCA (RBF kernel; 4th row), with the degree of sequentiality indicated by $\zeta$. PCA (2nd row) appeared to have captured the low-rank noise dynamics instead. The percentage of the total data variance explained is indicated in grey. The training (80 conditions) and test (20 conditions) set results are shown on the left and right columns, respectively. Colour-coding indicates time in radians, from earlier time steps (blue) to more recent ones (red).

preferred direction $\theta_n$, around which its response function falls off according to a Von Mises tuning curve. That is, at position $\theta$, the unit activity is given by

$$f(\theta - \theta_n) = \exp\left(\frac{\cos(\theta - \theta_n) - 1}{\kappa}\right) \tag{18}$$

where $\kappa$ denotes the tuning curve width. Therefore, as the particle moves around the circle, different, albeit similarly-tuned units, will be recruited. The signal was then mixed with low-rank, temporally-reversible noise, generated in the same fashion as that described earlier. Notably, this dataset can be related to the fast sequences of precisely timed hippocampal place cell spikes recorded during replay events (Drieu & Zugaro, 2019; Ólafsdóttir et al., 2015).

Results for PCA, SCA and kSCA are shown in Figure 4 for $\sigma^2 = 1.25$ and $\kappa = 0.1$. Similar to Figures 2 and 3, SCA (training: $\zeta = 0.92$, test: $\zeta = 0.96$) and kSCA (training: $\zeta = 0.96$, test: $\zeta = 1.0$), but not PCA (training $\zeta = 0.0$, test: $\zeta = 0.02$), were able to identify a 2-dimensional, highly sequential manifold,

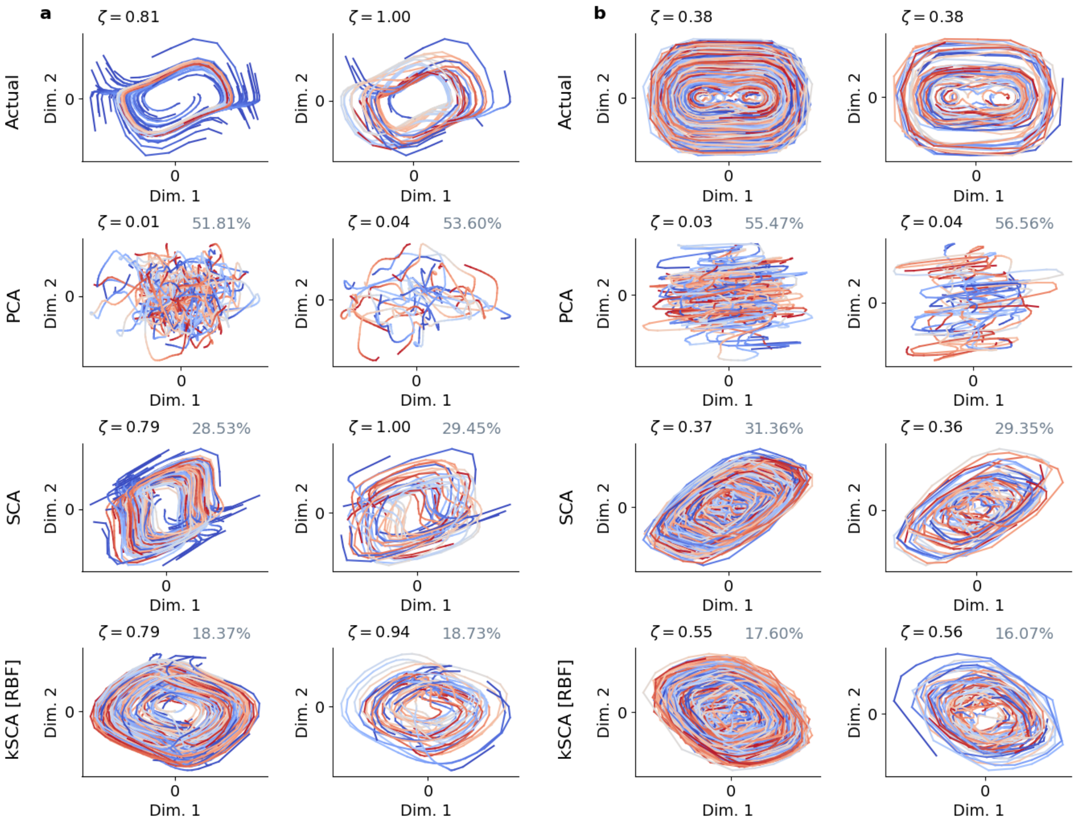

Figure 3: **SCA and kSCA recover embedded dynamical systems**. Same as Figure 2 but for (a) the Van der Pol (noise variance $\sigma^2 = 2$) and (b) the Duffing (noise variance $\sigma^2 = 3$) oscillators, whose determined derivatives imply statistical sequentiality.

in which the neural dynamics unfold. Analogous results were obtained across values of $\kappa$, up to a point for sufficiently small $\kappa \leq 10^{-3}$ (i.e., little co-activity between units), where SCA and kSCA failed (not shown). This could be because the sequence becomes higher-dimensional, making it less straightforward to infer a corresponding 2-dimensional plane. Still, neighbouring biological receptive fields (e.g., in the hippocampus; Drieu and Zugaro, 2019) do often overlap, supporting that a (k)SCA approach could prove appropriate to unveil population dynamics in a neuroscience setting.

### 3.3 Performance differences between SCA and kSCA for non linearly accessible latent dynamics

For the time series investigated thus far, kSCA-mediated nonlinear dimensional reduction did not seem crucial to recover ground-truth, non-reversible dynamics. Yet, as alluded to in the introduction, certain trajectories that are intuitively non-reversible are missed by linear SCA, which constitutes an important limitation of the method. This is exemplified in Figure 5. Pure rotations in the polar domain were generated in the same manner as for Figure 2. These were then converted to the Cartesian domain as $x = r\cos(\theta)$ and $y = r\sin(\theta)$, for radius $r$ and angle $\theta$ (in radians), yielding the infinity-looking trajectories in Figure 5 (upper row). The training set was associated with $\zeta = 0.02$, already demonstrating that sequentiality was not captured by the linear SCA metric.

Again, the trajectories were embedded in a 50-dimensional space and temporally-correlated noise was

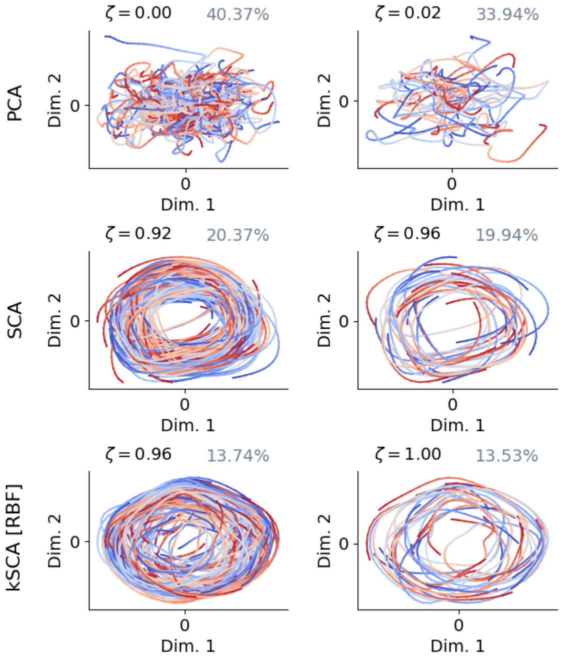

Figure 4: **SCA and kSCA recover the dynamics of a biologically-inspired high-dimensional system**. Sequential dynamics of a travelling particle, whose position is encoded by $N = 50$ direction-sensitive neurons (tuning width $\kappa = 0.1$), corrupted by noise of variance $\sigma^2 = 1.25$, could be extracted by SCA and kSCA but not by PCA. Figure arrangement is the same as for Figure 2 (except that multi-dimensional ground-truth trajectories are not shown).

added ($\sigma^2 = 0.75$). Had noise been absent, it would have been possible to retrieve the ground-truth dynamics using linear SCA, as it would return its input, with no de-mixing of the relevant trajectories taking place (not shown). Yet, when incorporating noise, SCA can only perform well if there is a sequential signal it can pick up. This was not the case, and linear SCA failed to retrieve the ground-truth dynamics (Figure 5). In contrast, noise was not as detrimental for kSCA. Indeed, it appeared to have successfully converted the 2 wing-trajectory into a single, more rotation-like pattern with high $\zeta = 0.67$ (training set) and $\zeta = 0.41$ (test set; Figure 5). The rotational structure that can be learned with kSCA is more apparent in the bottom row of Figure 5. It shows learned projections based on the same polar to Cartesian rotation dataset embedded in 50-dimensional space, but without adding noise. kSCA yielded $\zeta = 0.80$ (training set) and $\zeta = 0.45$ (test set).

The results highlight the potential of kSCA in identifying curved manifolds where sequential dynamics unfold, even when these are not accessible to linear SCA. This flexibility supports that the kSCA method can be used to robustly discover latent states across a large range of possible dynamical systems. Appendix B presents another example of non-reversible trajectories that are considered reversible by linear SCA, but not by kSCA, with applications to the neural bases of decision-making.

## 3.4 Discussion

This Section has highlighted the performance differences between PCA, SCA, and kSCA. In particular, it has demonstrated that the learning goals inherent to PCA and (k)SCA, variance and sequentiality-related, respectively, result in varying robustness to noise corruption. This appears particularly beneficial for processing real-life noise-laden datasets. These extend beyond the hippocampal reactivation-inspired

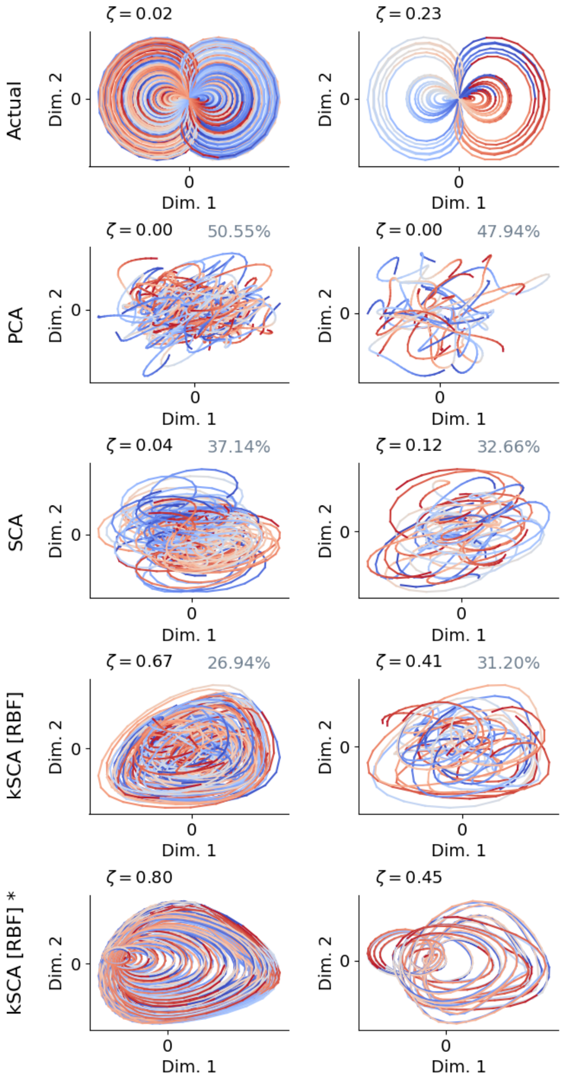

Figure 5: **Differences in SCA and kSCA performance for certain datasets**. Same as Figure 2 up to the second-to-last row, but for polar domain rotations converted to Cartesian coordinates (noise variance $\sigma^2 = 0.75$). Bottom row, indicated by the * sign, shows learned kSCA projections based on the same dataset (i.e., polar to Cartesian rotations) projected in 50-dimensional space, though without adding temporally correlated noise. In that case, SCA can recover infinity-looking trajectories, but achieves low $\zeta = 0.03$ (training set; not shown).

dataset in Section 3.2. For instance, functional magnetic resonance imaging (fMRI)-recorded blood-oxygen level dependent (BOLD) signals are typically dominated by slow, global, and reversible fluctuations (Bolt et al., 2022), which need to be statistically controlled for (Chiu et al., 2012). It is plausible that a (k)SCA-based approach could help de-mixing such reversible fluctuations from signals of interest.

Section 3.3 has also demonstrated that certain non-reversible datasets fail to be recognised as such by linear SCA. Instead, they are deemed highly reversible with $\zeta \approx 0$. Consequently, such datasets do not constitute a strong enough signal to be captured by linear SCA when sufficient levels of distractor noise are present. This is an important limitation, as it restricts the range of forms of dynamics whose spatio-temporal organisation can be studied with linear SCA. Importantly, these concerns were alleviated using RBF kSCA, thereby showing the potential usefulness of the kSCA algorithm.

# 4 SCA in the motor domain

## 4.1 Motivation & background

Having validated (k)SCA on synthetic datasets, it is now deployed on electrophysiologically-recorded neurons in PMd and M1. Differing theoretical accounts have been put forward to characterise the response patterns of motor and premotor cortical neurons, as well as the mechanisms by which they control movement (Churchland & Shenoy, 2007; Churchland et al., 2012). One early view has been that neural activity represents movement parameters, such as hand velocity or target position, which was supported by evidence of spatial (e.g., direction) tuning (Georgopoulos et al., 1982). Still, detailed descriptions of the spatio-temporal properties of neuron responses challenged this view. These have repeatedly been described as multiphasic and highly heterogeneous, spanning a larger space than would be anticipated if they encode a moderate number of movement parameters (Churchland & Shenoy, 2007). This has led researchers to propose that the motor cortex instead constitutes a dynamical system that drives muscles, generating and controlling movement (Churchland et al., 2012; Suresh et al., 2020). In this case, motor control is not contingent upon a direct mapping between individual neurons and specific movement covariates (although such a relationship may occur incidentally), but rather depends on enforcing the right latent state-space trajectory at the neural population level (Churchland & Shenoy, 2007; Gallego et al., 2017). Reports of dependencies amongst motor cells, consistent with population-level coding, aligned with a dynamical account of motor cortical function (Elsayed & Cunningham, 2017; Paninski et al., 2004).

Additional support for a dynamical systems perspective was obtained using jPCA-based dimensionality reduction, which revealed the existence of rotations of the population state in monkeys (Churchland et al., 2012), but also in other species (incl. humans; Pandarinath et al., 2015). Notably, population-level rotational dynamics were replicated using alternative approaches to jPCA (Lara et al., 2018; Pandarinath et al., 2018a; Rutten et al., 2020a; Sani et al., 2021; Schimel et al., 2022). This shows that the structure identified in Churchland et al. (2012) was not due to the rotation-oriented search, thereby addressing potential concerns (Lebedev et al., 2019). Simulation studies further suggested that rotational structure in biological data results from latent dynamics. Researchers trained a recurrent neural network (RNN) to generate the electromyographic (EMG) signals elicited during reaching. jPCA projections of the resulting neural population responses revealed rotations resembling those observed in the motor cortex (Sussillo et al., 2015; see also Hennequin et al., 2014; Michaels et al., 2016, for other instances of artificial dynamical systems yielding rotational dynamics despite not being trained on empirical neural data).

However, the precise means by which low-dimensional circuit dynamics relate to movement are yet to be elucidated. In this regard, isolating the least reversible components of neural activity through (k)SCA could prove valuable. Indeed, if these estimate the movement-generating activity manifold, chances are that they will be most reflective of the associated behaviour. The spatio-temporal structure of non-reversibility could then be explored, and the nature of the relationship with associated behavioural signals could be assessed.

## 4.2 Dataset and pre-processing

(k)SCA was applied to the `MC_Maze` neural dataset (https://dandiarchive.org/dandiset/000128) from Pei et al. (2021). Data was recorded as a rhesus monkey engaged in a Maze task involving delayed reaching movements (Churchland et al., 2010). Specifically, the animal used a cursor, projected above its fingertips. At the beginning of each trial, it held the cursor inside a central fixation point. A target was then presented, along with virtual barriers to avoid, as well as unreachable distractor targets that needed to be ignored. Following a subsequent go cue, the monkey broke fixation and made a reach directed at the target. Different trials comprised different boundaries configurations and target locations, yielding $K = 108$ behavioural conditions.

Electrophysiological recordings were performed using multi-electrode arrays, implanted in the M1 and PMd cortical regions. Spike sorting was then conducted offline. In total, the dataset included 182 neurons over 1721 trials. The analyses below either focused on peri-stimulus time histograms (PSTHs) or on single trials. Notably, single trial-level insight is crucial to understand the representational properties of natural movements, which are typically non-repeated and hence cannot be averaged across trials (Pandarinath et al., 2018b).

Pre-processed outputs provided by Pei et al. (2021) were relied on. Specifically, to compute PSTHs, activity was convolved with a Gaussian kernel (70 ms standard deviation) before being averaged across trials belonging to the same condition, thereby providing a de-noised representation. For single trials, only Gaussian smoothing with the same 70 ms standard deviation was performed (Pei et al., 2021). In both cases, data was aligned based on movement onset, from 250 ms before to 450 ms after, and neural activity was grouped in 20-ms bins. Additionally, $v_x-$ and $v_y-$fingertip velocities were monitored, thereby allowing to relate neural activity to behaviour. Again, both same-condition trial averages and individual trial versions of the monkey hand velocity data were considered.

PSTH and single trial neural responses were further pre-processed prior to dimensionality reduction. These were first divided by a normalisation constant given by the maximum between the maximal activity for each neuron across conditions and times and 0.1. This is analogous to the soft-normalisation in Churchland et al. (2012), which ensured that weakly active cells had less than unity range and that highly active neurons did not dominate. For PCA and linear SCA, activity of each neuron was then mean-centred across conditions (this is automatically achieved in the kSCA implementation; see Section 2.2). As noted previously (Churchland et al., 2012), removing the mean ensures that the data variance retained is condition-specific. Otherwise, spatial projections would be similar for all conditions, potentially making it more difficult to rule out more trivial, dynamical system-unrelated reasons underlying population-level findings.

## 4.3 (k)SCA unveils rotation-like structure in neural population responses

SCA and RBF kSCA projections were learned from the neural PSTHs according to the optimisation procedure described previously. That is, training was performed on mini-batches of 100 pairs using the Adam optimiser with initial learning rate $10^{-3}$ (see Section 3 for more details). The procedure was repeated for subspace dimensionalities $d$ ranging from 2 to 10, and convergence was confirmed (see losses for this Section in Appendix C). Results are shown in Figures 6a, b, c, d. Again, for comparison, PCA projections are included.

The projections for $d = 2$ are displayed in Figure 6a. While PCA failed to detect rotational patterns ($\zeta = 0.00$), SCA ($\zeta = 0.26$) and kSCA ($\zeta = 0.30$) both exhibited rotational structure analogous to that accessed through jPCA (Churchland et al., 2012), which directly seeks pure rotations. In contrast, (k)SCA searches for rotational dynamics in an indirect fashion through identification of sequential behaviour. Being able to still recover rotations thus adds to the previous work (Lara et al., 2018; Rutten et al., 2020a) that validates the findings in Churchland et al. (2012).

Sequentiality, quantified using the $\zeta$ index, is additionally displayed as a function of the subspace dimensionality in Figure 6b. $\zeta$ was higher for spatial projections to higher-dimensional subspaces, potentially due to the emergence of multiple sequential (e.g., rotatory) planes. Moreover, nonlinear subspace identification with kSCA appeared to confer greater sequentiality relative to SCA. Still, both kSCA and SCA exhibited a higher $\zeta$ relative to PCA, in line with their sequentiality-maximising optimisation goal. The percentage of variance captured by the projections is also indicated in Figure 6c. The explained variance was higher for PCA relative to SCA and kSCA, reflecting the variance-maximising goal of PCA.

Finally, similarity between latent trajectories extracted by PCA, SCA, and kSCA, was examined using canonical correlation analysis (CCA). Briefly, CCA seeks vectors (i.e., canonical directions) that linearly transform pairs of projections (each reshaped as $KT \times d$) such that the re-weighted datasets (i.e., canonical variables) are maximally correlated. A subsequent pair of canonical variables is then extracted, subject to the constraint that these are uncorrelated to the previous pair of canonical variables. This procedure can be repeated up to the minimum number of units across both datasets (i.e., $d$ here). The resulting sets of canonical variables are naturally ordered according to their degree of correlation $\rho$, where $\rho = 1$ indicates that the two datasets comprise the same underlying patterns, whereas $\rho = 0$ denotes no common underlying pattern (Raghu et al., 2017; Sussillo et al., 2015). The average canonical correlation $\overline{\rho}$ (i.e., sum of the canonical correlations divided by $d$) is reported in Figure 6d as a summary statistic. $\overline{\rho}$ was the highest across subspace dimensionalities for kSCA and SCA, indicating similar projections, in line with the visualisations in Figure 6a. This is consistent with the observation that linear (rotational) dynamics, which are accessible to both SCA and kSCA, are a dominant feature of motor population responses (Churchland et al., 2012; Lara et al., 2018). In contrast, kSCA- and PCA-, as well as SCA- and PCA-extracted trajectories were more dissimilar according to CCA.

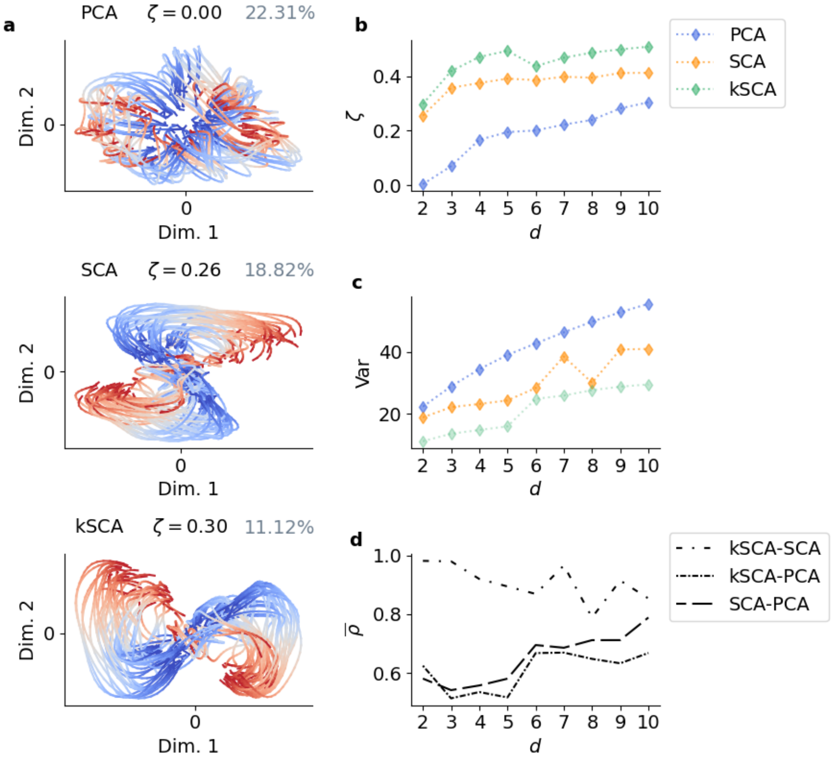

Figure 6: **Application of (k)SCA to PSTHs**. (a) 2-dimensional embeddings obtained via PCA, SCA, and kSCA. PCA produced a low $\zeta = 0.00$, reversible projection. SCA and kSCA captured rotational-like structure, with $\zeta = 0.26$ and $\zeta = 0.30$, respectively. In contrast, the explained variance was higher for PCA (22.31%) relative to SCA (18.82%) and kSCA (11.12%). Time is colour-coded, from earlier time steps in blue (250 ms prior to movement onset) to more recent ones in red (450 ms after movement onset). (b) $\zeta$ as a function of the dimensionality of the embeddings. The kSCA-extracted projections were the most sequential, followed by the SCA- and the PCA-obtained ones. (c) Percentage of variance explained as a function of dimensionality. It was the highest for PCA, followed by SCA, and kSCA. Note that the variance explained values for kSCA, indicated in faded green, should technically not be directly compared with those obtained for PCA and SCA due to their sensitivity to inducing points (see Section 2.5). (d) Average canonical correlation $\overline{\rho}$ as a function of dimensionality. Trajectories were the most similar for kSCA-SCA, followed by SCA-PCA, and by kSCA-PCA.

The above analyses were also conducted on the single trial neural data. Most of the claims deduced from condition-averaged data remained true: (k)SCA produced rotating projections (Figure 7a). Moreover, kSCA attained the highest $\zeta$ scores, followed by SCA, with PCA projections exhibiting the lowest sequentiality (Figure 7b). The variance explained was higher for PCA relative to SCA and kSCA (Figure 7c). Finally, for most $d$ settings, kSCA and SCA trajectories exhibited the highest similarity according to CCA (Figure 7d). This alignment with PSTH-derived findings highlights the capacity of (k)SCA to discover population-level latent dynamics (e.g., rotational) based on single trial data. Note that this is not a trivial endeavour given the trial-to-trial variability in the activity of individual neurons (Galgali et al., 2023).

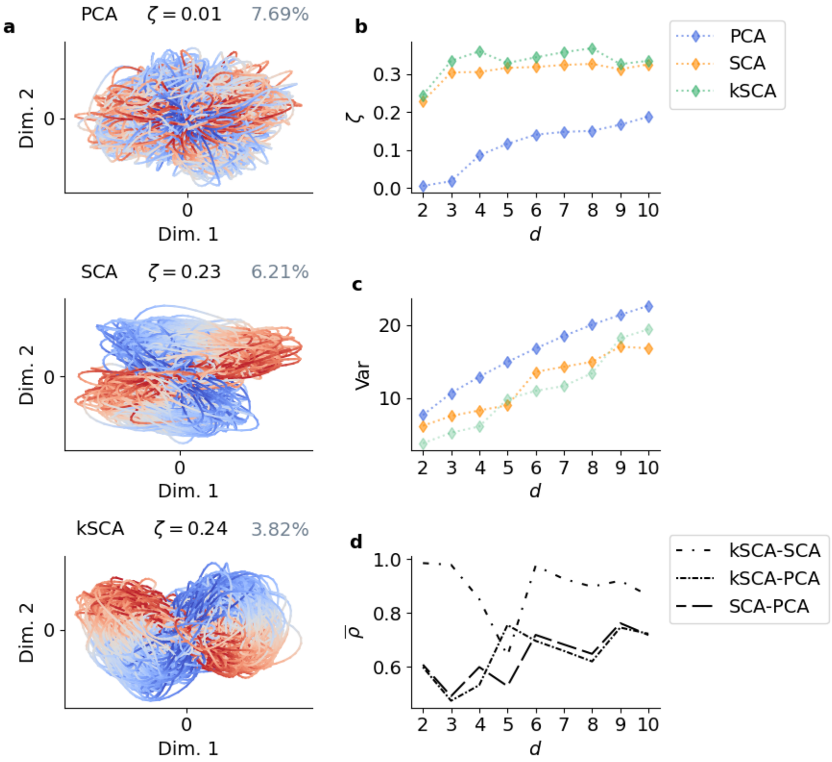

Figure 7: **Application of (k)SCA to single trial neural data**. Same as Figure 6 but for single trial neural spiking data.

## 4.4 Regression-based mapping between (k)SCA-represented neural activity and behaviour

### 4.4.1 Predicting hand velocity

We then evaluated the extent to which the projections in Section 4.3 mapped onto behaviour. The PSTH projections were separated into a test set (20 behavioural conditions) and a training set (88 remaining conditions), and were reshaped collapsing the condition and time dimensions. The training set trajectories were then entered as input to a linear regression model that predicted hand velocity with a 100-ms lag (determined to maximise decoding in Pei et al., 2021), which accounted for the fact that behaviour lags behind the neural activity that controls it. The model was trained to minimise the mean squared error (MSE) between predicted and actual hand velocity. L2 weight regularisation constrained the regression to mitigate over-fitting. The regularisation parameter $\lambda$ was selected to maximise MSE performance via leave-one-out cross-validation (Pedregosa et al., 2011).

Test set $R^2$ scores are shown as a function of the dimensionality of the subspace in Figure 8 (left). SCA and kSCA achieved similar performances, except for a $R^2$ drop for SCA at $d = 8$, which could be due to local minima in the loss landscape. Note that, with extended computational and temporal resources, training multiple instances of the (k)SCA dimensionality reduction models, starting at varying initialisation settings, could have further helped confirm the robustness of the findings. Still, kSCA therefore did not appear to strongly improve predictive performance relative to SCA. This is consistent with the CCA-revealed output similarity between both methods in the motor control context (Figure 6d).

At low dimensions, there was no predictive advantage for (k)SCA relative to PCA, suggesting that capturing more variance, rather than sequentiality, benefits hand velocity prediction. For higher dimensions, (k)SCA appeared to enhance behavioural predictivity relative to PCA, though this effect was not strong. Notably, this difference started to emerge at $d = 6$. This also constituted a point at which the variance explained by (k)SCA grew (Figure 6c), and the CCA-indicated similarity between kSCA-PCA and SCA-PCA increased (Figure 6d). Therefore, though sequentiality might contribute to accounting for hand velocity behaviour, variance captured remains essential.

These regression-based analyses were replicated for the single trial data (Figure 9). The procedure was similar except that 200 trials (out of 1721) were used for the test set. Results were analogous to the condition-averaged case: a predictive advantage for (k)SCA emerged at points where CCA-indicated overlap with PCA trajectories increased.

It is also noteworthy that, for both PSTHs and individual trials, hand velocity predictivity was systematically lower for all dimensionality reduction methods relative to the $R^2$ obtained based on the $N$ neural activity traces (grey horizontal line). This is consistent with the previously identified underlying dimensionality of the data, found to be 10–20 dimensions (Churchland & Shenoy, 2007; Yu et al., 2009).

### 4.4.2 Predicting movement-generating torques

For all experiments thus far, the advantage of (k)SCA over PCA was limited. This could be because, while muscle contractions giving rise to movement in the Maze task are largely sequential, the observed hand position and velocity might not be. For example, the monkey could move the cursor forward and backwards to reach the target, implying statistical reversibility, despite different underlying muscle commands. This is supported by low a sequentiality score for hand velocity data ($\zeta$=0.04). To further explore this, we instead evaluated predictions of movement-generating torques, extracted by G. H. from a two-jointed arm model (Kao et al., 2021; Li & Todorov, 2004).

The model posits a two-link arm, where the upper and lower arms are connected at the elbow. The two links are each associated with mechanical parameters $L_1$ and $L_2$ lengths, $M_1$ and $M_2$ masses, as well as $I_1$ and $I_2$ moments of inertia, determined previously (Kao et al., 2021; Li & Todorov, 2004). Additionally, a distance $S_2$ separates the centre of mass of the lower limb from the elbow. These allow to form matrices for $\mathcal{M}$ inertia, $\mathcal{X}$ centripetal and Coriolis forces, as well as $\mathcal{B}$ joint friction (Kao et al., 2021)

$$\mathcal{M}(\boldsymbol{\theta}) = \begin{pmatrix} a_1 + 2a_2\cos\theta_2 & a_3 + a_2\cos\theta_2 \\ a_3 + a_2\cos\theta_2 & a_3 \end{pmatrix}$$

$$\mathcal{X}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = a_2 \sin\theta_2 \begin{pmatrix} -\dot{\theta}_2(2\dot{\theta}_1 + \dot{\theta}_2) \\ \dot{\theta}_1^2 \end{pmatrix} \qquad \mathcal{B} = \begin{pmatrix} 0.05 & 0.025 \\ 0.025 & 0.05 \end{pmatrix} \tag{19}$$

where $a_1 = I_1 + I_2 + M_2 L_1^2$, $a_2 = M_2 L_1 S_2$, $a_3 = I_2$, and $\boldsymbol{\theta} = (\theta_1; \theta_2)^\top$ indicates the joint angles determining the arm position (see Kao et al., 2021 for further details). A differential equation describing the evolution of $\boldsymbol{\theta}$ is then defined as

$$\mathcal{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathcal{X}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathcal{B}\dot{\boldsymbol{\theta}} = \boldsymbol{m}(t) \tag{20}$$

where $\boldsymbol{m}(t)$ corresponds to the momentary torques of interest at time step $t$.

Together, the angular position and velocity, $\boldsymbol{\theta}$ and $\dot{\boldsymbol{\theta}}$, respectively, were used to define a state. The dynamics of the arm model were integrated given the current state and the momentary torques, yielding angular velocity $\dot{\boldsymbol{\theta}}$ and acceleration $\ddot{\boldsymbol{\theta}}$ according to Equation 20. At each time step, hand position estimates in Euclidean space could be obtained from the angular state as

$$\begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} = \begin{pmatrix} L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2) \\ L_1 \sin\theta_1 + L_2 \sin(\theta_1 + \theta_2) \end{pmatrix} \tag{21}$$

The time series of torques were optimised using Adam with initial learning rate $10^{-1}$. An MSE loss enforced a matching between the states-derived estimated hand position and velocity and the actual time-varying position (discretised as $x(t) = x(t-1) + v_x(t)\Delta t$ and $y(t) = y(t-1) + v_y(t)\Delta t$, with $\Delta t = 10{-}3$) and velocity. An L2 squared norm penalty with parameter $\lambda = 10^{-3}$ on the torques additionally helped ensure that these were as small as possible, while capturing behaviour.

Predicting torques enhanced the distinction between sequentiality and variance-focused dimensionality reduction methods (Figure 8, right). Therefore, the results support a greater information content overlap between (k)SCA neural projections and behaviour in torque space, which better reflects the sequential nature of the movement generation process. Note that torques were only obtained for the condition-averaged data. Establishing the extent to which the results transfer to individual trials could thus constitute the focus of future work.

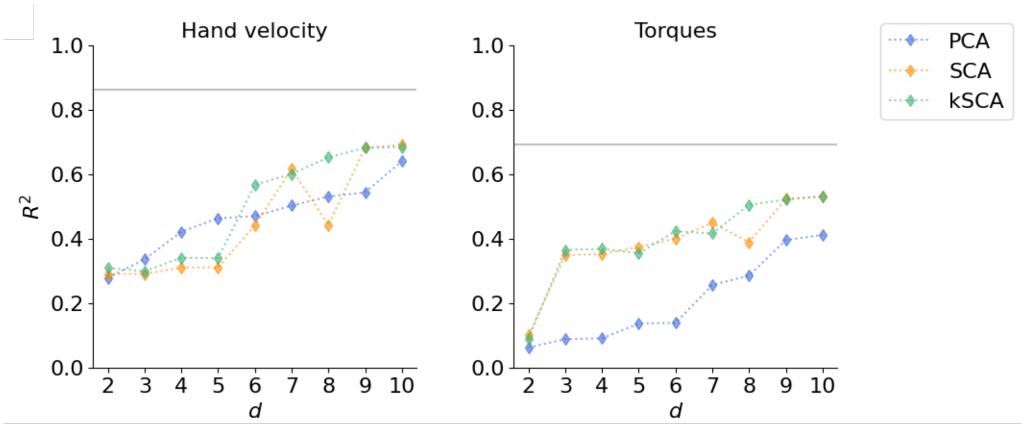

Figure 8: **Information content overlap between (k)SCA-projected population activity and behaviour for PSTHs**. Test $R^2$ scores as a function of the subspace dimensionality for the linear regression model. PCA, SCA, and kSCA-extracted neural manifolds predicting hand velocity (left) and torques (right). Grey horizontal lines denote test $R^2$ achieved by the $N$ neural activity traces.

## 4.5 Most sequential components of neural data reflect the most sequential components of behaviour

### 4.5.1 Canonical correlation analysis

The above results suggested that the most sequential components of neural activity map onto behaviour, at least for sequential signals such as torques. It is thus possible that performing (k)SCA also on behavioural data, and hence extracting its most non-reversible components, could better reveal a putative

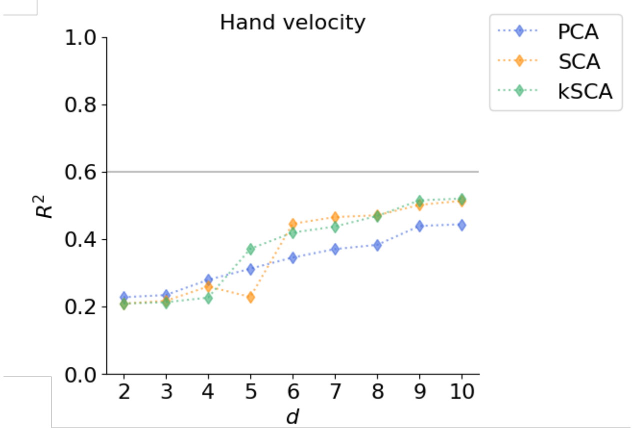

Figure 9: **Information content overlap between (k)SCA-projected population activity and behaviour for single trials**. Same as Figure 8 (left) but for single trial neural spiking data.

isomorphism (i.e., structure-preserving mapping) between brain and behaviour. To examine this, SCA, kSCA, and PCA were applied to augmented behavioural data, which combined hand position, velocity, and torques, thereby providing a more exhaustive characterisation of the movements with $\zeta = 0.30$. Moreover, the mapping between neural and behavioural projections was evaluated using methods providing a finer-grained insight into possible differences across dimensionality reduction methods relative to regression.

First, CCA was applied to (k)SCA and PCA-extracted projections of neural activity and of augmented behaviour (reshaped as $KT \times d$), with a lag of 100 ms. The dimensionality $d$ of the projections used was 6, the maximum possible here, matching the dimensionality of the augmented behaviour. Results for PCA, SCA, kSCA are shown in Figures 10a, b, and c, respectively. Columns correspond to the response patterns shared across neural activity and behaviour, where each individual line indicates a different behavioural condition unfolding over time. The canonical correlations denote the Pearson correlation coefficient between the canonical variables. For the most correlated canonical variable CV1, neural and behavioural patterns were similar irrespective of the dimensionality reduction technique used. Still, at least one very high correlation can often be anticipated, particularly so for datasets with broad similarities. This is the case here as, irrespective of the dimensionality reduction method used, both neural and behavioural datasets encode the same reaching task, suggesting shared basic temporal properties such as motor preparation followed by motion-associated activity (Sussillo et al., 2015).

For less correlated canonical variables (e.g., CV5, CV6), neural and behavioural patterns matched to a greater extent for kSCA and SCA. In contrast, PCA-derived neural and behaviour canonical variables appeared more discrepant. Quantitatively, this is reflected in the canonical correlations, which remained higher even at CV6 for SCA ($\rho = 0.12$) and kSCA ($\rho = 0.22$), whereas $\rho$ had dropped to 0.01 for PCA. To summarise, the average canonical correlation $\overline{\rho}$ was additionally calculated for all dimensionality reduction methods. It was higher for SCA (0.58) and kSCA (0.65) than for PCA (0.52). These results support a greater number of matching patterns for the (k)SCA-extracted most sequential components of neural activity and of behaviour relative to PCA.

### 4.5.2 Representational similarity analysis

To complement these findings, shared structure between PCA or (k)SCA-represented behavioural and neural data was additionally assessed using representational similarity analysis (RSA; Kriegeskorte et al., 2008). RSA characterises representations as the dissimilarity structure of activity patterns, assessed in

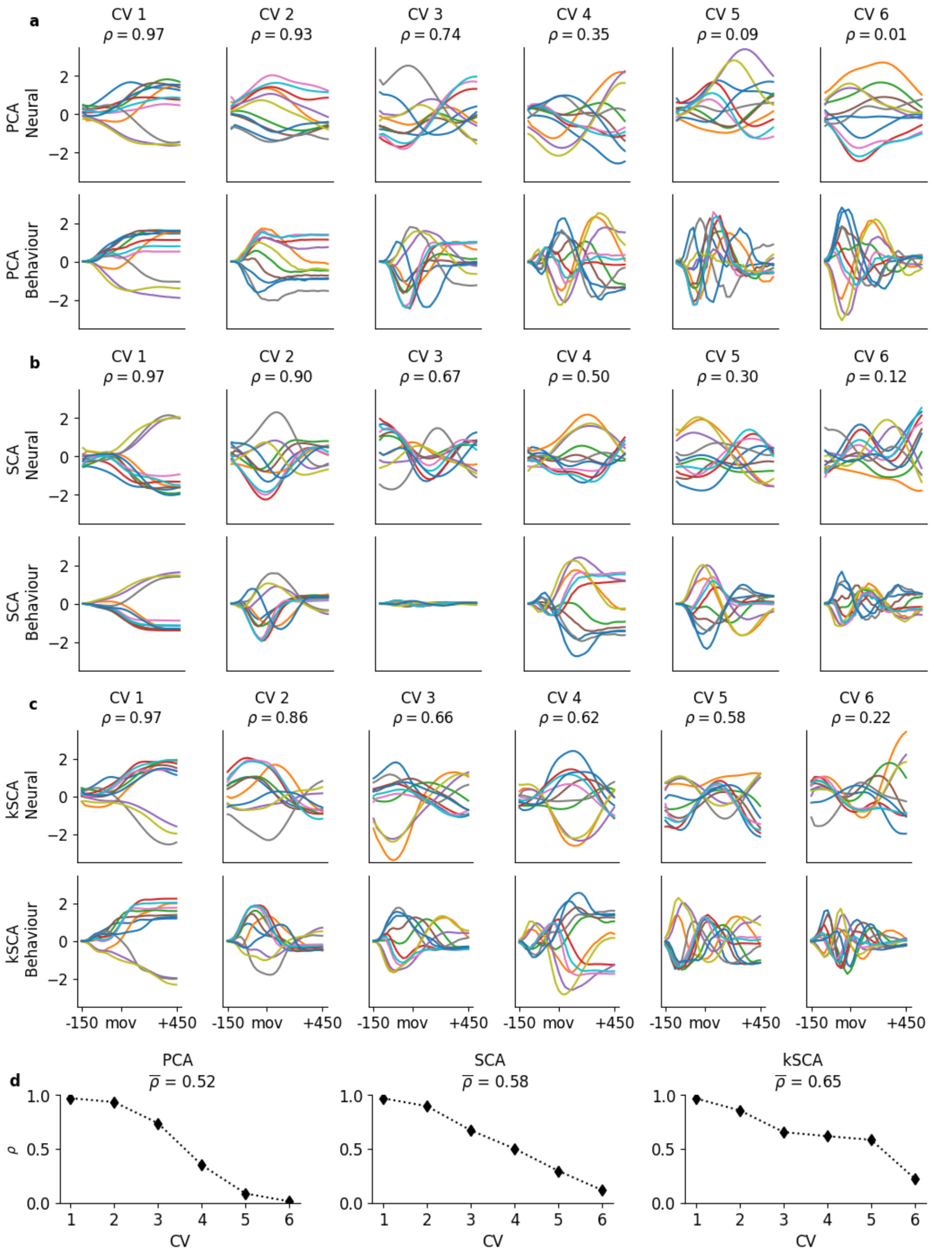

Figure 10: **Canonical correlation analysis for PCA, SCA, and kSCA**. (a) Time-varying canonical variables for 11 example conditions (randomly selected) for 6-dimensional PCA projections of neural PSTHs (upper row) and of augmented behaviour (lower row). Each column denotes a set of canonical variables, from the most (CV1) to the least (CV6) correlated ones. The canonical correlation $\rho$ is indicated. Time is from 150 ms prior to movement onset to 450 ms after, corresponding to the movement timing after accounting for a 100 ms lag. (b) Same for SCA. (c) Same for kSCA. (d) Full spectrum of canonical correlations $\rho$, along with the average across canonical variables $\overline{\rho}$.

terms of representational dissimilarity matrices (RDMs). These are $K \times K$ square symmetric matrices, where each off-diagonal entry is the dissimilarity between the spatial patterns elicited by two different conditions. On-diagonal elements denote comparisons for the same condition, and are thus equal to 0. Here, the spatial patterns are the $d$-dimensional vectors obtained from either PCA or (k)SCA. $d$ was again set to 6. We used 1 minus the Pearson correlation coefficient to measure the distance between the patterns (see Kao et al., 2021 for an analogous, correlation-based approach). Different RDMs were obtained for each time step, thereby allowing to probe the temporal evolution of representations in a more straightforward fashion relative to regression and CCA.

The Pearson correlation $r$ between the flattened RDMs of spatially projected neural data and augmented behaviour was computed at each time step (with a 100 ms lag). The results are displayed in Figure 11a. Notably, after movement was initiated, neural and behavioural RDMs became more correlated for (k)SCA relative to PCA, an effect that persisted for a few hundreds of ms. This is also shown in Figure 11b, which plots the RDMs 150 ms after movement onset. For behaviour, all dimensionality reduction approaches appeared to have yielded representations that discriminated between conditions in an almost binary fashion. At the neural level, this was also true for SCA and kSCA, but to a lesser extent for PCA, thereby explaining a lower $r = 0.68$ relative to $r = 0.76$ (SCA) and $r = 0.78$ (kSCA). Notably, these binary-like representational properties were already hinted by the 2-wing nature of the (k)SCA projections in Figure 6a, absent for PCA. Processing neural data using (k)SCA thus appears to better reveal the behaviourally relevant aspects of population activity.

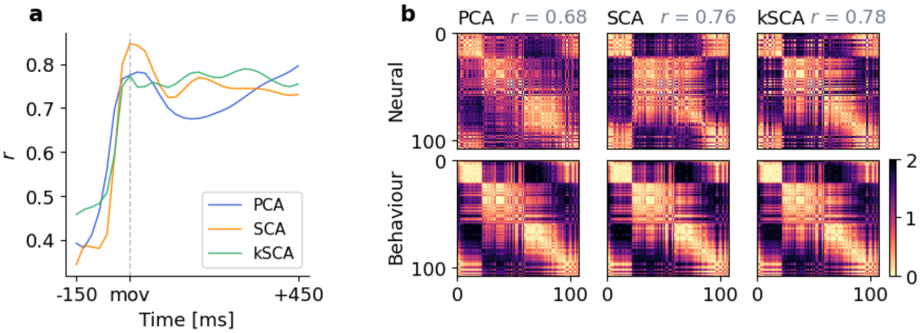

Figure 11: **Representational similarity analysis for PCA, SCA, and kSCA**. (a) Temporal evolution of the Pearson correlation coefficient $r$ between flattened RDMs of spatially-projected neural PSTHs and augmented behaviour, with a lag of 100 ms. (b) Neural (upper row) and behavioural (lower row) RDMs 150 ms after movement onset for PCA (left), SCA (middle) and kSCA (right) projections. The Pearson correlation $r$ between the flattened RDMs is indicated. For visibility, the conditions have been sorted according to reach direction, heuristically-derived from the angle at the end point of hand position.

## 4.6 Discussion

There exists convincing evidence that the motor system acts as an autonomous pattern generator, at least for the control of certain movements such as reaching (but see Suresh et al., 2020). The results in Section 4.3 further highlight the dynamical representational properties of the motor cortex. Yet, accepting a computation through dynamics view of motor function (Pandarinath et al., 2018a) will ultimately require deepening our understanding of how movement arises downstream of population dynamics.

To address this, we first conducted regression-based analyses. Behavioural predictivity of joint torques was generally greater for (k)SCA-represented neural data relative to PCA. Moreover, CCA revealed a resemblance between the most sequential components of neural responses and of behaviour, which persisted across canonical variables. RSA additionally showed that (k)SCA promoted greater resemblance between the RDM-indicated representational geometry of neural and behavioural responses. Overall, these results are consistent with a matching between the structure of neural and behavioural activity. Notably, such an isomorphism supports that the neural code produced by a dynamical motor system is adapted to the resulting behaviour.

Note that, unlike other existing methods (Sani et al., 2021), (k)SCA learned neural dynamics that were relevant w.r.t. the measured behavioural variables despite the absence of behavioural learning signals. This is essential, helping to ensure that (k)SCA can be applied even in the case of little behavioural correlates (e.g., evidence accumulation during decision-making, hippocampal system reactivation during sleep).

It is also worth noting that previous methods have attained greater behavioural predictivity, for example on single trials, than the results presented here (Pandarinath et al., 2018a; Schimel et al., 2022). This was the case for latent factor analysis via dynamical systems (LFADS), a sequential variational autoencoder, whose latent representations even surpassed population activity when predicting hand kinematics. In LFADS, a recurrent neural network (RNN) provides a generative model of spiking activity, assumed to emanate from low-dimensional dynamics (Pandarinath et al., 2018a). The variational lower bound on the log-likelihood of the observed neural activity given the rates predicted by the model is then maximised. Yet, while highly powerful, such deep learning-based techniques are computationally intensive. Therefore, (k)SCA could be preferable for problems that do not warrant a deep learning approach.

# 5 Conclusion and future work

## 5.1 Summary

Improving our understanding of dynamical computation will ultimately necessitate to identify the dynamical rule that determines the temporal evolution of neuronal activity. This can be achieved using latent dynamical models, which provide differential equations that model the latent state in terms of linear (Kao et al., 2015) or nonlinear (Wang et al., 2005) dynamics. Deep RNN-based models such as LFADS (Pandarinath et al., 2018a) also facilitate dynamical system identification. The features of the dynamics, which can be illustrated with a flow field, then provide insight into circuit computation. In contrast, while dimensionality reduction approaches can provide latents that estimate the dynamical state, they typically do not allow to statistical identify the underlying computational dynamics (Duncker & Sahani, 2021). (k)SCA (and previously GPFADS; Rutten et al., 2020a) contribute to address this limitation, as they yield low-dimensional embeddings most consistent with dynamics. Specifically, (k)SCA focuses on identifying and leveraging temporal non-reversibility, ubiquitous in state-space trajectories governed by lawful flow fields (at least for a largely non-input-driven system). Additionally, kSCA allows for the identification of nonlinear subspaces, a yet absent property of most existing dimensionality reduction approaches exploring neural dynamics (Churchland et al., 2012; Lara et al., 2018; Mackevicius et al., 2019; Rutten et al., 2020a; Sani et al., 2021; but see Low et al., 2018; Nieh et al., 2021).

The analyses on synthetic datasets demonstrated the utility of the non-reversibility-based learning goal in SCA, as well as the nonlinear flexibility conferred by kSCA. When applied to motor cortical data, (k)SCA discovered rotational dynamics that appeared to better predict behaviour relative to PCA, consistent with the proposal that population dynamics underpin movement generation (Churchland et al., 2012). Moreover, CCA and RSA suggested a structural mapping between the most sequential components of neural activity and of behaviour. This supports that the dynamic nature of the motor control neural code allows it to be adapted to the corresponding movements. Together, these results illustrate the potential of (k)SCA in addressing questions about representations and computations in neural populations.

## 5.2 Directions for future work

The kSCA experiments reported in this thesis relied on the RBF kernel. Yet, this kernel is not necessarily the most appropriate for all problems relevant to kCSA. For example, it might be necessary to incorporate periodicity when investigating neural circuits such as those supporting the head direction system, where tuning curves exhibit cyclic boundary conditions. More generally, the optimal kernel type is not necessarily known in advance. To enhance expressivity, the non-parametric, and hence, flexible nature of kernel approaches could be combined with neural networks, which can act as universal function approximators (Cybenko, 1989) to uncover meaningful representations. This idea constitutes the basis of deep kernel learning, where the inputs to the kernel (e.g., RBF) are the outputs of deep models such as convolutional neural networks (Wilson et al., 2016). The neural network parameters are optimised jointly with the kernel parameters (e.g., RBF length-scale). In the original deep kernel learning framework, which was applied to Gaussian process regression problems, this was done through the marginal likelihood. In the kSCA setting, the parameters could instead maximise the sequentiality objective. Conveniently, scalability can also be improved using inducing point approximation in deep kernel learning (Wilson et al., 2016).

It is also worth noting that the inner products in the (k)SCA objective reduced the spatial dimension. Alternatively, the temporal dimension could be targeted. The cyclic property of the trace indeed allows to swap elements to yield $T \times T$ matrices in equations 5 and 6. Therefore, the kSCA algorithm can readily be capitalised upon for kernelising in time. Future work could explore potential applications, such as smoothing neural spikes prior to dimensionality reduction.

Moreover, the analyses applied to a neuroscience context focused on the motor system. This made sense since motor-related activity is tied to behavioural signals (e.g., hand kinematics), thereby promoting investigation of the link between neural activity and behaviour. However, in the future, (k)SCA could prove useful in other brain areas, such as decision-making ones. In reaction time variants of the random-dot motion paradigm, accuracy and reaction time tend to increase as a function of the difficulty of the task, such that higher motion coherence yields more accurate, faster choices. This is consistent with accumulation of noisy evidence up to a decision threshold (Gold & Shadlen, 2007). Notably, neural correlates of this behaviour appear to confine to a lower-dimensional space, where activity associated with different saccade choices grows towards distinct directions with differing magnitudes according to evidence strength (Mante et al., 2013). (k)SCA could potentially discover such a subspace, and could be leveraged to further reveal its properties (e.g., clear identification of the decision threshold).

Another potential neuroscience application of (k)SCA is to the navigation system. In addition to the precisely ordered hippocampal place cell spike sequences mentioned in the Introduction, other spatially-tuned neuronal populations have been proposed to exhibit dynamical structure. Attractors (i.e., involving multiple stable states) were used to describe head-direction cells, which indicate direction, as well as grid cells, which support self-motion integration and trajectory planning (Chaudhuri et al., 2019; McNaughton et al., 2006; Seelig & Jayaraman, 2015). Extracting non-reversible latents from navigation-related activity could thus yield meaningful trajectories. Analysing their spatio-temporal organisation could potentially help to disambiguate the navigation neural code, a challenging endeavour, for example since the hippocampus encodes a wide range of variables including position-, but also sensory- and decision-related ones (Bimbard et al., 2023; Nieh et al., 2021). Moreover, navigation-related activity appears to be constrained to nonlinear subspaces. This was true for the head-direction circuit (Chaudhuri et al., 2019; Jensen et al., 2020) as well as for place cells (Low et al., 2018; Nieh et al., 2021). Therefore, in contrast with the motor system, a nonlinear, kSCA-based approach might become essential.

Importantly, for the above systems, analyses on a trial-by-trial basis might become crucial given substantial variations across trials in decision-making and navigation tasks (Low et al., 2018; Nieh et al., 2021; Yu et al., 2009), preventing to average responses across trials. Notably, the learning goal inherent to PCA appears particularly unsuitable when confronted with single trial noise. PCA does not distinguish between maximising the variance shared between neurons from the variance independent across neurons (Kao et al., 2015; Pandarinath et al., 2018b; Yu et al., 2009), thus possibly resulting in a poor characterisation of the intrinsic manifold. By contrast, in seeking non-reversible latents consistent with a lawful flow field, (k)SCA restricts the trajectories that can be identified (e.g., clockwise rotating trajectories cannot unexpectedly reverse to counterclockwise, and vice-versa; Kao et al., 2015), potentially enhancing robustness to trajectory-deviating noise. Single-trial analyses (beyond those in Section 4) that directly evaluate this claim, could constitute an interesting avenue for future work.

## 5.3 Conclusion

Large-scale neural recordings are being increasingly prevalent to study the circuit dynamics underlying behaviour (Jun et al., 2017; Siegle et al., 2021). Population-level findings have the potential to offer qualitatively different insight relative to single-neuron responses (Chaudhuri et al., 2019; Elsayed & Cunningham, 2017; Pandarinath et al., 2018b). (k)SCA complements existing neural data analysis approaches that interpret neural population activity as evolving on a lower-dimensional manifold (Churchland et al., 2012; Low et al., 2018; Nieh et al., 2021; Pandarinath et al., 2018a; Schimel et al., 2022), with a focus on extracting non-reversible latents. This thesis leveraged (k)SCA to characterise the neural dynamics underlying motor control, though the method holds promise in probing collective dynamics in a wider range of (neural) systems involving non-reversible time-series.

# References

Acharya, L., Aghajan, Z. M., Vuong, C., Moore, J. J., & Mehta, M. R. (2016). Causal influence of visual cues on hippocampal directional selectivity. *Cell*, *164*(1), 197–207.

Berlinet, A., & Thomas-Agnan, C. (2004). *Reproducing kernel hilbert spaces in probability and statistics*. Springer Science & Business Media.

Bimbard, C., Sit, T. P., Lebedeva, A., Reddy, C. B., Harris, K. D., & Carandini, M. (2023). Behavioral origin of sound-evoked activity in mouse visual cortex. *Nature Neuroscience*, *26*(2), 251–258.

Bishop, C. M., & Nasrabadi, N. M. (2006). *Pattern recognition and machine learning* (Vol. 4). Springer.

Bolt, T., Nomi, J. S., Bzdok, D., Salas, J. A., Chang, C., Thomas Yeo, B., Uddin, L. Q., & Keilholz, S. D. (2022). A parsimonious description of global functional brain organization in three spatiotemporal patterns. *Nature Neuroscience*, *25*(8), 1093–1103.

Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., & Zhang, Q. (2018). *JAX: Composable transformations of Python+NumPy programs* (Version 0.3.13). http://github.com/google/jax

Chaudhuri, R., Gerçek, B., Pandey, B., Peyrache, A., & Fiete, I. (2019). The intrinsic attractor manifold and population dynamics of a canonical cognitive circuit across waking and sleep. *Nature Neuroscience*, *22*(9), 1512–1520.

Chiu, Y.-C., Esterman, M. S., Gmeindl, L., & Yantis, S. (2012). Tracking cognitive fluctuations with multivoxel pattern time course (mvptc) analysis. *Neuropsychologia*, *50*(4), 479–486.

Churchland, M. M., Cunningham, J. P., Kaufman, M. T., Foster, J. D., Nuyujukian, P., Ryu, S. I., & Shenoy, K. V. (2012). Neural population dynamics during reaching. *Nature*, *487*(7405), 51–56.

Churchland, M. M., Cunningham, J. P., Kaufman, M. T., Ryu, S. I., & Shenoy, K. V. (2010). Cortical preparatory activity: Representation of movement or first cog in a dynamical machine? *Neuron*, *68*(3), 387–400.

Churchland, M. M., & Shenoy, K. V. (2007). Temporal complexity and heterogeneity of single-neuron activity in premotor and motor cortex. *Journal of Neurophysiology*, *97*(6), 4235–4257.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, *2*(4), 303–314.

DeepMind, Babuschkin, I., Baumli, K., Bell, A., Bhupatiraju, S., Bruce, J., Buchlovsky, P., Budden, D., Cai, T., Clark, A., Danihelka, I., Dedieu, A., Fantacci, C., Godwin, J., Jones, C., Hemsley, R., Hennigan, T., Hessel, M., Hou, S., . . . Viola, F. (2020). *The DeepMind JAX Ecosystem*. http://github.com/google-deepmind

Drieu, C., & Zugaro, M. (2019). Hippocampal sequences during exploration: Mechanisms and functions. *Frontiers in Cellular Neuroscience*, *13*, 232.

Duncker, L., & Sahani, M. (2021). Dynamics on the manifold: Identifying computational dynamical activity from neural population recordings. *Current Opinion in Neurobiology*, *70*, 163–170.

Elsayed, G. F., & Cunningham, J. P. (2017). Structure in neural population recordings: An expected byproduct of simpler phenomena? *Nature Neuroscience*, *20*(9), 1310–1318.

Galgali, A. R., Sahani, M., & Mante, V. (2023). Residual dynamics resolves recurrent contributions to neural computation. *Nature Neuroscience*, *26*(2), 326–338.

Gallego, J. A., Perich, M. G., Miller, L. E., & Solla, S. A. (2017). Neural manifolds for the control of movement. *Neuron*, *94*(5), 978–984.

Georgopoulos, A. P., Kalaska, J. F., Caminiti, R., & Massey, J. T. (1982). On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. *Journal of Neuroscience*, *2*(11), 1527–1537.

Gold, J. I., & Shadlen, M. N. (2002). Banburismus and the brain: Decoding the relationship between sensory stimuli, decisions, and reward. *Neuron*, *36*(2), 299–308.

Gold, J. I., & Shadlen, M. N. (2007). The neural basis of decision making. *Annual Review of Neuroscience*, *30*(1), 535–574.

Hanks, T. D., Mazurek, M. E., Kiani, R., Hopp, E., & Shadlen, M. N. (2011). Elapsed decision time affects the weighting of prior probability in a perceptual decision task. *Journal of Neuroscience*, *31*(17), 6339–6352.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., . . . Oliphant, T. E. (2020). Array programming with numpy. *Nature*, *585*(7825), 357–362.

Hennequin, G., Vogels, T. P., & Gerstner, W. (2014). Optimal control of transient dynamics in balanced networks supports generation of complex movements. *Neuron*, *82*(6), 1394–1406.

Hofmann, T., Schölkopf, B., & Smola, A. J. (2008). Kernel methods in machine learning. *The Annals of Statistics*, *36*(3).

Jeffery, K. J., Wilson, J. J., Casali, G., & Hayman, R. M. (2015). Neural encoding of large-scale three-dimensional space—properties and constraints. *Frontiers in Psychology*, *6*, 927.

Jensen, K., Kao, T.-C., Tripodi, M., & Hennequin, G. (2020). Manifold gplvms for discovering non-euclidean latent structure in neural data. *Advances in Neural Information Processing Systems*, *33*, 22580–22592.

Jun, J. J., Steinmetz, N. A., Siegle, J. H., Denman, D. J., Bauza, M., Barbarits, B., Lee, A. K., Anastassiou, C. A., Andrei, A., Aydın, Ç., et al. (2017). Fully integrated silicon probes for high-density recording of neural activity. *Nature*, *551*(7679), 232–236.

Kao, J. C., Nuyujukian, P., Ryu, S. I., Churchland, M. M., Cunningham, J. P., & Shenoy, K. V. (2015). Single-trial dynamics of motor cortex and their applications to brain-machine interfaces. *Nature Communications*, *6*(1), 7759.

Kao, T.-C., Sadabadi, M. S., & Hennequin, G. (2021). Optimal anticipatory control as a theory of motor preparation: A thalamo-cortical circuit model. *Neuron*, *109*(9), 1567–1581.

Kriegeskorte, N., Mur, M., & Bandettini, P. A. (2008). Representational similarity analysis-connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, *2*, 249.

Kuzmina, E., Kriukov, D., & Lebedev, M. (2024). Neuronal travelling waves explain rotational dynamics in experimental datasets and modelling. *Scientific Reports*, *14*(1), 3566.

Lara, A. H., Cunningham, J. P., & Churchland, M. M. (2018). Different population dynamics in the supplementary motor area and motor cortex during reaching. *Nature Communications*, *9*(1), 2754.

Lebedev, M. A., Ossadtchi, A., Mill, N. A., Urpí, N. A., Cervera, M. R., & Nicolelis, M. A. (2019). Analysis of neuronal ensemble activity reveals the pitfalls and shortcomings of rotation dynamics. *Scientific Reports*, *9*(1), 18978.

Lee, A. K., & Wilson, M. A. (2002). Memory of sequential experience in the hippocampus during slow wave sleep. *Neuron*, *36*(6), 1183–1194.

Li, W., & Todorov, E. (2004). Iterative linear quadratic regulator design for nonlinear biological movement systems. *First International Conference on Informatics in Control, Automation and Robotics*, *2*, 222–229.

Low, R. J., Lewallen, S., Aronov, D., Nevers, R., & Tank, D. W. (2018). Probing variability in a cognitive map using manifold inference from neural dynamics. *BioRxiv*, 418939.

Lynch, G. F., Okubo, T. S., Hanuschkin, A., Hahnloser, R. H., & Fee, M. S. (2016). Rhythmic continuous-time coding in the songbird analog of vocal motor cortex. *Neuron*, *90*(4), 877–892.

Mackevicius, E. L., Bahle, A. H., Williams, A. H., Gu, S., Denisenko, N. I., Goldman, M. S., & Fee, M. S. (2019). Unsupervised discovery of temporal sequences in high-dimensional datasets, with applications to neuroscience. *Elife*, *8*, e38471.

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

Mastrogiuseppe, F., & Ostojic, S. (2018). Linking connectivity, dynamics, and computations in low-rank recurrent neural networks. *Neuron*, *99*(3), 609–623.

McNaughton, B. L., Battaglia, F. P., Jensen, O., Moser, E. I., & Moser, M.-B. (2006). Path integration and the neural basis of the'cognitive map'. *Nature Reviews Neuroscience*, *7*(8), 663–678.

Mi, D., Li, Z., Lim, L., Li, M., Moissidis, M., Yang, Y., Gao, T., Hu, T. X., Pratt, T., Price, D. J., et al. (2018). Early emergence of cortical interneuron diversity in the mouse embryo. *Science*, *360*(6384), 81–85.

Michaels, J. A., Dann, B., & Scherberger, H. (2016). Neural population dynamics during reaching are better explained by a dynamical system than representational tuning. *PLoS Computational Biology*, *12*(11), e1005175.

Nádasdy, Z., Hirase, H., Czurkó, A., Csicsvari, J., & Buzsáki, G. (1999). Replay and time compression of recurring spike sequences in the hippocampus. *Journal of Neuroscience*, *19*(21), 9497–9507.

Navratilova, Z., Hoang, L. T., Schwindel, C. D., Tatsuno, M., & McNaughton, B. L. (2012). Experience-dependent firing rate remapping generates directional selectivity in hippocampal place cells. *Frontiers in Neural Circuits*, *6*, 6.

Newsome, W. T., Britten, K. H., & Movshon, J. A. (1989). Neuronal correlates of a perceptual decision. *Nature*, *341*(6237), 52–54.

Nieh, E. H., Schottdorf, M., Freeman, N. W., Low, R. J., Lewallen, S., Koay, S. A., Pinto, L., Gauthier, J. L., Brody, C. D., & Tank, D. W. (2021). Geometry of abstract learned knowledge in the hippocampus. *Nature*, *595*(7865), 80–84.

Ólafsdóttir, H. F., Barry, C., Saleem, A. B., Hassabis, D., & Spiers, H. J. (2015). Hippocampal place cells construct reward related sequences through unexplored space. *eLife*, *4*, e06063.

Pandarinath, C., Ames, K. C., Russo, A. A., Farshchian, A., Miller, L. E., Dyer, E. L., & Kao, J. C. (2018b). Latent factors and dynamics in motor cortex and their application to brain–machine interfaces. *Journal of Neuroscience*, *38*(44), 9390–9401.

Pandarinath, C., Gilja, V., Blabe, C. H., Nuyujukian, P., Sarma, A. A., Sorice, B. L., Eskandar, E. N., Hochberg, L. R., Henderson, J. M., & Shenoy, K. V. (2015). Neural population dynamics in human motor cortex during movements in people with als. *eLife*, *4*, e07436.

Pandarinath, C., O'Shea, D. J., Collins, J., Jozefowicz, R., Stavisky, S. D., Kao, J. C., Trautmann, E. M., Kaufman, M. T., Ryu, S. I., Hochberg, L. R., et al. (2018a). Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature Methods*, *15*(10), 805–815.

Paninski, L., Shoham, S., Fellows, M. R., Hatsopoulos, N. G., & Donoghue, J. P. (2004). Superlinear population encoding of dynamic hand trajectory in primary motor cortex. *Journal of Neuroscience*, *24*(39), 8551–8561.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *the Journal of Machine Learning Research*, *12*, 2825–2830.

Pei, F., Ye, J., Zoltowski, D., Wu, A., Chowdhury, R. H., Sohn, H., O'Doherty, J. E., Shenoy, K. V., Kaufman, M. T., Churchland, M., et al. (2021). Neural latents benchmark'21: Evaluating latent variable models of neural population activity. *arXiv*.

Pfenning, A. R., Hara, E., Whitney, O., Rivas, M. V., Wang, R., Roulhac, P. L., Howard, J. T., Wirthlin, M., Lovell, P. V., Ganapathy, G., et al. (2014). Convergent transcriptional specializations in the brains of humans and song-learning birds. *Science*, *346*(6215), 1256846.

Picardo, M. A., Merel, J., Katlowitz, K. A., Vallentin, D., Okobi, D. E., Benezra, S. E., Clary, R. C., Pnevmatikakis, E. A., Paninski, L., & Long, M. A. (2016). Population-level representation of a temporal sequence underlying song production in the zebra finch. *Neuron*, *90*(4), 866–876.

Quinonero-Candela, J., & Rasmussen, C. E. (2005). A unifying view of sparse approximate gaussian process regression. *the Journal of Machine Learning Research*, *6*, 1939–1959.

Raghu, M., Gilmer, J., Yosinski, J., & Sohl-Dickstein, J. (2017). Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. *Advances in Neural Information Processing Systems*, *30*.

Rahimi, A., & Recht, B. (2007). Random features for large-scale kernel machines. *Advances in Neural Information Processing Systems*, *20*.

Rasmussen, C. E., & Williams, C. K. (2006). *Gaussian processes for machine learning*. MIT press Cambridge, MA.

Roitman, J. D., & Shadlen, M. N. (2002). Response of neurons in the lateral intraparietal area during a combined visual discrimination reaction time task. *Journal of Neuroscience*, *22*(21), 9475–9489.

Russo, A. A., Bittner, S. R., Perkins, S. M., Seely, J. S., London, B. M., Lara, A. H., Miri, A., Marshall, N. J., Kohn, A., Jessell, T. M., et al. (2018). Motor cortex embeds muscle-like commands in an untangled population response. *Neuron*, *97*(4), 953–966.

Rutten, V., Bernacchia, A., & Hennequin, G. (2020b). Sequential components analysis. *Cosyne, Denver, CO*, *T-22*.

Rutten, V., Bernacchia, A., Sahani, M., & Hennequin, G. (2020a). Non-reversible gaussian processes for identifying latent dynamical structure in neural data. *Advances in Neural Information Processing Systems*, *33*, 9622–9632.

Sani, O. G., Abbaspourazad, H., Wong, Y. T., Pesaran, B., & Shanechi, M. M. (2021). Modeling behaviorally relevant neural dynamics enabled by preferential subspace identification. *Nature Neuroscience*, *24*(1), 140–149.

Saxena, S., & Cunningham, J. P. (2019). Towards the neural population doctrine. *Current Opinion in Neurobiology*, *55*, 103–111.

Schimel, M., Kao, T.-C., Jensen, K. T., & Hennequin, G. (2022). Ilqr-vae: Control-based learning of input-driven dynamics with applications to neural data. *International Conference on Learning Representations*.

Scholkopf, B., & Smola, A. J. (2002). *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT press.

Seelig, J. D., & Jayaraman, V. (2015). Neural dynamics for landmark orientation and angular path integration. *Nature*, *521*(7551), 186–191.

Shi, Y., Wang, M., Mi, D., Lu, T., Wang, B., Dong, H., Zhong, S., Chen, Y., Sun, L., Zhou, X., et al. (2021). Mouse and human share conserved transcriptional programs for interneuron development. *Science*, *374*(6573), eabj6641.

Siegle, J. H., Ledochowitsch, P., Jia, X., Millman, D. J., Ocker, G. K., Caldejon, S., Casal, L., Cho, A., Denman, D. J., Durand, S., et al. (2021). Reconciling functional differences in populations of neurons recorded with two-photon imaging and electrophysiology. *eLife*, *10*, e69068.

Silverman, B. W. (1985). Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society: Series B (Methodological)*, *47*(1), 1–21.

Skaggs, W. E., & McNaughton, B. L. (1996). Replay of neuronal firing sequences in rat hippocampus during sleep following spatial experience. *Science*, *271*(5257), 1870–1873.

Stachenfeld, K. L., Botvinick, M. M., & Gershman, S. J. (2017). The hippocampus as a predictive map. *Nature Neuroscience*, *20*(11), 1643–1653.

Suresh, A. K., Goodman, J. M., Okorokova, E. V., Kaufman, M., Hatsopoulos, N. G., & Bensmaia, S. J. (2020). Neural population dynamics in motor cortex are different for reach and grasp. *eLife*, *9*, e58848.

Sussillo, D., Churchland, M. M., Kaufman, M. T., & Shenoy, K. V. (2015). A neural network that finds a naturalistic solution for the production of muscle activity. *Nature Neuroscience*, *18*(7), 1025–1033.

Wang, J., Hertzmann, A., & Fleet, D. J. (2005). Gaussian process dynamical models. *Advances in Neural Information Processing Systems*, *18*.

Wang, W., Dang, Z., Hu, Y., Fua, P., & Salzmann, M. (2019). Backpropagation-friendly eigendecomposition. *Advances in Neural Information Processing Systems*, *32*.

Wilson, A., & Nickisch, H. (2015). Kernel interpolation for scalable structured gaussian processes (kiss-gp). *International Conference on Machine Learning*, 1775–1784.

Wilson, A. G., Hu, Z., Salakhutdinov, R., & Xing, E. P. (2016). Deep kernel learning. *Artificial Intelligence and Statistics*, 370–378.

Yu, B. M., Cunningham, J. P., Santhanam, G., Ryu, S., Shenoy, K. V., & Sahani, M. (2009). Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. *Journal of Neurophysiology*, *102*, 614–635.

# A Mathematical derivation of the SCA algorithm

The following computes each term of second-to-last line in equation 5 to prove that it evaluates to the sequentiality objective used throughout this thesis

$$\begin{aligned}
\mathbb{E}_{v\sim\mathcal{N}(0,I)}[\mathrm{Tr}(Y_kV^\top)\mathrm{Tr}(Y_{k'}V^\top)\mathrm{Tr}(Y_kY_{k'}^\top)] &= \mathbb{E}[(y^\top v)(v^\top z)]\mathrm{Tr}(Y_kY_{k'}^\top) \\
&= (y^\top\mathbb{E}[vv^\top]z)\mathrm{Tr}(Y_kY_{k'}^\top) \\
&= (y^\top Iz)\mathrm{Tr}(Y_kY_{k'}^\top) \\
&= \mathrm{Tr}^2(Y_kY_{k'}^\top)
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}[\mathrm{Tr}(Y_kV^\top)\mathrm{Tr}(Y_kY_{k'}^\top VY_{k'}^\top)] &= \mathbb{E}[(y^\top v)\mathrm{Tr}(Y_{k'}^\top Y_kY_{k'}^\top V)] \\
&= \mathbb{E}[(y^\top v)(v^\top\mathrm{vec}(Y_{k'}^\top Y_kY_{k'}^\top)] \\
&= y^\top\mathbb{E}[vv^\top]\mathrm{vec}(Y_{k'}^\top Y_kY_{k'}^\top) \\
&= y^\top\mathrm{vec}(Y_{k'}^\top Y_kY_{k'}^\top) \\
&= \mathrm{Tr}(Y_kY_{k'}^\top Y_kY_{k'}^\top)
\end{aligned}$$

$$\mathbb{E}[\mathrm{Tr}(Y_{k'}V^\top)\mathrm{Tr}(Y_kV^\top Y_kY_{k'}^\top)] = \mathrm{Tr}(Y_kY_{k'}^\top Y_kY_{k'}^\top)$$

$$\begin{aligned}
\mathbb{E}[\mathrm{Tr}(Y_kV^\top Y_kY_{k'}^\top VY_{k'}^\top)] &= \mathbb{E}[\mathrm{Tr}(V^\top Y_kY_{k'}^\top VY_{k'}^\top Y_k)] \\
&= \mathbb{E}[v^\top\mathrm{vec}(Y_kY_{k'}^\top VY_{k'}^\top Y_k)] \\
&= \mathbb{E}[v^\top[(Y_k^\top Y_{k'})\otimes(Y_kY_{k'}^\top)]v] \\
&= \mathrm{Tr}[(Y_k^\top Y_{k'})\otimes(Y_kY_{k'}^\top)] \\
&= \mathrm{Tr}[Y_k^\top Y_{k'}]\mathrm{Tr}[Y_kY_{k'}^\top] \\
&= \mathrm{Tr}^2(Y_kY_{k'}^\top)
\end{aligned}$$

which uses the cyclic nature of the trace, $\mathrm{Tr}(AB) = \mathrm{vec}(A^\top)^\top\mathrm{vec}(B)$, $\mathrm{vec}(AXB) = (B^\top\otimes A)\mathrm{vec}(X)$, as well as $\mathrm{Tr}(A\otimes B) = \mathrm{Tr}(A)\mathrm{Tr}(B)$.

# B Additional synthetic dataset to evaluate kSCA versus SCA: drift diffusion model

Another example highlighting SCA/kSCA performance differences relates to the progressive integration of noisy evidence during decision making, for example during the random-dot motion task (see Introduction). This process can be described by the drift diffusion model (DDM) of behaviour (Gold & Shadlen, 2002). In the continuous time limit, it is given by the stochastic differential equation

$$dx = \mu_k dt + \sigma dW \tag{22}$$

where $x$ is the decision variable (DV), whose temporal evolution reflects the accumulation of evidence. $\mu_k$ denotes the drift rate, with greater values implying higher stimulus strength, or motion coherence. $dW$ is a standard Wiener process, such that its variance grows linearly with time. $\sigma$ scales this diffusive process. Without the $\sigma dW$ diffusion term, the DV would grow linearly with slope $\mu_k$. Conversely, the absence of the $\mu_k$ term would result in a Brownian motion-like trajectory, with standard deviation proportional to $\sqrt{t}$. The stochastic diffusion in equation 22 can be simulated using

$$x(t+1) = x(t) + \mu_k dt + \sigma\sqrt{dt}\mathcal{N}(0,1) \tag{23}$$

where $\mathcal{N}(0,1)$ indicates a random number with mean 0 and standard deviation 1, generated anew at each time step.

Here, we generated trajectories with motion coherence $\mu_k \in \pm\{0.0, 0.02, 0.04, 0.08, 0.16, 0.32, 0.64\}$ (i.e., 13 in total), where the sign models the direction of motion (e.g., leftward versus rightward). Each motion coherence level was repeated for 30 trials (20 training, 10 test). $\sigma$ was set to 0.35 to match empirical psychometric functions (Gold & Shadlen, 2007). There were $T = 100$ time steps. The trial and condition (i.e., motion coherence) dimensions were collapsed to allow for single trial-based analysis. The dataset was then projected onto a 50-dimensional orthonormal basis, before being fed to a relu function, thereby providing a (simplistic) model of LIP activity during evidence integration. Training was performed in a similar fashion as that described in Section 3. While PCA and SCA yielded $\zeta = 0.0$ on both the training and the test sets, kSCA achieved $\zeta = 0.26$ (training) and $\zeta = 0.24$ (test). This supports the ability of kSCA to identify sequentiality in inherently 1-dimensional, expansion-like data. Moreover, though preliminary, the results suggest that (k)SCA could prove useful to examine the neural correlates of decision-making (see also Discussion). Note however that real neural datasets are expected to differ in important aspects. These include the termination of the increase in neural activity when a putative decision threshold is reached (versus uninterrupted growth here).

# C Losses

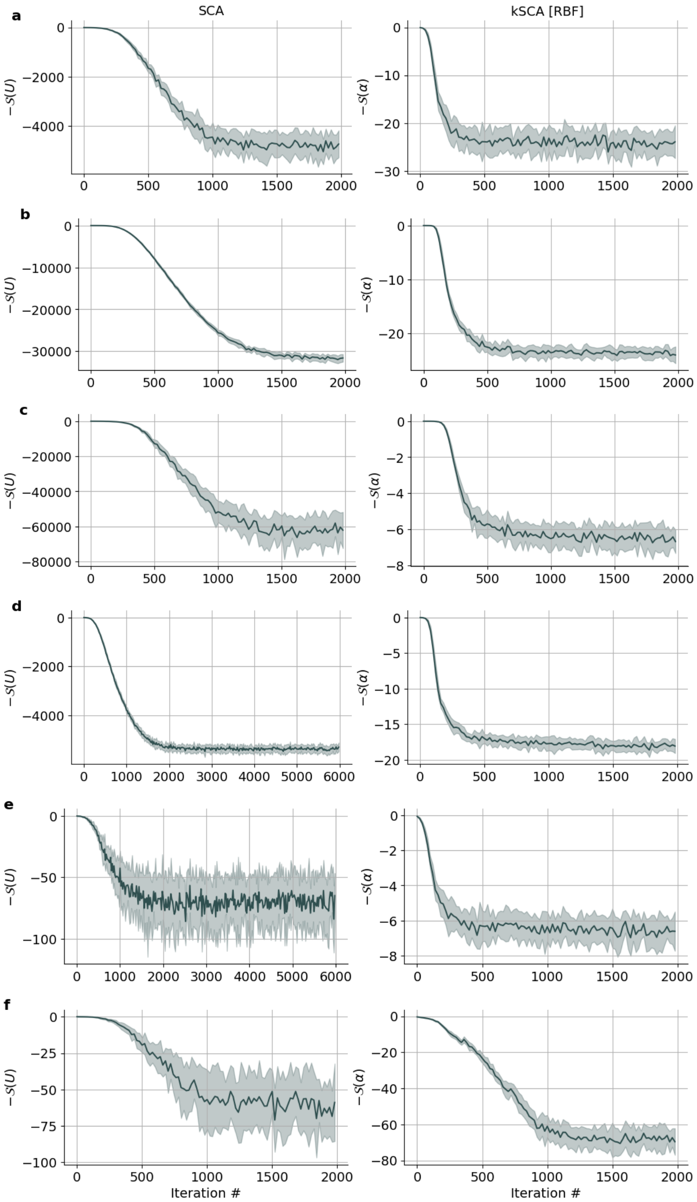

Figure C.1: **Losses corresponding to the learned projections in Section 3**. (a) SCA (left) and kSCA (right) losses for the rotation dataset in Figure 2. Inspection of the (k)SCA losses confirms that convergence had been reached. For illustration purposes, the mean losses across 20 consecutive iterations are displayed, with standard deviations denoted by the fill. (b) Same for the Van der Pol oscillator (Figure 3a). (c) Same for the Duffing oscillator (Figure 3b). (d) Same for the particle (Figure 4). (e), (f) Same for the polar rotations converted to the Cartesian domain (Figure 5), with (e) and without (f) noise corruption.

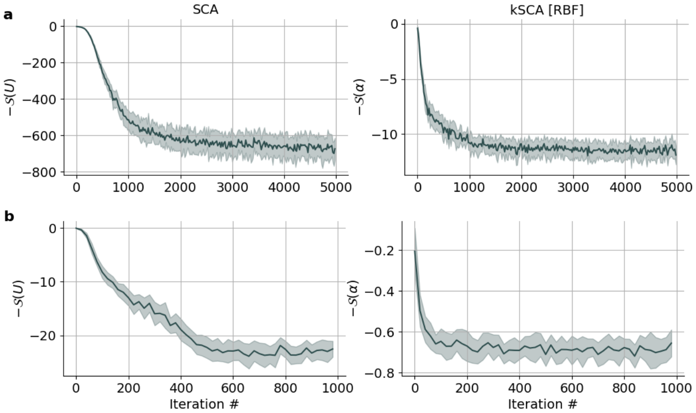

Figure C.2: **Losses corresponding to the learned projections in 6 dimensions in Section 4**. (a) Losses for the PSTHs-extracted projections in Figure 6a. (b) Losses for the single trial-extracted projections in Figure 7a. Analogous outcomes (i.e., convergence) were obtained for the other *d* settings. Mean losses across 20 consecutive iterations are displayed, with standard deviations denoted by the fill.