SIMPL: Scalable and hassle-free optimization of neural representations from behaviour

Summary High-dimensional neural activity in the brain is known to encode low-dimensional, time-evolving, behaviour-related latent variables. A fundamental goal of neural data analysis consists of identifying such variables and their mapping to neural activity. The canonical approach is to assume the latent variables *are* behaviour (e.g. the measured location of the animal) and visualise the subsequent tuning curves. However, significant mismatches between behaviour and the encoded variables may still exist—the agent may be thinking of another location, or be uncertain of its own—distorting the tuning curves and decreasing their interpretability. To address this issue a variety of methods have been proposed to learn this latent variable in an unsupervised manner; these techniques are typically expensive to train, come with many hyperparameters, or scale poorly to large datasets complicating their adoption in practice. To solve these issues we propose SIMPL (Scalable Iterative Maximization of Population-coded Latents); an EM-style algorithm which iteratively optimises latent variables and tuning curves. SIMPL is fast, scalable and exploits behaviour as an initial condition to improve convergence and identifiability. SIMPL accurately recovers latent variables in a biologically-inspired spatial navigation task, outperforming a contemporary neural-network based equivalent. When applied to a large rodent hippocampal dataset SIMPL rapidly finds a modified latent space with smaller, more numerous, and more uniformly-sized place fields than those based on behaviour, suggesting the brain encodes space with greater resolution than previously thought.

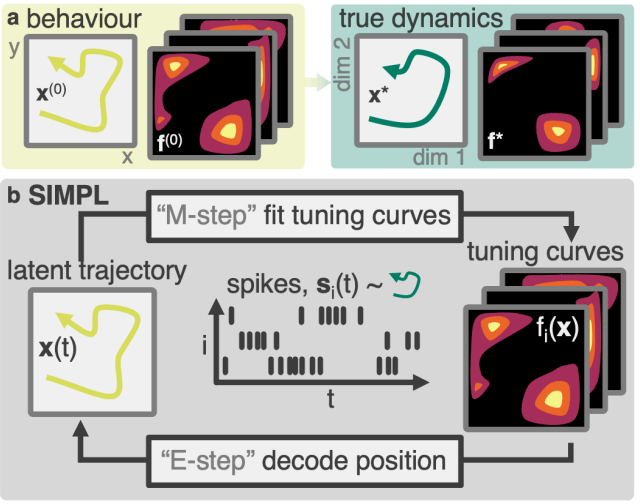

Figure 1: SIMPL **(a)** Animal behaviour—and tuning curves fit to behaviour (left)—often approximate the true latent dynamics (right). **(b)** SIMPL optimises latents and tuning curves by iteratively decoding and refitting one from the other, starting from behaviour and converging on the true dynamics.

Challenges It is well known that high-dimensional neural data is often explained by low-dimensional "latent" variables which correlate strongly with behaviour. For example in the hippocampus, where an animal's position modulates the activity of grid cells and place cells etc. through their distinctive (and celebrated) tuning curves. The fact that *so much* hippocampal variance can be explained by behaviour means it is uncommon to explore further, particularly since existing techniques are costly[1,2], lack easy-to-use implementations[3,4], are too simplistic to model complex tuning curves[3,5], or scale poorly to large datasets[4,5]. This confounds tuning curve interpretation: for instance non-local events like replay[6], theta sweeps[7], or route planning might cause a place cell for position Y to fire when the animal is at X incorrectly suggesting the cell has a field at X. Similarly, if an animal is uncertain about its position due to a lack of sensory cues, sharply tuned fields may appear enlarged and blurred. Other sources of discrepancies could exist. In summary: *To accurately study the true tuning curves we must know the true encoded latent, not just behaviour,* motivating the search for a latent-discovery technique which is fast and scalable and exploits the first-order similarity between latent and behaviour.

SIMPL SIMPL seeks an estimate of the true, unknown latent trajectory $\mathbf{x}^*(t)$ and tuning curves $\mathbf{f}^*(\mathbf{x})$ which led to the observed spike train data $\mathbf{s}(t)$. It does so by iterating a two-step procedure closely related to expectation-maximisation (EM)[8], Fig. 1b. For the "M-step" of each epoch (e) SIMPL fits tuning curves $\mathbf{f}^{(e)}$ using kernel density estimation; i.e. smoothing spikes against $\mathbf{x}^{(e)}$ with a small Gaussian kernel. For the "E-step" SIMPL decodes a new latent $\mathbf{x}^{(e+1)}$ by finding the maximum likelihood estimate of $\mathbf{x}$ given the latest tuning curves and Kalman smoothing it: $\mathbf{x}^{(e+1)} = \text{KalmanSmooth}[\mathbf{x}_{MLE}]$ where $\mathbf{x}_{MLE,t} := \text{argmax}_{\mathbf{x}} \log p(\mathbf{s}_t|\mathbf{x},\mathbf{f}^{(e)})$. SIMPL requires an initial estimate $\mathbf{x}^{(0)}$ which we take to be behaviour (a good "first guess", Fig. 1a). Our design choices impose minimal assumptions on the structure of the data and are cheap, scaling well to large datasets—O(1 hour, 200 neurons, 10^6 spikes, dt=100 ms)→O(<1 min on a CPU-laptop). SIMPL's decoding and fitting subroutines closely match those already widely used in the field. In a theoretical analysis we draw a formal link between SIMPL and EM for a broad class of generative models. A Python/JAX implementation and demo is provided[9].

Artificial grid cell dataset We generated a large synthetic dataset of 225 spiking grid cells for an agent undergoing a 1 hour smooth random walk in a 1 m^2 environment[10]. Initial behaviour was

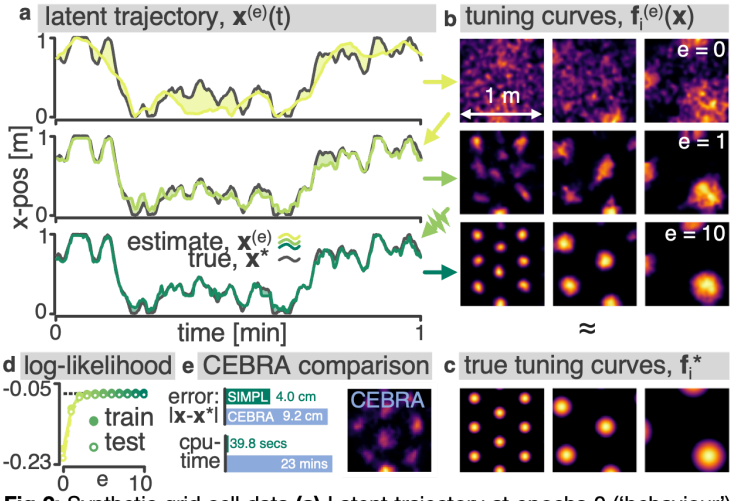

Fig 2: Synthetic grid cell data **(a)** Latent trajectory at epochs 0 ('behaviour'), 1 and 10. Black shows ground truth latent. x-coordinate only. **(b)** Estimated tuning curves (3 of 225 cells). **(c)** Ground truth tuning curves. **(d)** Training curves (dotted line = ground truth ceiling) **(e)** CEBRA comparison.

generated by adding slow noise to the true latent ($\mathbf{x}^{(0)}=\mathbf{x}^*+\boldsymbol{\eta}$, $\langle|\boldsymbol{\eta}|\rangle$=20cm), simulating an animal "uncertain" of its own position, sufficient to blur almost all grid-structure from the behavioural tuning curves (Fig. 2ab top). SIMPL recovered accurate estimates of the true latent and tuning curves (Fig. 2ab, bottom). For comparison, we trained CEBRA[2]—a popular neural network based tool where behaviour serves as contrastive labels—on this dataset using default parameters (Fig. 2e). CEBRA was able to find a good estimate of the latent which was better than behaviour but worse than SIMPL and took over 30x longer to train. In a further analysis (not shown) we ran SIMPL from random initial conditions: SIMPL recovered estimates of the ground truth up to an isomorphism (such as $\phi(\mathbf{x}^*)$ & $\mathbf{f}^*\circ\phi^{-1}$ instead of $\mathbf{x}^*$ & $\mathbf{f}^*$, for some mapping ϕ), leading to tuning curves which predicted the spikes well but did not visually resemble the original hexagonal tuning curves. Conversely, using behaviour as initial conditions strongly biased the solution towards unwarped latent spaces ($\phi\approx$Id), addressing the identifiability issue [11] commonly suffered by latent variable models.

Hippocampal dataset Finally, we ran SIMPL on a large hippocampal dataset, described in Tanni et al.[12] and compared tuning curves before and after. We found that optimised tuning curves—which remained visually similar to, but explained held-out spikes "better" than, their behavioural counterparts, Fig. 3ac—had more place fields and were more spatially-informative. The median place field shrunk in area by 25%, increased in firing rate by 45% and became rounder (Fig. 3d). Large place fields often sharpened or fragmented (Fig. 3a). The optimised latent remained close to behaviour with occasional sharp jumps. These changes were replicated in a dataset from a second animal but not in a control dataset where spikes were resampled from the behaviour and behavioural tuning curves, indicating they are real and not SIMPL artefacts. Also, a previously observed non-uniformity in the place field size-distribution (fields in the middle of an environment are larger than those at the edges[12]) was reduced. Together, these results imply hippocampal cells encode space with more precision and uniformity than previously thought and that latent optimization is an important step to reveal the true structure of the hippocampal code.

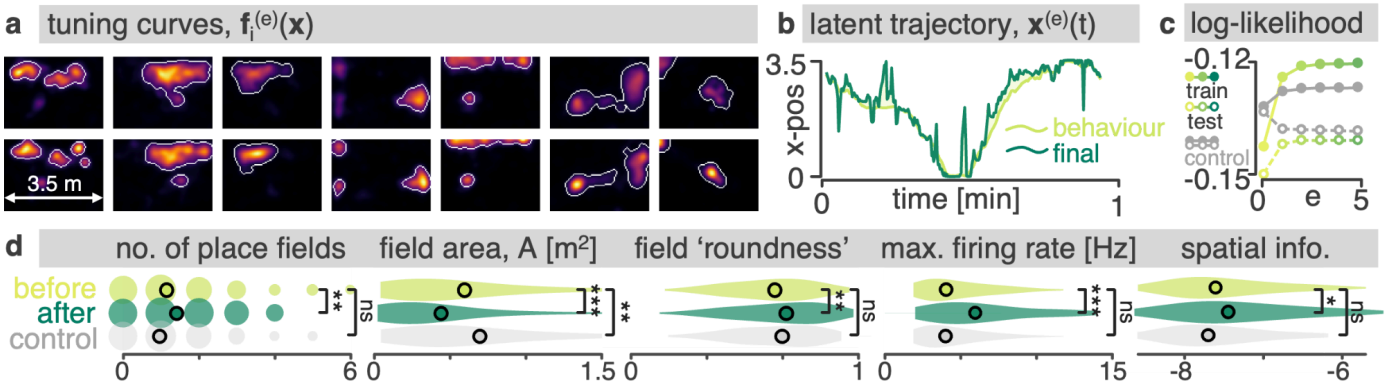

Figure 3: Hippocampal data **(a)** 10 of 226 tuning curves before (top) and after (bottom) SIMPL. **(b)** Initial (behaviour) and final latent estimate. x-coord only. **(c)** Log-likelihood of test and train spikes. **(c)** Place cell / field statistics before and after optimisation. Control model = Grey.

Conclusion We propose SIMPL: a minimal technique to optimise latents and tuning curves from behavioural initialisations, effective on large neural datasets. Like other tools SIMPL will have limitations: in such cases users could experiment with alternative fitting or decoding subroutines – e.g. parametric curve fitting, an RNN decoder etc. which will come with their own costs and benefits. In practice, SIMPL could function as a processing step in any pipeline for analysing neural tuning curves, allowing researchers to rapidly deconfound behaviour from the true latent and therefore make stronger claims about the true structure of representations in the brain.

References **[1]** Pandarinath et al., LFADs (2018) **[2]** Schneider and Lee et al., CEBRA (2023) **[3]** Smith and Brown (2004) **[4]** Wu et al., P-GPLVM (2017) **[5]** Yu et al., GPFA (2008) **[6]** Carr et al. (2011) **[7]** Maurer et al. (2006) **[8]** Dempster et al. (1977) **[9]** https://anonymous.4open.science/r/simpl **[10]** George et al., RatInABox (2024) **[11]** Zhou et al. (2020) **[12]** Tanni et al. (2022)