Abstract—Recently, there has been a focus on drawing progress on representation learning to obtain more identifiable and interpretable latent representations for spike trains, which helps analyze neural population activity and understand neural mechanisms. Most existing deep generative models adopt carefully designed constraints to capture meaningful latent representations. For neural data involving navigation in cognitive space, based on insights from studies on cognitive maps, we argue that the good representations should reflect such directional nature. Due to manifold mismatch, models utilizing the Euclidean space learn a distorted geometric structure that is difficult to interpret. In the present work, we explore capturing the directional nature in a simpler yet more efficient way by introducing hyperspherical neural latent variable models (SNLVM). SNLVM is an improved deep latent variable model modeling neural activity and behavioral variables simultaneously with hyperspherical latent space. It bridges cognitive maps and latent variable models. We conduct experiments on modeling a static unidirectional task. The results show that while SNLVM has competitive performance, a hyperspherical prior naturally provides more informative and significantly better latent structures that can be interpreted as spatial cognitive maps.

Index Terms—Latent variable models, neural population spikes, hyperspherical latent space, cognitive navigation.

I. INTRODUCTION

The brain encodes external input and spontaneous activity with neural population spikes [1], [2], [3]. Previous neural electrophysiology and computational neuroscience studies suggest that while the large-scale neural population activities are high-dimensional, they have low-dimensional latent factors governing the coordinated actions of neurons [4].

Manuscript received 7 August 2022; revised 8 September 2022; accepted 2 October 2022. Date of publication 10 October 2022; date of current version 20 October 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61836003 and in part by the Technology Innovation 2030 under Grant 2022ZD0211700. (Corresponding author: Zhu Liang Yu.)

Yicong Huang is with the School of Automation Science and Engineering, South China University of Technology, Guangzhou 510641, China, and also with the School of Computing, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: 201920116424@mail.scut.edu.cn).

Zhu Liang Yu is with the School of Automation Science and Engineering, South China University of Technology, Guangdong 510641, China, and also with the Pazhou Laboratory, Guangzhou 510641, China (e-mail: zlyu@scut.edu.cn).

Discovering the latent factors is beneficial to revealing the unobserved and intriguing nature of the neural mechanisms underlying complex perceptual or cognitive processes [5], [6], [7]. There have been tremendous studies developing latent variable models to capture the low-dimensional structure from the high-dimensional neural activity. These approaches range from traditional machine learning techniques like principal components analysis (PCA) [8] and Gaussian process [9] to deep neural networks like sequential variational auto-encoders (VAEs) [10] and neural ordinary differential equations [11].

The community has shown that deep generative models provide more promising results in finding the latent representations and latent factors in terms of accuracy and efficiency [10], [12]. Most deep generative models are based on VAEs [13]. While they are theoretically solid upon a well-studied probabilistic framework, these methods also inherit some intrinsic drawbacks of VAEs. In particular, the clusters of latent representations of different neural patterns may become entangled and thus difficult to identify and interpret. One main reason for the entanglement issue is that the default Gaussian prior distribution in VAEs has a limitation of encouraging points to concentrate towards the origin [14] and may introduce undesirable local optima [15]. Some works have tried to disentangle the latent representations by carefully designing additional constraints and the corresponding network architecture [16], [17]. These approaches significantly improve the separability between clusters and hence the interpretability of the data. However, the inferred latent distribution structures remain obscure to analyze neural data.

We investigated how to learn a better latent structure that helps explain neural data. Since most neural data are recorded based on navigating cognition [18], [19], [20], [21], [22], an insightful latent structure should mirror the directional nature. Nevertheless, the Euclidean space without further constraints is difficult to capture such characteristics through optimization. Due to manifold mismatch, existing models may lose topological information in the Euclidean latent space [23]. From the standpoint of representation learning, one probably loses information about important properties in the data because of the geometric distortion. Eventually, it harms analyzing neural data and discovering latent factors.

Noticing these facts and inspired by recent insights of deep learning in manifold [24], we attempt to address the geometric
structure issue by considering better priors. Specifically, in the present work, we draw on recent advances in non-Euclidean deep learning and introduce hyperspherical neural latent variable models (SNLVM). SNLVM improves existing deep generative models via simply replacing the Gaussian distribution on a Euclidean space with the Power Sphere distribution on a hyperspherical space [25]. We validate SNLVM on a reaching dataset, a task that the subject involves unitary cognitive mode during one trial. Compared with existing deep generative models, our model provides a more informative latent structure that can capture the navigating cognition nature.

II. BACKGROUND AND RELATED WORK

A. Notations

Suppose we have simultaneously recorded spike trains from $N$ neurons and corresponding behavior at $T$ time steps. Let $\mathbf{X} \in \mathbb{R}^{N \times T}$ be the observed Poisson spike counts matrix with elements $x_{i,t}$ denoting the spike count of neuron $i \in \{1, \ldots, N\}$ at time $t \in \{1, \ldots, T\}$. Let $\mathbf{Y} \in \mathbb{R}^{B \times T}$ be the observed corresponding behavior, with $B$ being the dimension of the behavior. We consider latent variable models to find unobserved latent factors $\mathbf{Z} \in \mathbb{R}^{M \times T}$ that well explain $\mathbf{X}$ with $M \ll N$. Generally, the literature assumes that the latent factors evolve in terms of a dynamical system.

B. Latent Variable Models (LVMs)

We focus on LVMs for analyzing neural population spikes. There have been numerous LVMs proposed to extracting the lower-dimensional factors [26]. Most early works find the smooth latent dynamics via Gaussian process [9], [27], [28]. In recent studies, deep neural networks based on dynamical systems in a data-driven manner are more efficient and high-performance models for interpreting neural spikes. These state-space methods range from linear dynamical systems [29], [30], [31] to recurrent neural networks [10], [12], [17], [32]. For the present work, the most relevant approaches are latent factor analysis via dynamical systems (LFADS) [10] and targeted neural dynamical modeling (TNDM) [17].

LFADS is an extension of variational auto-encoders (VAEs) [13] for neural spiking data. The encoder and the decoder are both recurrent neural networks (RNNs). The model is trained by minimizing the evidence lower-bound (ELBO) of the marginal log-likelihood of parameterized distribution utilizing the reparameterization trick:

$$
\log p_\theta(\mathbf{X}) \geq \mathbb{E}_{q_\phi(Z|\mathbf{X})}[\log p_\theta(\mathbf{X}|Z)] - \beta KL\left(q_\phi(Z|\mathbf{X})\|p(Z)\right),
$$

(1)

where $q_\phi(Z|\mathbf{X})$ is a variational approximation for the intractable posterior $p(Z|\mathbf{X})$ parameterized by $\phi $, $p_\theta(\mathbf{X}|Z)$ is the generative model parameterized by $\theta $, $\beta > 0$ is a hyperparameter, and $KL(\cdot\|\cdot)$ is the Kullback-Leibler divergence. In a nutshell, LFADS first encodes the observed spikes to latent representations $\mathbf{g} \in \mathbb{R}^L$ and then generates latent factors $\mathbf{Z}$ with the decoder based on the given initial states $\mathbf{g}$. Finally, a linear operator maps $\mathbf{Z}$ to infer firing rates. LFADS enables one to conveniently evaluate the latent factors via the static latent representations $\mathbf{g}$. Like the vanilla VAEs, LFADS assumes that the encoded variable $\mathbf{g}$ follows a Gaussian distribution, whose parameters are inferred from $\mathbf{X}$. Studies have shown that LFADS significantly outperforms previous approaches based on probabilistic models.

TNDM is inspired by a recently proposed model Preferential Subspace Identification (PSID) [31]. It decomposes the latent dynamics into behaviorally relevant and behaviorally irrelevant dynamics. However, PSID is a linear state-space model and cannot capture more complex nonlinear neural dynamics. TNDM combines PSID and LFADS. It decomposes the latent space and draws on recent progress on unsupervised representation learning [33], [34], introducing an additional independence penalty to encourage the latent space to capture different information from the data. TNDM shows the merit of explicitly introducing behavior modeling. It can be seen as improving LFADS by introducing behavior modeling and an unsupervised learning constraint.

C. Hyperspherical VAEs

To extract better latent representations that retain the underlying geometric structure of data, some previous work has explored distributions defined on non-Euclidean space and applied them to deep generative models. One such distribution that is convenient to implement yet has been illustrated to yield more benefits than the standard Gaussian assumption is the von Mises-Fisher (vMF) distribution, or the Gaussian distribution on a hypersphere. S-VAEs first systematically discuss drawbacks of the Gaussian prior and proposes to use the vMF distribution to capture data based on the intuition of manifold matching [23]. However, it requires a rejection sampling scheme to sample from a vMF distribution. Thus despite its generalizability, it may suffer from scalability issues. An alternative is to leverage the Power Spherical (PS) distribution [25] which preserves important properties and aspects of the vMF distribution, but is more stable and efficient.

III. METHODOLOGY

A. Cognitive Maps

The very first step of a cognitive process is to perceive and learn the environment. The capacity of organizing knowledge, generalizing to new information, and responding to perception correspondingly often refers to cognitive maps [35], [36], [37]. The most famed studies around the concept of cognitive maps are based on investigating spatial behaviors [36]. One of the most vital aspects is spatial navigation [20], [38], [39], i.e., planning routes based on knowledge from the real-world space. Specifically, in the hippocampal-entorhinal system, neurons fire with distinct patterns at different spatial places [40]. Studies have shown that, for example, while place cells respond to the current location of the subject [41], firing patterns of grid cells can represent relationships between locations [42]. These activities encode knowledge about different spatial variables and their relationships. Eventually, in a navigation-based cognitive process, this system gives spatial cues to drive spontaneous output, such as motor control.
Recently, studies also suggest that brain constructs representations for general patterns of abstract relations to support complex functions [43], [44], [45], such as memory [46], [47], [48] and decision-making [49], [50], [51]. Response sequences result in trajectories through cognitive space [20], a space that encodes different knowledges [52]. Understandably, since searching this abstract space can also be considered navigation, the spatial representations involve in supporting flexible cognition and behavior [20], [38], [43]. Yet in this paper, we focus on the simplest form of cognitive maps, spatial navigation.

Navigation is all about directions. Mathematically, perhaps a spherical space is the best choice to describe vector directions, like a compass. For neural population activities based on behavior involving spatial navigation, a good LVM is supposed to capture such nature. In other words, there are some intermediate representations reflecting a circular structure that can be interpreted as navigation-based cognition. Unsurprisingly, existing improved LVMs may evaluate whether their latent representations can preserve the so-called “task structure” [16], [17], [53]. However, vector magnitude in the widely used Euclidean space is abundant and even detrimental for modeling navigation, which distorts the latent distribution structure. Even with carefully designed constraints, it is difficult to restore the geometric structure via parameter optimization. Ultimately, it hinders interpreting neural data. Therefore, if we expect the latent representations to mirror the navigation nature, the widely used Euclidean space is a suboptimal alternative.

### B. Manifold Mismatch

Nickel and Kiela [54] is one of the first works proposing to adopt non-Euclidean latent space for deep representation learning. It shows that hyperbolic space is inherently superior for modeling hierarchically structural data like trees and graphs and inspires the following studies to choose a latent topology more suitable for data [55], [56].

Generally, it has been of significant interest to explore deep generative models with non-Euclidean latent space. In practice, most data points are generated by latent variables lying in a non-trivial \( M \)-dimensional manifold \( \mathcal{M} \) and observed in a high-dimensional space \( \mathbb{R}^N \) with \( M \ll N \). Deep generative models try to recover the unobserved latent manifold \( \mathcal{M} \) with a smooth parameterized map as the encoder \( f_{enc} : \mathbb{R}^N \rightarrow \mathbb{R}^D \), where \( \mathbb{R}^D \) is the latent space. However, commonly \( \mathcal{M} \) is not homeomorphic to \( \mathbb{R}^D \). Therefore, if \( D \leq M \) then \( f_{enc} \) will never be a homeomorphism, which means that information about the topological structure of \( \mathcal{M} \) is lost, more or less. Otherwise, if \( D > M \) then \( \mathcal{M} \) can be embedded in \( \mathbb{R}^D \) when \( D \) is ample large. However, it would be problematic when sampling points in the latent space, since not the whole space is feasible for generating valid data points.

For one thing, due to limitations of a Gaussian prior, the inferred representations tend to be entangled and thus are difficult to interpret. A feasible improvement is introducing additional prior knowledge. For example, one can design constraints for learning disentangled and more meaningful representations. Existing works such as Swap-VAE [16] and TNDM [17] assimilate recent advances in deep unsupervised learning to isolate different information in different subspaces. These approaches successfully separate stable target-specific neural patterns and instance-specific individual fluctuations. A simpler solution is modeling external variables [17], [53]. It improves LVMs in identifiability significantly by introducing a strong constraint. Hence, the issue of disentanglement is already somehow alleviated.

For another thing, the manifold mismatch issue also damages interpretability. Deep neural networks learn representations favored for the objective function. However, existing models have not considered the geometric structure of the latent representations. Through gradient-based parameter optimization, neural networks can not naturally obtain latent representations with a circular structure. The main reason is that the Euclidean latent space is improbably homeomorphic to \( \mathcal{M} \), which is the cognitive space that generates observed neural data. Thus a better alternative would be to specify a latent space homeomorphic to \( \mathcal{M} \) and to use distributions on \( \mathcal{M} \). The existence of a homeomorphism ensures that the model can preserve the latent structure without topological information loss, such that we can interpret the latent representations as spatial navigation in cognitive space. Eventually, it inspires us to adopt a hyperspherical latent space, which is naturally superior for modeling directional vectors.

### C. Hyperspherical Latent Space

We introduce hyperspherical neural latent variable models (SNLVM), a deep generative model utilizing a hyperspherical latent space and behavior modeling to learn more interpretable representations for neural dynamics. Unlike previous works, the latent representations by SNLVM can be directly interpreted as navigation-based cognition. We outline the neural network architecture in Figure 1. SNLVM improves LFADS with two ingredients, hyperspherical latent space and behavior modeling. Note that unlike mGPLVM defining latent factors directly on a manifold [57], we assume that Euclidean latent factors are generated from latent variables on the hypersphere. We also include a comparison of network architectures between SNLVM and the baseline models in Figure 2.

For the community, there are additional desirable properties favored to learn representations on a unit hypersphere for neural networks. For example, when the intermediate features have a unit norm, the model becomes more stable, since matrix multiplication is ubiquitous. Moreover, when the features within a category are well clustered, they will lie on a hyperspherical crown and are linearly separable from the rest features, which is one of the most expected attributes [58].

A simple yet widely adopted distribution on the unit hypersphere \( S^{L-1} \) is the von Mises-Fisher (vMF) distribution, like a Gaussian distribution on a hypersphere. Analogously, the vMF distribution is parameterized by a mean direction parameter \( \mu \in S^{L-1} \) and a concentration parameter \( \kappa \geq 0 \). For the special case of \( \kappa = 0 \), the vMF distribution becomes a uniform distribution on \( S^{L-1} \). The probability density function of the vMF distribution for a random vector \( \mathbf{x} \in S^{L-1} \) is...
associated behavior simultaneously. Latent factors are generated based on the latent representation via the decoder. Linear models are used to reconstruct neural firing rates and the difference between LFADS, TNDM and SNLVM.

Fig. 2. Comparison of neural network architectures. We show the SNLVM network architecture. The neural population activity is encoded to a hyperspherical latent space with an RNN, and the Euclidean latent factors are generated based on the latent representation via the decoder. Linear models are used to reconstruct neural firing rates and the associated behavior simultaneously.

Algorithm 1 Sample From a Power Spherical Distribution

Input: dimension $L$, mean direction $\mu$, concentration $\kappa$

Sample $z \sim Beta(a, b) \{a = \frac{L - 1}{2} + \kappa$ and $b = \frac{L - 1}{2}\}$
Sample $v \sim \mathcal{U}(S^{L-2})$

$t \leftarrow 2\pi - 1$

$y \leftarrow [t; (\sqrt{1 - t^2}) v^T]^T$ [concatenation]

$\hat{u} \leftarrow e_i - \mu$ \{ $e_i = [1, 0, \ldots, 0]^T$ is the base vector\}

$u = \hat{u}/\|\hat{u}\|_2$

$x \leftarrow (I_d - 2uu^T)y$ \{ $I_d$ is the $d \times d$ identity matrix\}

Return: $x$

With the Uniform distribution $\mathcal{U}(S^{L-1})$ as the prior, the KL divergence is

$$KL \left( PS(\mu, \kappa) || U \left(S^{L-1}\right) \right)$$

$$= - \log N_L(\kappa) + \kappa (\log 2 + \psi(a) + \psi(a + b)) + \log \frac{2\pi(\frac{L}{2})}{\Gamma(\frac{L}{2})},$$

where $\psi(x) = \frac{\Gamma'(x)}{\Gamma(x)}$. Note that the KL divergence term only depend on $\kappa$, which is consistent with the intuition that $PS(\cdot, 0)$ is equivalent to $\mathcal{U}(S^{L-1})$. In this work, we treat $\kappa$ as a hyperparameter and fine-tune it to investigate how the PS distribution and the hyperspherical latent space work. Given a fix $\kappa$, the KL divergence term is a constant independent of network parameters. The same consideration is also adopted in Xu and Durrett [14]. They have shown that such a setting is beneficial to preventing posterior collapse, which refers to the regularization term forcing the model to estimate the posterior distributions closed to the prior.

D. Sampling Procedure

To leverage the reparameterization trick to efficiently optimize the objective function with gradient-based methods, we follow the scheme proposed in De Cao and Aziz [25] to sample from a PS distribution.

We summarize the sampling procedure for a PS distribution in Algorithm 1. In a nutshell, the sampling technique first decomposes points on $S^{L-1}$ with its tangential component and

---

1One may consider using a Gaussian prior followed by normalization to obtain unit vectors. Mostly it works. However, the effect of variational parameters $\mu$ and $\sigma$ will be entangled because a Gaussian distribution with a smaller $\mu$ will be less concentrated (smaller $\kappa$) after normalization. Besides, despite its seeming complexity, the sampling algorithm for a PS distribution is sufficiently efficient.
normal component as \( y = t\mu + \left(\sqrt{1-t^2}\right)v^T \), where \( v \) is a randomly sampled vector on \( S^{L-2} \) tangent to the hypersphere \( S^{L-1} \) at \( \mu \). Then it reduces to sampling a magnitude \( t \) defined by an affine transformation of a Beta distribution. The sampling module has a well-defined gradient w.r.t. parameters \( \mu \) and \( \kappa \).

The predominant difference is that for the vMF distribution, one need to sample the magnitude \( t \) from a more complex distribution with an unnormalized density as

\[
p(t; \kappa, L) \propto e^{\kappa(1 - t^2)^{L-3}},
\]

where \( t \in [-1, 1] \). One needs an acceptance-rejection sampling scheme to sample from this distribution. In contrast, sampling from a Beta distribution is more stable and efficient.

### E. Regularization and Objective Function

Previous work has shown that involving behavior information in neural dynamics analysis is a simple yet promising improvement for LVMs. This strategy of modeling the relationship between the latent representations and task variables is also known as a hybrid scheme [53]. Both discrete labels [53] and continuous recordings [17], [31] help regularize the model. Specifically, maximizing the distribution likelihood of task variables serves as a strong prior, promoting latent representation to align with the actual unobserved latent manifold that generates the neural spikes based on the behavior. Consequently, we follow previous studies and define the ultimate objective function for our model as

\[
\max_{\phi, \theta} \mathbb{E}_{q_{\phi}(Z|X)} \left[ \log p_{\theta}(X|Z) \right] + \lambda_b \mathbb{E}_{q_{\phi}(Z|X)} \left[ \log p_{\theta}(Y|Z) \right]
\]

where \( \lambda_b > 0 \) is a hyperparameter as a Lagrange multiplier for an additional constraint of behavior decoding. Since the KL divergence is a constant given a fix \( \kappa \), we simply omit the term in the objective function. The model computes hyperspherical variational posterior distributions with an RNN given neural population spikes \( X \). We sample latent representations hyperspherical latent representations \( g \) from the prior distributions. Then we generate latent factors \( Z \) from \( g \). Two linear model project \( Z \) to firing rates and behavior trajectory, respectively. Finally, we compute the objective function, the Poisson likelihood of inferring firing rates and the likelihood of inferring behavior.

### IV. Experimental Setup

#### A. Dataset

To verify the validity of SNLVM, we apply our model to data from a recently published thesus macaque reaching experiment [59]. The data were recorded when a macaque was trained to perform a center-out reaching task in eight different directions on the computer screen. The dataset we use in the present work comprises simultaneously recorded neural population spikes from the primary motor cortex and associated hand position as the behavior. Because each trial is unidirectional, the spatial navigation is static for each trial, respectively. We evaluate our model and compare it with LFADS and TNDM on single-session data aggregated from one of the six trained macaques. The dataset contains 176 trials total, and we use 80% as the training set (136 trials), 10% as the validation set (17 trials), and 10% as the test set (17 trials).

#### B. Model Configurations

We implement SNLVM, LFADS, and TNDM as sequential VAEs and we fix the network architectures to be the same. After fine-tuning the hyperparameters, we fix all shared hyperparameters to be the same as well. We set the latent space as 64-dimensional. We implement the RNN encoder and the RNN decoder with GRU with 64 system states. We fine-tune and fix the KL divergence loss term \( \beta \) to be 0.1 for LFADS and TNDM based on the principle of \( \beta \)-VAE [60], [61]. We fix the behavior decoding loss term \( \lambda_b \) to be 0.1 for TNDM and SNLVM. We train all models on a single Nvidia Titan RTX GPU for 1000 epochs with early stopping with batch size as 16. We train all models with the Adam optimizer with a learning rate starting from 0.01 with adaptive decay. We implement our model based on the code by TNDM [17].

### V. Results

#### A. Quantitative Evaluation

We first wonder whether our model can well explain neural population spikes, since the very first requirement for LVMs is that models should provide promising results in inferring single-trial neuronal firing rates for the neural population spikes. The latent variables are meaningful and provide insights into the neural data only if they faithfully explain the observed neural activities. To assess the goodness of fit quantitatively, we compute the Poisson log-likelihood as the primary metric. We also compute the root mean square error (RMSE) between inferred firing rates and empirical ground-truth firing rates, or peri-stimulus time histogram (PSTH). They are obtained by averaging neural spikes across trials within one particular condition. We evaluate behavioral decoding accuracy by fitting a linear regression model from latent factors to behavior and computing \( R^2 \) between decoded behavior and ground-truth behavior. These three metrics and their improved variants are common tools for evaluating LVMs [62]. All metrics are computed on held-out data.

We compare SNLVM with LFADS and TNDM by setting different numbers of the latent factors. We summarize the results in Figure 3. We observe that all models show gradual improvement with increasing latent factors since more latent factors can cover more variance of the neural population activity. We also observe that SNLVM and TNDM lightly but consistently outperform LFADS in fitting neural activity and reconstructing behavior. Moreover, the performances of SNLVM and TNDM are relatively close. The results indicate that SNLVM has competitive performance compared with previous deep LVMs. Note that SNLVM is unnecessary to significantly outperform others in terms of these metrics since its superiority lies in capturing latent structure.

2Our code is in https://github.com/HilbertHuangHitomi/SNLVM
B. Latent Representations Topology and Latent Factors

To elucidate the merit of adopting the hyperspherical latent space, we compare the estimated latent representations and the generated latent factors by LFADS, TNDM, and SNLVM. We train the models with 4 latent factors and visualize the results in Figure 4.

1) Latent Representations by t-SNE: As shown by t-SNE, LFADS well clusters the representations of neural population spikes with the same direction owing to the expressiveness of deep neural networks as promised. Nevertheless, the latent variables are more sensitive to noise and hence the clusters seem entangled. In the contrary, the inferred latent variables by TNDM and SNLVM show significantly better separability between different neural patterns and aggregation within the same ones. TNDM can be seen as LFADS + behavior modeling, while SNLVM can be seen as LFADS + hypersphere + behavior modeling. Thus we deduce that this is achieved by involving behavior decoding in modeling. Behavior modeling encourages the representations to fit neural spikes and external variables simultaneously, and hence serves as a strong regularization, resulting in a more constrained model. Thus we can find that the estimated posterior distributions by TNDM and SNLVM are more compact and identifiable than LFADS, a benefit of hybrid modeling [53].

2) Latent Representations by PCA: Notably, PCA reveals the most important distinction in the latent topology between the models. Since t-SNE is a nonlinear dimensionality reduction technique attempting to preserve the local structure for neighbor points, it may lose important global structure information and thus may be misleading. Although mostly, PCA is not suitable for revealing the complex structure of a nonlinear manifold, it provides more authentic geometric information about the data, as it maintains most distance properties. Therefore, we examine the latent space structure by applying PCA as well. We observe that all models provide a circular or nearly circular latent structure consistent with other models on the same task [16], [53]. Remind that each trial only involves one target direction. In terms of neurophysiology, the subject maintains unitary neural states of cognitive navigation to perform one particular reaching trial. It implies that the navigational direction in the cognitive space is static for each trial, respectively. Therefore, the latent representations are supposed to preserve this cognitive nature. For LFADS, the Gaussian distributional restriction encourages the points to cluster in the center, eventually leading to unavoidable entanglement. TNDM introduces additional priors, especially behavior decoding, which significantly improve the separability of different classes, yet the geometric structure of latent representations remains vague. The results suggest that behavior modeling can not address the latent structure issue. Eventually, for these models, it is difficult to explain why the model learns such structure and what it mean in neurophysiology. Compared with LFADS and TNDM, SNLVM precisely and elegantly captures a better structure reflecting the spatial navigation in cognitive space. The results imply that hyperspherical latent space introduces a meaningful structure we can interpret. Besides, as a byproduct, the clusters by SNLVM are more compact and separable.

3) Latent Factors: We find that the variance of the latent factors by TNDM and SNLVM is smaller and has a clearer separation than LFADS. It again verifies the merit of involving task variables [53]. Moreover, TNDM introduces a prior to separate behaviorally relevant and behaviorally irrelevant factors. Thus half of the factors only contain homogeneous information. These factors are trivial because they repeat similar dynamics, containing little information about the neural data. However, this prior may be problematic, because it explicitly limits the model to capture more complex dynamics. By contrast, all 4 factors by SNLVM capture non-trivial neural dynamics. These factors are crucial to revealing more information about the neural manifold. Moreover, the process that spatial cognitive states drive observed neural activity is consistent with RNNs, generating dynamics with static initial states. Thus a more accurate latent structure helps the model discover appropriate initial states, which further helps the decoder find correct latent factors.

C. Analysis on Held-Out Direction

To further illustrate SNLVM provides more informative representations, we conduct an experiment with one held-out direction. In this experiment, we held out neural spikes and the corresponding behavior of one reaching direction during training. We investigate whether the models can infer meaningful latent representations and correct behavior reconstruction for the neural population spikes they have no access to during training. The results are shown in Figure 5.
Fig. 4. Comparison of latent space learned by LFADS, TNDM and SNLVM. We show the results via PCA and t-SNE with normalization on the left. We show the 4 generated latent factors given the inferred latent variables as the initial states. The colors for different reaching directions are presented on the right.

Fig. 5. Comparison of inferred latent representations and behavior decoding for the held-out direction by LFADS, TNDM, and SNLVM. The transparent plots are held-in trials, and the black ones are trials of the held-out direction. We visualize the high-dimensional representations via PCA and normalization. We show the ground-truth behavior trajectories with dash lines and the reconstruction with solid lines. We also provide a quantitative measure for the results (mean±std). The colors for different reaching directions are presented on the right.

We observe that LFADS, TNDM, and SNLVM all obtain a decent latent structure without the held-out trials. Given the held-out neural population spikes, the inferred latent representations are close to the results when they are trained with all data. As shown by the behavior reconstruction results, one can see that all models provide imperfect yet roughly correct behavioral trajectories. It indicates that these deep generative models successfully capture the latent structure and provide promising performance to unknown patterns.

Nevertheless, SNLVM shows the distinct merit of learning a better topological structure above LFADS and TNDM. For LFADS and TNDM, the latent structure is irregular and thus obscure to find the held-out data. The latent representations of the held-out data mix up with the rest completely. On the contrary, for SNLVM, compared with training with the whole dataset, the distribution of the clusters in the latent space shows unnatural asymmetry around the position that the held-out data should be. The results are consistent with the cognitive process. When a subject search a location that has never seen before, it naturally refers to familiar sites and identifies a reliable direction accordingly. SNLVM reproduces the process elegantly. Relatively, models with the Euclidean space and the Gaussian distribution are more difficult to obtain such topological structure. Thus it is hard to interpret latent representations with neurophysiological meaning, even though we understand faintly that they are relevant to the task.

We also investigate the results quantitatively. We fit a linear model to classify the held-out direction from the rest ones given the latent representations. We evaluate the result via mean classification accuracy (ACC) for the held-out direction. It shows that SNLVM provides a much separable representations. Therefore, the results illustrate significant superiority of leveraging the hyperspherical latent space over the Euclidean space.

VI. DISCUSSION

A. Effects of Different $\kappa$

In our experiments, we find that it is beneficial to fix $\kappa$ as a hyperparameter. It seems that it may reduce the flexibility and the expressiveness of the model to use a fixed $\kappa$ for all the approximate posterior distributions. We find that when $\kappa$ is amply large it is insensitive to the model performance because the behavior decoding scheme provides robust inductive bias. A wide range of $\kappa$ values appear to work well in this
paper. Moreover, compared with the vMF distribution, the PS distribution is more stable given large $\kappa$.

We perform a sweep over the setting of $\kappa$ and visualize the inferred latent variables. As shown in Figure 6, we can see how the concentration parameter $\kappa$ influences the latent structure. We find that with behavior modeling, when $\kappa$ is larger than $10^2$, our model is insensitive to it and the latent structure stays the same. Consequently, to further understand $\kappa$ empirically, we also zero out the behavior modeling term in the objective function to eliminate its gain. We observe that when $\kappa$ is small, since all the posteriors may distribute more uniformly on the hypersphere, the representations are blended. On the contrary, the latent representations show clear separability and a better circular structure when $\kappa$ is large enough. With extremely large $\kappa$, one can imagine that all posterior distributions may locate on a hyperspherical crown. Due to the PS distribution providing the numerical stability, the model works well even $\kappa$ seems unusually larger than the model parameters.

Furthermore, we notice that the optimum $\kappa$ is relatively larger than the trainable parameters in the model with proper $L_2$ regularization. It can be seen as an outlier of the distribution of the parameters. Hence from the perspective of optimization, it may be difficult to learn the best mapping due to training stability issues. It illustrates why manually setting $\kappa$ provides better results.

### B. Ablation Study on Behavior Modeling

Both SNLVM and TNDM show the advantages of involving behavior decoding in the training process. As mentioned above, modeling task variables serves as an influential prior and facilitates the model to learn a more identifiable latent distribution. We wonder whether our model can still provide a more interpretable latent structure than LFADS without it. The result may further support the utilization of non-Euclidean latent space. We set $\lambda_b$ in the objective function to zero to investigate the model.

We first compare SNLVM and LFADS to quantitatively evaluate our model. The results are in Table I. We observe that the performance of SNLVM without behavior modeling is almost equivalent to LFADS. Unsurprisingly, now the only difference between LFADS and SNLVM is the assumption of latent variables. Although such an assumption may change the latent topological structure, it does not introduce additional inductive bias. Therefore, the quantitative results are close.

Then we compare the learned latent structure qualitatively. We summarize the results in Figure 7. We observe that without

**TABLE I**

**The Quantitative Results of Ablation Study on Behavior Modeling. We Investigate the Models by Setting Different Latent Factors. We Denote SNLVM Without Behavior Modeling as SNLVM−**

| metric | PNLL | RMSE | $R^2$ |
|--------|------|------|-------|
| model  | LFA1S | SNLVM− | LFA1S | SNLVM− | LFA1S | SNLVM− |
| 2      | $-1765.62 \pm 0.87$ | $-1764.12 \pm 0.95$ | 0.097 | 0.096 | 0.485 | 0.512 |
| 4      | $-1749.03 \pm 1.48$ | $-1748.11 \pm 1.01$ | 0.088 | 0.087 | 0.888 | 0.899 |
| 6      | $-1741.14 \pm 0.36$ | $-1740.84 \pm 0.62$ | 0.087 | 0.087 | 0.926 | 0.927 |
| 8      | $-1737.90 \pm 2.01$ | $-1737.31 \pm 0.70$ | 0.086 | 0.085 | 0.933 | 0.937 |
| 10     | $-1734.24 \pm 1.39$ | $-1734.20 \pm 0.44$ | 0.084 | 0.084 | 0.943 | 0.950 |
behavior modeling, the geometrical structure of the latent variables inferred by SNLVM still preserves a circular geometry of the reaching directions. The predominant difference is that without behavior variables, noise in neural activities may affect the latent representations more significantly. Therefore the variance of the approximation posterior distributions is relatively larger. The results are promising indicators that behavior decoding does regularize the model to learn more compact representations for the same neural pattern, and thus it is more robust to noise. Moreover, whether including behavior decoding or not, the latent structure learned by SNLVM is consistently better than LFADS based on Gaussian prior in Euclidean space on preserving a circular structure of the task. The results agree with previous work and verify the advantages of using a hyperspherical latent space. Note that removing behavior modeling, our model is almost the same as LFADS except for the assumption of the latent space. We do not apply additional constraints to the latent space and intentionally encourage the model to learn this property.

C. Computational Efficiency

Experiments have shown that sampling from a PS distribution is at least $6 \times$ faster and more stable than from a vMF distribution [25]. In our experiment, because the sampling algorithm takes only one minor part of the model, the most computation cost is on the encoder and decoder, so the acceleration is much less significant. Across different experiment settings, SNLVM with PS distribution is about $2.5 \times$ faster than with vMF distribution on average. The effect varies according to the computing device. For example, in our test, the value decreases to $1.9 \times$ using a single Nvidia RTX 2060 GPU for Laptop.

VII. CONCLUSION

In this work, we discuss the neurophysiological implication of latent representations of LVMs. At first glance, it seems natural and obvious that the latent structure should be similar to the task setting. However, interpreting the latent representations in neurophysiology is crucial to further understanding more complex cognitive processes. Previous work fails to answer the question. Based on cognitive maps and spatial cognitive space, we point out that the latent representations are supposed to preserve the cognitive navigation nature. Then by introducing a simpler yet more effective prior, hyperspherical latent space, we improve existing VAE-based deep generative models to capture the latent topological structures. The geometric structure of the latent variables is naturally more informative that can be interpreted as cognitive navigation. Our insight introduces cognitive maps in LVMs for the first time. Nevertheless, it is also worth mentioning that our model needs further evaluation on other task. For example, current experiments only verified tasks with static cognitive navigation but not dynamic in one trial. How our model helps model abstract cognitive maps remains underexplored. We hope our theory can provide more insights for the community to understand complex cognitive or perceptual processes.

ACKNOWLEDGMENT

The authors would like to thank all reviewers for their constructive suggestions and Prof. Zhenghui Gu for her help in improving this article.

REFERENCES

[1] S. Saxena and J. P. Cunningham, “Towards the neural population doctrine,” Current Opinion Neurobiol., vol. 55, pp. 103–111, Apr. 2019.
[2] S. Vyas, M. D. Golub, D. Sussillo, and K. V. Shenoy, “Computation through neural population dynamics,” Annu. Rev. Neurosci., vol. 43, pp. 249–275, Jul. 2020.
[3] R. B. Ebitz and B. Y. Hayden, “The population doctrine in cognitive neuroscience,” Neuron, vol. 109, no. 19, pp. 3055–3068, Oct. 2021.
[4] J. P. Cunningham and B. M. Yu, “Dimensionality reduction for large-scale neural recordings,” Nature Neurosci., vol. 17, no. 11, pp. 1500–1509, 2014.
[5] J. M. Shine et al., “Human cognition involves the dynamic integration of neural activity and neuromodulatory systems,” Nature Neurosci., vol. 22, pp. 289–296, Jan. 2019.
[6] J. M. Shine et al., “The low-dimensional neural architecture of cognitive complexity is related to activity in medial thalamic nuclei,” Neuron, vol. 104, no. 5, pp. 849–855, 2019.
[7] R. Chaudhuri, B. Gerçek, B. Pandey, A. Peyrache, and I. Fiete, “The intrinsic attractor manifold and population dynamics of a canonical cognitive circuit across waking and sleep,” Nature Neurosci., vol. 22, no. 9, pp. 1512–1520, 2019.
[8] M. B. Ahrens et al., “Brain-wide neuronal dynamics during motor adaptation in zebrafish,” Nature, vol. 485, no. 7399, pp. 471–477, 2012.
[9] M. Y. Byron, J. P. Cunningham, G. Santhanam, S. I. Ryu, K. V. Shenoy, and M. Sahani, “Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity,” in Proc. Adv. Neural Inf. Process. Syst., 2009, pp. 1881–1888.
[10] C. Pandarinath et al., “Inferring single-trial neural population dynamics using sequential auto-encoders,” Nature Methods, vol. 15, no. 10, pp. 805–815, Oct. 2018.
[11] T. D. Kim, T. Z. Luo, J. W. Pillow, and C. Brody, “Inferring latent dynamics underlying neural population activity via neural differential equations,” in Proc. Int. Conf. Mach. Learn., 2021, pp. 5551–5561.
[12] M. R. Keshtkaran and C. Pandarinath, “Enabling hyperparameter optimization in sequential autoencoders for spiking neural data,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 15937–15947.
[13] D. P Kingma and M. Welling, “Auto-encoding variational Bayes,” 2013, arXiv:1312.6114.
[14] J. Xu and G. Durrett, “Spherical latent spaces for stable variational autoencoders,” in Proc. Conf. Empirical Methods Natural Lang. Process., 2018, pp. 4503–4513.
[15] J. Lucas, G. Tucker, R. B. Grosse, and M. Norouzi, “Don’t blame the ELBO! A linear VAE perspective on posterior collapse,” in Proc. Adv. Neural Inf. Process. Syst., vol. 32, 2019, pp. 9408–9418.
[16] R. Liu et al., “Drop, swap, and generate: A self-supervised approach for generating neural activity,” in Proc. Adv. Neural Inf. Process. Syst., vol. 34, 2021, pp. 1–13.
[17] C. Hurwitz et al., “Targeted neural dynamical modeling,” in Proc. Adv. Neural Inf. Process. Syst., vol. 34, 2021, pp. 1–14.
[18] A. Finkelstein, D. Derdkman, A. Rubin, J. N. Foerster, L. Las, and N. Ulanovsky, “Three-dimensional head-direction coding in the bat brain,” *Nature*, vol. 517, no. 7533, pp. 159–164, Jan. 2015.

[19] E. I. Moser, M-B. Moser, and B. L. MaNaughton, “Spatial representation in the hippocampal formation: A history,” *Nature Neurosci.*, vol. 20, pp. 1448–1464, Nov. 2017.

[20] J. L. S. Bellmund, P. Gärdenfors, E. I. Moser, and C. F. Doeller, “Navigating cognition: Spatial codes for human thinking,” *Science*, vol. 362, no. 6415, Nov. 2018, Art. no. eaat6766.

[21] D. E. Angelaki et al., “A gravity-based three-dimensional compass in the mouse brain,” *Nature Commun.*, vol. 11, no. 1, pp. 1–13, Dec. 2020.

[22] R. M. Griess et al., “Irregular distribution of grid cell firing fields in rats exploring a 3D volumetric space,” *Nature Neurosci.*, vol. 24, no. 11, pp. 1567–1573, Nov. 2021.

[23] T. R. Davidson, L. Falorsi, N. D. Cao, T. Kipf, and J. M. Tomczak, “Hyperspherical variational auto-encoders,” in *Proc. 34th Conf. Uncertainty Artif. Intell.*, 2018, pp. 856–865.

[24] J. Park, J. Cho, H. J. Chang, and J. Young Choi, “Unsupervised hyperbolic representation learning via message passing auto-encoders,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 5516–5526.

[25] N. De Cao and W. Aziz, “The power spherical distribution,” 2020, arXiv:2006.04437.

[26] C. Hurwitz, N. Kudryashova, A. Onken, and M. H. Hennig, “Building population models for large-scale neural recordings: Opportunities and pitfalls,” 2021, arXiv:2102.01807.

[27] Y. Zhao and I. M. Park, “Variational latent Gaussian process for recovering single-trial dynamics from population spike trains,” *Neural Comput.*, vol. 29, no. 5, pp. 1293–1316, May 2017.

[28] A. Wu, N. A. Roy, S. Keeley, and J. W. Pillow, “Gaussian process based nonlinear latent structure discovery in multivariate spike train data,” in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 3499–3508.

[29] J. H. Macke, L. Buesing, J. P. Cunningham, B. M. Yu, K. V. Shenoy, and A. M. Sahani, “Empirical models of spiking in neural populations,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1350–1358.

[30] Y. Gao, E. W. Archer, L. Paninski, and J. P. Cunningham, “Linear dynamical neuronal population models through nonlinear embeddings,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 163–171.

[31] O. G. Sani, H. Abbaspourazad, Y. T. Wong, B. Pesaran, and M. M. Shanmichi, “Modeling behaviorally relevant neural dynamics enabled by preferential subspace identification,” *Nature Neurosci.*, vol. 24, no. 1, pp. 140–149, Jan. 2021.

[32] Q. She and A. Wu, “Neural dynamics discovery via Gaussian process recurrent neural networks,” in *Proc. 35th Uncertainty Artif. Intell. Conf.*, vol. 115, PMLR, 2020, pp. 454–464.

[33] A. Kumar, P. Sattigeri, and A. Balakrishnan, “Variational inference of disentangled latent concepts from unlabeled observations,” in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–16.

[34] H. Kim and A. Mnih, “Disentangling by factorising,” in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 2649–2658.

[35] E. C. Tolman, “Cognitive maps in rats and men,” *Psychol. Rev.*, vol. 55, no. 4, p. 189, 1948.

[36] J. O’keefe and L. Nadel, *The Hippocampus as a Cognitive Map*, Oxford, U.K.: Oxford Univ. Press, 1978.

[37] J. C. R. Whittington, D. McCaffary, J. J. W. Bakermans, and T. E. J. Behrens, “How to build a cognitive map: Insights from models of the hippocampal formation,” 2022, arXiv:2202.01682.

[38] R. A. Epstein, E. Z. Patai, J. B. Julian, and H. J. Spiers, “The cognitive map in humans: Spatial navigation and beyond,” *Nature Neurosci.*, vol. 20, no. 11, pp. 1504–1513, 2017.

[39] J. J. Tuukker, P. Beed, M. Brecht, R. Kempter, E. I. Moser, and D. Schmitz, “Microcircuits for spatial coding in the medial entorhinal cortex,” *Physiol. Rev.*, vol. 102, no. 2, pp. 653–688, Apr. 2022.

[40] E. I. Moser, E. Kropff, and M-B. Moser, “Place cells, grid cells, and the brain’s spatial representation system,” *Annu. Rev. Neurosci.*, vol. 31, no. 1, pp. 69–89, Jul. 2008.