No Free Lunch from Deep Learning in Neuroscience: A Case Study through Models of the Entorhinal-Hippocampal Circuit

Anonymous Author(s)
Affiliation
Address
email

Abstract

Research in Neuroscience, as in most scientific disciplines, is undergoing a renaissance based on deep learning. Unique to Neuroscience, deep learning models can be used not only as a tool but interpreted as models of the brain. The central claims of recent deep learning-based models of brain circuits are that they make novel predictions about neural phenomena or shed light on the fundamental functions being optimized. We show, through the case-study of grid cells in the entorhinal-hippocampal circuit, that one often gets neither. We begin by reviewing the principles of grid cell mechanism and function obtained from analytical and first-principles modeling efforts, then rigorously examine the claims of deep learning models of grid cells. Using large-scale hyperparameter sweeps and theory-driven experimentation, we demonstrate that the results of such models may be more strongly driven by particular, non-fundamental, and post-hoc implementation choices than fundamental truths about neural circuits or the loss function(s) they might optimize. Finally, we discuss why these models cannot be expected to produce accurate models of the brain without the addition of substantial amounts of inductive bias, an informal No Free Lunch result for Neuroscience. In conclusion, caution and consideration, together with biological knowledge, are warranted in building and interpreting deep learning models in Neuroscience.

1 Introduction

Within the past decade, deep learning (DL) has leapt from obscurity to underpinning nearly every success story in machine learning e.g., [41, 50, 4] and increasingly many advances in fundamental science research e.g., [28, 12, 3]. In neuroscience, deep learning is similarly gaining widespread adoption as a useful method for behavioral and neural data analysis [58, 48, 37, 45, 22, 36, 44, 33, 40].

But DL offers a unique contribution to neuroscience that goes beyond its role in other fields, in that neural networks can be viewed as models of the brain. The success of DL in matching or surpassing of human performance means it is now possible to construct models of circuits that underlie human intelligence. As a recent review wrote, “Many researchers are excited by the possibility that deep neural networks may offer theories of perception, cognition and action for biological brains. This approach has the potential to radically reshape our approach to understanding neural systems." [49].

The primate visual system has provided a fertile and productive ground for this approach [47]: “[I]t has been shown that deep artificial neural networks (ANNs) can mimic, closely in some cases, the

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

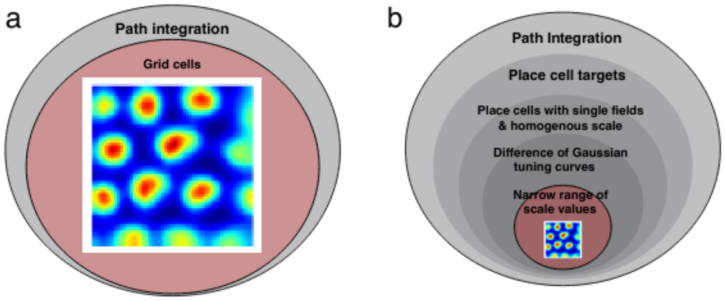

Figure 1: (a) Prior work [11, 1, 52, 61, 42] presents a story that path integration generically drives the formation of grid cells. (b) In this work, we show that only a small fraction of hyperparameters yield grid cells.

representational transformations in primate perceptual systems [32, 31] and thereby can be leveraged to manipulate neural activity [2]." The progress here has been notable because of the previous dearth of good models of higher-level biological visual areas.

Broadly, the essential claims of DL-based ANN models of the brain that we examine here are that 1) Because they are trained on a specific objective function, if the resulting representations match what has been observed in the brain, then they reveal the objective function of the brain, or 2) That these models, when trained on behavioral or otherwise sensibly motivated objective functions, should make novel predictions about representations in the brain.

However, given the nascent nature of such approaches and the excitement accompanying the claims, we should examine them carefully. In deep learning and deep reinforcement learning, performance improvements attributed to novel objective functions and algorithms have instead been shown to stem from seemingly minor and often-unstated implementation choices [60, 24, 16, 27]. In this paper, we ask whether Neuroscientists should similarly be cautious that DL-based models of neural circuits may tell us less about fundamental scientific truths and more about programmers' particular implementation choices, and ask whether these approaches are more post-hoc than predictive.

To explore these questions, we evaluate recent DL-based models of grid cells in the entorhinal-hippocampal circuit. The medial entorhinal cortex (MEC) and hippocampus (HPC) are part of the hippocampal formation, a system that displays beautiful and fascinating properties. In 1971, HPC was shown to contain *place cells*, neurons which fire if and only if the animal is at particular location(s) [43], and MEC was later shown to contain *grid cells*, neurons which fire if and only if the animal is at the vertices of a hexagonal lattice [23]. As a matter of terminology, we refer to cells that are periodically active at the vertices of either a square or hexagonal lattice as **lattice cells**, and refer to hexagonal lattice cells as **grid cells**. Over the past 40 years, the hippocampal formation has proved a rich vein for learning about how the brain organizes spatial and episodic memory for experimentalists and theorists alike (Section 2), with many mysteries remaining. A recent series of papers [11, 1, 52, 61, 42]) have used DL to claim that the task of path integration (i.e. integrating self-motion estimates to track spatial position over time) drives the formation of grid cells. In this paper, we use code from prior publications to demonstrate that these results are due entirely to implementation details that tell us more about those choices than they do about MEC. Specifically, by leveraging theoretically-guided large-scale hyperparameter exploration and hypothesis-driven experimentation, we demonstrate that: 1) Networks trained on a path integration objective almost always optimally encode position, but almost never learn lattice cells (hexagonal or square) to do so. 2) The emergence of lattice cells depends wholly on a specifically chosen encoding of the supervised target, not on the task itself. 3) The grid periods and period ratios likewise depend on hyperparameters choices and are not set by the task. 4) The chosen encoding requires many other hyperparameter choices to produce lattice cells that are highly sensitive, in the sense that small alterations result in loss of lattice cells.

Deep learning produces grid cells and their attendant properties only after making many specific design choices and arduously searching hyperparameter configurations to generate such responses, effectively baking grid cells into the task-trained networks. In other words, it is improbable that DL models of path integration would have produced grid cells as a novel prediction simply from task-training, had grid cells not already been known to exist. Moreover, it is unclear what added intepretability or understanding these models contribute, beyond or even up to what has already been shown for these particular circuits. These results challenge the notion that DNNs offer a free lunch for Neuroscience in terms of discovering the brain's objective functions or generating novel a priori predictions about single-neuron representations, and warn that caution is needed when building and interpreting such models.

We emphasize that our work would not have been possible without the open-source code released by [52, 61], for which the authors should be commended; by making their code available, we have been able to present novel insights that we hope will contribute to a clearer understanding of the risks and rewards of using and interpreting DL models in Neuroscience.

2 Background and related work

Grid cells [23] are found in the medial entorhinal cortex (MEC) of mammals. Grid cells are tuned to represent the spatial location of animals as they freely traverse 2D space. Each cell fires at every vertex of a triangular lattice that tiles the explored space, regardless of the speed and direction of movement through the space. As a population, they exhibit several striking properties [23] that provide support for a specialized circuit; here we some of the properties most relevant to testing grid cell models, as we will below: 1) The spatial period of the periodic grid response is independent of the spatial environment's size and shape for familiar spaces. 2) The response is updated even in the dark, in the absence of external cues. 3) They form discrete modules (clusters), such that all cells within a module share a common period and orientation, with monotonically increasing grid scales along the dorso-ventral axis of the MEC [56]. Adjacent modules exhibit specific period ratios whose values are all close to 1.4. 4) The grid cells within each module exhibit stable cell-cell relationships across environments and during sleep [62, 59, 20, 19], unlike place cells, which remap across environments and do not exhibit preserved structure during sleep.

The basic mechanism underlying grid cells is believed to be that described by continuous attractor models [7, 18, 5]. They operate according to a basic principle: Translation-invariant lateral connectivity within the grid cell network that results in Turing instability and pattern formation. These models explained how velocity inputs can be converted into updated spatial estimates (path integration) by grid cells and made several predictions, which have been confirmed in experiments: 1) Stable cell-cell relationships regardless of environment [62, 59, 20, 19]; 2) A toroidal attractor manifold in neural state space across awake and sleep states [19, 21].

The counterintuitive encoding of position by grid cells (a local, non-periodic variable such as location encoded by a non-local, periodic code) has led to theoretical examination of its properties, with results showing that the grid code provides three properties not shared with many conventional (unimodal or grandmother cell-like) neural codes: 1) Equi-norm translation-invariant representations for path integration; 2) Exponentially large capacity of internally generated states as a function of neuron number [17, 55, 39]; 3) Intrinsic error correcting code [55].

3 Networks trained on path integration tasks optimally encode position but rarely learn lattice (hexagonal or square) cells

Path integration is the task of using self-motion estimates to track one's spatial position over time, a crucial component of spatial navigation. The prominent message of DL models of grid cells is that training recurrent networks to path integrate causes the networks to learn lattice cells [11, 1, 52, 61, 42]. As one paper [52] wrote in their section titled "Optimally encoding position yields diverse grid-like patterns," "Why do these diverse architectures, across diverse tasks (both navigation and

autoencoding), all converge to a grid-like solution, and what governs the lattice structure of this solution?" As we shall see, most networks do not converge to a grid-like solution, and what governs the structure (when it emerges) is determined entirely by what the programmer bakes in.

We start by running large-scale hyperparameter sweeps across common implementation choices: 1) Network architectures: RNN [15]; LSTM [26]; GRU [9]; UGRNN [10] 2) Activation functions: Linear; Sigmoid; Tanh; Rectified Linear 3) Optimizers: Adam [34]; RMSProp [25] 4) Supervised Targets: Cartesian spatial position; high-dimensional Place Cell (PC) population code with Gaussian tuning curves (used by [1]); high-dimensional PC population code with Difference-of-Gaussians tuning curves (used by [52, 42] 5) Loss functions: mean squared error (MSE) on the agent's Cartesian spatial position (used by [29, 11]); geodesic distance on the agent's polar spatial position (used by [30]); softmax cross entropy on a high-dimensional population of place cell (PC) units (used by [1, 52, 42]) 6) Miscellaneous: weight initialization, weight regularization, randomization seed.

When training networks on supervised place cell (PC) targets, we additionally swept: 1) The place cells' receptive field σ i.e. the standard deviation of the Gaussian tuning curve (often denoted $\sigma_E \stackrel{\text{def}}{=} \sigma$ in the literature), 2) Whether the PC population's fields have homogeneous or heterogeneous receptive fields, 3) In the case of PC targets with DoG tuning curves, the place cell's surround scale s i.e. the ratio between the inhibitory and excitatory Gaussian's standard deviations (often denoted $\sigma_I \stackrel{\text{def}}{=} s\,\sigma_E = s\sigma$ in the literature), 4) The number of fields per place cell.

Evaluating the entire hyperparameter volume is computationally prohibitive, so we evaluated a subvolume we considered most consistent with previous approaches as well as most illustrative of previous approaches' limitations; complete details of every run sweep configurations are provided in Appendix C.

To evaluate whether a network learns to optimally encode its position, we measured the network's position decoding error using the same methods as prior works [52, 42]. Specifically, we computed position decoding error using the networks' output Cartesian positions (if trained on Cartesian position targets) or using the networks' decoded positions based on their predicted PC population activity (if trained on PC targets). Any network was considered to have achieved optimal position encoding if its position decoding error fell below 6 cm; this threshold was chosen based on noise inherent in the position decoding algorithm.

In total, we trained 415 networks and found that almost every hyperparameter configuration succeeds in learning to path integrate (Fig. 2 A), but few learn lattice cells (hexagonal or rectangular) representations (Fig. 2 B). This is consistent with earlier work [29, 30] demonstrating that networks can learn to path integrate and solve other navigational problems (e.g. estimating which of several environments correspond to the current location) without lattice cells emerging as a solution.

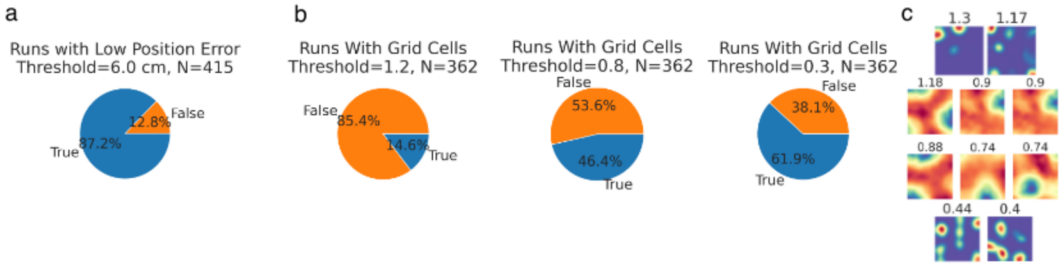

Figure 2: (a) Almost all configurations of loss function, target encoding and hyperparameter choices leads to networks that solve the path integration problem (i.e. achieve low position decoding error) but (b) very few networks learn grid-like cells. We use three different grid score thresholds: 0.3, 0.8 and 1.2. (c) As an important disclaimer, grid scores fall far short of an ideal metric to identify grid cells: They pick up on spurious features of ratemaps like triangular symmetry. Unlike previous approaches that used lax thresholds like 0.3, we recommend a more stringent threshold of 1.2.

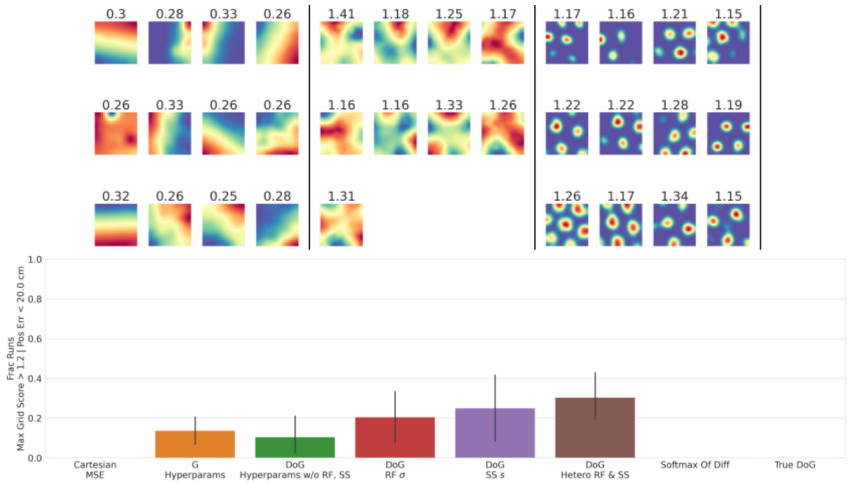

Figure 3: (a) Rate maps of highest-scoring units in deep networks trained on i) Cartesian readouts, ii) Gaussian readouts, iii) specifically selected (tuned) DoG readouts. i)-ii) do not learn any lattice cells. (b) Only networks trained on Difference-of-Gaussian (DoG) place cell tuning curves display lattice-like cells.

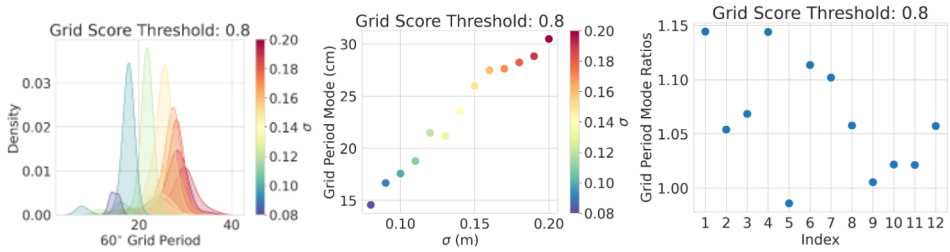

Figure 4: (a) Over a wide sweep of DoG receptive fields σ, grid scores are unimodal, meaning multiple grid modules do not emerge. (b) The chosen receptive field σ determines the grid period mode, with slope predicted by Fourier analysis. (c) Even when comparing grid score modes from runs with different σ values, adjacent grid period modes follow an arithmetic progression, not a geometric progression.

4 Lattice cell emergence requires a specific choice of supervised target encoding and is highly sensitive to small changes

We next sought to characterize when lattice cells are learnt if the target encodings are changed. We tested three different encodings of position in 2d space: a) cartesian readouts b) Gaussian readouts c) specifically selected (tuned) Difference-of-Gaussian (DoG) readouts. Delving into our previous large-scale hyperparameter sweeps, we found that (cf. Fig.3) only the tuned DoG readouts led to the formation of grid cells; Cartesian position, Gaussian place cells, and other possible supervised targets did not result in lattices (Fig. 3ab).

5 Grid periods are determined by the programmer's supervised target not the task, are not invariant, and multiple modules do not usually emerge

To demonstrate that the programmer's choice of DoG tuning affects most aspects of the formed representations, we swept over different scales of the place cell DoG and found that the period of the formed grid-like representation is completely determined by the scale of the DoG, Fig.4. Almost all runs had a unimodal distribution of grid periods, suggesting that the network had learnt only 1 module of grid cells, Fig.4(a). The period of this single module in every run increased monotonically

with the width of the DoG place cell, Fig.4(b) and lastly, the period ratios computed from the peaks of successive peaks in Fig.4(a) lead to non-biological values.

6 Fourier analysis explains the preceding empirical results

Our hyperparameter sweeps were guided by earlier Fourier analysis carried out by [7, 13, 53]. We restate the essence of the analysis here, and explain why it predicts these results. For a Difference-of-Gaussians (DoG) place cell tuning curve

$$f(\Delta x) = \alpha_E \exp\Big(-\frac{(\Delta x)^2}{2\sigma_E^2}\Big) - \alpha_I \exp\Big(-\frac{(\Delta x)^2}{2\sigma_I}\Big) \tag{1}$$

the Fourier transform is given by

$$\tilde{f}(k) = \alpha_E \sigma_E \exp\left(-\frac{\sigma_E^2 k^2}{2}\right) - \alpha_I \sigma_I \exp\left(-\frac{\sigma_I^2 k^2}{2}\right) \tag{2}$$

For (linearized) dynamics that approximate $\dot{x} \sim -x + f(x)$, the solution will be periodic if $\tilde{f}(k)$ contains sufficient power at its maxima, given by $[k^*]^2 = \frac{2}{\sigma_E^2 - \sigma_I^2} \log\left(\alpha_E \sigma_E^3 / \alpha_I \sigma_I^3\right)$ and that $\tilde{f}(k^*) > 1$. In particular, the inhibitory surround contained in $f(\Delta x)$, with strength σ_I, is key to pattern formation. With $\sigma_I \to \infty$ or $\alpha_I \to 0$, the maxima is at the origin, resulting in no pattern formation.

The theory laid out in [53] (with similar results first shown by [7] for pattern formation in attractor networks of grid cells and then by [13] for non-negative PCA of place cell targets) shows that the Fourier transform of the place cell activity (in particular, its second-moment matrix) is what sets the activity of the RNN units. Gaussian place cell targets have $\alpha_I = 0$, placing the peak in their Fourier spectrum at the origin, thereby causing DC modes to dominate and no lattices to form, regardless of the width Gaussian tuning curve's receptive field σ_E. In order to learn periodic lattice representations, the Fourier transform of the place cell tuning curve must contain a maxima with sufficient power at a non-zero value. This requires networks to be trained on PC targets with difference-of-Gaussian tuning curves, and even then, only a narrow range of scale parameters (σ, s) will work.

This prior work tells us that we should *not* expect lattice cells, square or hexagonal, in any network trained on non-DoG targets, an expectation that agrees with our large-scale hyperparameter sweeps. This also raises the question of how previous papers [1, 52, 53] obtained lattice cells in task-trained LSTM and RNN networks using Gaussian place cell targets since the analysis suggests such networks should not exist and and our large-scale hyperparameter sweeps are unable to reproduce them.

7 Grid cells disappear under more realistic place cell tuning curves

The Fourier analysis in the above argument is only applicable when the population of place cells encode the readout in an approximately translationally-invariant manner; this happens when the place cell population is distributed isotropically in the room, and every place cell has a single peak of the same scale. However, experimentally recorded place cells differ significantly from overly simplified single-scale, single-field homogeneous Gaussian or DoG functions: many place cells have multiple fields, often each with a distinct scale [46, 14, 54]. This naturally leads to the question: Will DoG place cell targets with heterogeneous scales or multiple fields per place cell still produce grid cells?

7.1 Heterogeneous scales in Difference-of-Gaussian (DoG) Place Cell (PC) Targets

We first tested what effect heterogeneous DoG scales have on the formation of grid cells. We trained ideal grid-forming networks (RNN, ReLU, $\sigma = 0.12$ cm, $s = 2.0$, 4 seeds) on heterogeneous DoG PC supervised targets with receptive field $\sigma \sim Uniform(0.06, 0.18)$ cm and surround scale $s \sim Uniform(1.5, 2.5)$; these distributions were chosen to ensure the expected values were ideal for grid cell formation. We found heterogeneity in the PC scales prevents the formation of grid cells (Fig.

5). Specifically, heterogeneity causes a clear decrease in the bulk of the networks' grid scores, as well as a decrease in the high grid score tail. Of the few remaining units (13 / 2048) with high grid scores, the distribution of grid periods remained unimodal (Fig. 5 center), showing that grids of multiple scales are not learnt. The networks trained on heterogeneous PC targets achieved equalivalent position decoding error as networks trained on homogeneous PC targets, further demonstrating that optimal position encoding does not require lattice representations.

8 Lattice cell emergence relies on undescribed implementation details

Several of the preceding papers [52, 42, 53] purport to use Difference-of-Gaussian (DoG) tuning curves (Eqn. 1) for the supervised PC targets, but in trying to replicate their results, we discovered their code actually uses something different: a Difference-of-Softmaxes[1] (DoS). Specifically, the tuning curves are given by:

$$f(\Delta x_p) = \text{Softmax}\Big(-\frac{(\Delta x_p)^2}{2\sigma_E^2}\Big) - \text{Softmax}\Big(-\frac{(\Delta x_p)^2}{2\sigma_I^2}\Big) \tag{3}$$

where the softmaxes are taken over the place cell population (indexed here by p). This seemingly minor implementation detail proves critical to the emergence of grid cells, yet is not mentioned in any of the main texts or supplements (to the best of our knowledge). To understand why, recall from Section 5 that in order for lattice cells to emerge, Fourier power (i.e. the eigenvalues of the place cell second-moment matrix $\Sigma = P^T P$) must peak on an annulus of a sufficiently large radius; if the radius is too small, lattice cells will not emerge. By using a DoS instead of a DoG, the maximal eigenvalues of the second-moment matrix fall on an annulus with radius 3 (Fig. 6a), which is clearly separated from the origin, giving rise to lattice cells. But if a DoG was truly used, the radius of the annulus would be 1 (Fig. 6c), too small for lattice cells to emerge. We trained ideal grid-forming ReLU RNN networks on supervised PC targets with genuine DoG tuning curves, sweeping the receptive field σ and surround scale s. We found that using true Difference-of-Gaussian tuning curves did not result in grid cells (Fig. 6b). Of the few grid cells that did emerge, their periodicity was still unimodal.

9 How do path-integrating deep networks achieve high neural predictivity of mouse MEC data?

We conclude by introducing a puzzle. A recent NeurIPS 2021 spotlight [42] claimed that networks trained on single-field single-scale "difference-of-Gaussian" PC targets explain variance in mouse

[1] `https://github.com/ganguli-lab/grid-pattern-formation/blob/master/place_cells.py#L51-L56`

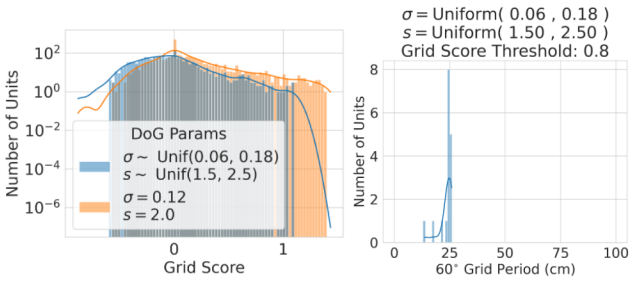

Figure 5: **Heterogeneous place cells prevent the formation of grid cells.** Even with difference-of-Gaussian (DoG) place cell targets and non-negative network nonlinearities, adding heterogeneity to the DoG scale hyperparameters (receptive field σ and surround scale s) causes grid cells to disappear (left). The few remaining grid cells still exhibit a single period, rather than multiple periods (right). 3 runs per condition.

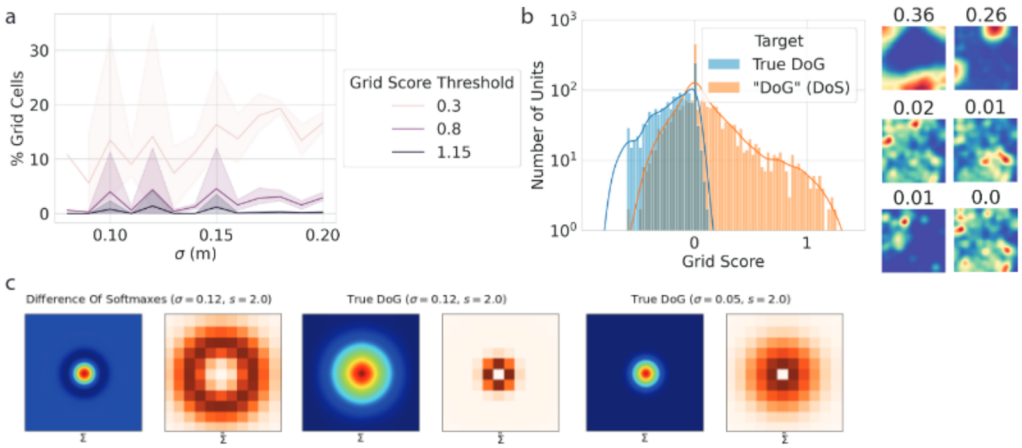

Figure 6: **Other changes to difference of gaussians affects the formation of grid cells:** (a) The existence of grid solutions even with DOG readout encodings is highly sensitive to parameters such as the target function receptive field σ, and small alterations result in the disappearance of grid cells regardless of grid score threshold. (b) (left) Comparison of grid scores of trained networks with true DoG shows that the DoG with normalization as in [52, 42] is important for grid cell formation, (right) Ratemaps of the highest scoring cells from the true DoG networks. (c) Empirically computing the Fourier transform of the place cell second-moment matrix explains why the particular choice of normalization affects representations.

MEC neural activity at nearly 100% of the variance explained by other mice's MEC neural activity. However, given our results demonstrating that these networks display properties inconsistent with biological grid cells (e.g., single grid periods, square lattices), and require supervised targets inconsistent with place cells, how are these networks able to predict mouse MEC neural activity so well?

The data preprocessing and analysis code is not public, so we are unable to investigate in detail. However, we offer a conjecture with preliminary supporting evidence. The analysis of [42] linearly regressed rate maps from one agent (mouse or network) onto rate maps from another mouse, and used Pearson correlation as a measure of "neural predictivity." We conjectured that different combinations of architectures and activations achieve different neural predictivity scores because different architecture-activation combinations learn dynamical systems of different intrinsic dimensionalities that then provide richer or poor bases for linear regressions.

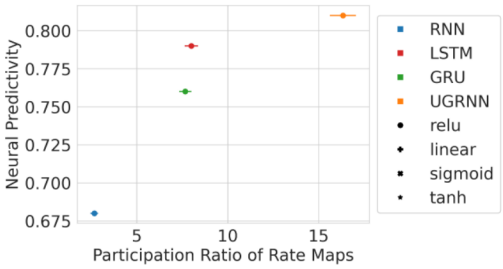

Figure 7: **Networks with higher (lower) dimensional rate maps display higher (lower) "neural predictivity" of mouse MEC rate maps.** This suggests that linear regression-based comparisons of artificial and biological networks may find "better" networks simply because those networks provide higher dimensional bases for the regressions. Each dot is an architecture-activation pair, with participation ratio [35] averaged over 5 runs. Neural predictivity scores from Fig 2A of [42].

To explore our conjecture, we trained the networks considered in [42]: 4 architectures (RNN, LSTM, GRU, UGRNN) and 4 activations (Linear, Sigmoid, Tanh, ReLU), adding 5 random seeds (0 through 4) per architecture-activation pair. For each trained network, we computed a standard linear measure of the intrinsic dimensionality (called participation ratio [35]) of the network's rate maps, and then plotted each network's participation ratio against the published neural predictivity score. We chose a linear measure of the networks' intrinsic dimensionalities since linear regression cannot fit nonlinear patterns in the data.

We found a clear trend that networks with higher (lower) dimensional rate maps have higher (lower) neural predictivity scores (Fig 7).

We caution that this correlation between network dimensionality and neural predictivity is not strong evidence, but we are unable to investigate further without access to preprocessing and analysis code. If our conjecture is correct, it raises a serious concern that linear regression-based comparisons of biological networks and deep artificial neural networks, widely used and cited in vision, language and audition, may also be misleading. DL-based models may not actually be close models of biological neural circuits, but rather may merely provide higher-dimensional bases than alternative models and thus trivially achieve higher correlation scores.

10 Discussion

For studies that use deep networks as models of the brain, there is a fundamental obstacle to making the claim that a given loss function is what the brain is optimizing for: If we know the responses of a significant fraction of units from biological networks performing a certain task, we cannot infer the loss function that it is optimizing since in principle, numerous different loss functions can have the same minima Fig.8(top); conversely, given a reasonable loss function that we guess based on an organism's ecological niche, we cannot infer a single solution (and thus build truly predictive single-cell tuning models), since there exist several minima to that objective Fig.8(bottom).

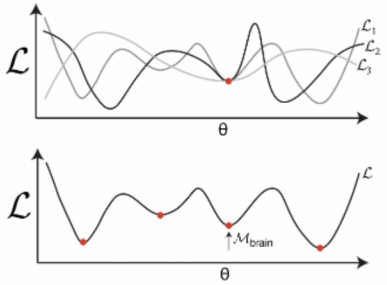

Figure 8: Challenges in achieving the two central claims of recent DL models of neuroscience: Top: Building a model that replicates observed neural responses does not guarantee that the loss function used is the brain's objective, as multiple objectives can share a solution. Bottom: A plausible loss function need not give the solution the brain has selected because of other constraints.

To break this degeneracy of multiple minima and arrive at a truly predictive models or a better understanding of the brain's objective functions, we must acknowledge, understand, and model the host of specific implicit biases and constraints present in the biological system we are trying to model. It is untenable to expect success without. This is what we refer to as an informal neural 'no-free-lunch'.

What can we learn from DL models about brain circuits without considering and studying biological inductive biases? Population-wide low-dimensional latent representations and dynamics that arise as necessary for solving difficult problems are possibly robust and sufficiently invariant features that might be predicted without the addition of detailed biological constraints. This can explain the success of the population-level analyses of the visual pathway [31, 2], as well as the population-level low-dimensional dynamics of circuits solving inference tasks [57]. In these cases, the emergent low-dimensional solutions are fundamental features of any system that solve the task, and by construction need not be specific to brains.

Coming back to grid cells, we have shown that they do not generically arise in networks trained to path-integrate. What other ingredient(s) might we need to add to the task of path integration so that we always get grid cells? There are many properties that a population code needs to have for the optimal solution to be grid-cell like. Based on theoretical arguments outlined before, we prescribe a (hypothetically sufficient) set of these properties informed by biology and existing theoretical results on the grid cell system: 1) non-negative activations; 2) equivariant population responses (i.e. the population response always lies on a hypersphere); 3) a path integrating (PI) code or in other words, translation invariant representations; 4) a high representational capacity [6, 55, 38]; 5) error correcting capabilities [6, 55]; and finally, 6) low spatial information per cell: in other words, the spatial information should be equally distributed across all cells equally.

References

[1] Andrea Banino, Caswell Barry, Benigno Uria, Charles Blundell, Timothy Lillicrap, Piotr Mirowski, Alexander Pritzel, Martin J. Chadwick, Thomas Degris, Joseph Modayil, Greg Wayne, Hubert Soyer, Fabio Viola, Brian Zhang, Ross Goroshin, Neil Rabinowitz, Razvan Pascanu, Charlie Beattie, Stig Petersen, Amir Sadik, Stephen Gaffney, Helen King, Koray Kavukcuoglu, Demis Hassabis, Raia Hadsell, and Dharshan Kumaran. Vector-based navigation using grid-like representations in artificial agents. *Nature*, 557(7705):429–433, May 2018.

[2] Pouya Bashivan, Kohitij Kar, and James J. DiCarlo. Neural population control via deep image synthesis. *Science*, 364(6439):eaav9436, May 2019. Publisher: American Association for the Advancement of Science.

[3] Marc G. Bellemare, Salvatore Candido, Pablo Samuel Castro, Jun Gong, Marlos C. Machado, Subhodeep Moitra, Sameera S. Ponda, and Ziyu Wang. Autonomous navigation of stratospheric balloons using reinforcement learning. *Nature*, 588(7836):77–82, December 2020. Number: 7836 Publisher: Nature Publishing Group.

[4] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners. *arXiv:2005.14165 [cs]*, July 2020. arXiv: 2005.14165.

[5] Yoram Burak and Ila Fiete. Do We Understand the Emergent Dynamics of Grid Cell Activity? *Journal of Neuroscience*, 26(37):9352–9354, September 2006. Publisher: Society for Neuroscience Section: Journal Club.

[6] Yoram Burak and Ila R Fiete. Unpublished observations. 2008.

[7] Yoram Burak and Ila R. Fiete. Accurate Path Integration in Continuous Attractor Network Models of Grid Cells. *PLOS Computational Biology*, 5(2):e1000291, February 2009. Publisher: Public Library of Science.

[8] Malcolm G. Campbell, Samuel A. Ocko, Caitlin S. Mallory, Isabel I. C. Low, Surya Ganguli, and Lisa M. Giocomo. Principles governing the integration of landmark and self-motion cues in entorhinal cortical codes for navigation. *Nature Neuroscience*, 21(8):1096–1106, August 2018. Number: 8 Publisher: Nature Publishing Group.

[9] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *NIPS Workshop Deep Learning and Representation Learning*, December 2014. Number: arXiv:1412.3555 arXiv:1412.3555 [cs].

[10] Jasmine Collins, Jascha Sohl-Dickstein, and David Sussillo. Capacity and Trainability in Recurrent Neural Networks. *International Conference on Learning Representations*, March 2017. Number: arXiv:1611.09913 arXiv:1611.09913 [cs, stat].

[11] Christopher J Cueva and Xue-Xin Wei. Emergence of grid-like representations by training recurrent neural networks to perform spatial localization. *International Conference on Learning Representations*, page 19, 2018.

[12] Alex Davies, Petar Veličković, Lars Buesing, Sam Blackwell, Daniel Zheng, Nenad Tomašev, Richard Tanburn, Peter Battaglia, Charles Blundell, András Juhász, Marc Lackenby, Geordie Williamson, Demis Hassabis, and Pushmeet Kohli. Advancing mathematics by guiding human intuition with AI. *Nature*, 600(7887):70–74, December 2021. Number: 7887 Publisher: Nature Publishing Group.

[13] Yedidyah Dordek, Daniel Soudry, Ron Meir, and Dori Derdikman. Extracting grid cell characteristics from place cell inputs using non-negative principal component analysis. *eLife*, 5:e10094, March 2016. Publisher: eLife Sciences Publications, Ltd.

[14] Tamir Eliav, Shir R. Maimon, Johnatan Aljadeff, Misha Tsodyks, Gily Ginosar, Liora Las, and Nachum Ulanovsky. Multiscale representation of very large environments in the hippocampus of flying bats. *Science*, 372(6545):eabg4020, May 2021. Publisher: American Association for the Advancement of Science.

[15] Jeffrey L. Elman. Finding Structure in Time. *Cognitive Science*, 14(2):179–211, 1990. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1207/s15516709cog1402_1.

[16] Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Firdaus Janoos, Larry Rudolph, and Aleksander Madry. Implementation Matters in Deep Policy Gradients: A Case Study on PPO and TRPO. *arXiv:2005.12729 [cs, stat]*, May 2020. arXiv: 2005.12729.

[17] Ila R Fiete, Yoram Burak, and Ted Brookings. What grid cells convey about rat location. *J Neurosci*, 28(27):6858–71, Jul 2008.

[18] Mark C Fuhs and David S Touretzky. A spin glass model of path integration in rat medial entorhinal cortex. *J Neurosci*, 26(16):4266–4276, 2006.

[19] Richard J. Gardner, Erik Hermansen, Marius Pachitariu, Yoram Burak, Nils A. Baas, Benjamin A. Dunn, May-Britt Moser, and Edvard I. Moser. Toroidal topology of population activity in grid cells. *Nature*, 602(7895):123–128, February 2022. Number: 7895 Publisher: Nature Publishing Group.

[20] Richard J Gardner, Li Lu, Tanja Wernle, May-Britt Moser, and Edvard I Moser. Correlation structure of grid cells is preserved during sleep. *Nat Neurosci*, 22(4):598–608, 04 2019.

[21] Richard J. Gardner, Li Lu, Tanja Wernle, May-Britt Moser, and Edvard I. Moser. Correlation structure of grid cells is preserved during sleep. *Nature Neuroscience*, 22(4):598–608, April 2019. Number: 4 Publisher: Nature Publishing Group.

[22] Joshua I. Glaser, Ari S. Benjamin, Raeed H. Chowdhury, Matthew G. Perich, Lee E. Miller, and Konrad P. Kording. Machine Learning for Neural Decoding. *eNeuro*, 7(4), July 2020. Publisher: Society for Neuroscience Section: Research Article: Methods/New Tools.

[23] Torkel Hafting, Marianne Fyhn, Sturla Molden, May-Britt Moser, and Edvard I. Moser. Microstructure of a spatial map in the entorhinal cortex. *Nature*, 436(7052):801–806, August 2005. Number: 7052 Publisher: Nature Publishing Group.

[24] Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep Reinforcement Learning that Matters. *arXiv:1709.06560 [cs, stat]*, January 2019. arXiv: 1709.06560.

[25] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Lecture 6e-RMSProp.

[26] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, November 1997.

[27] Andrew Ilyas, Logan Engstrom, Shibani Santurkar, Dimitris Tsipras, Firdaus Janoos, Larry Rudolph, and Aleksander Madry. A Closer Look at Deep Policy Gradients. *arXiv:1811.02553 [cs, stat]*, May 2020. arXiv: 1811.02553.

[28] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen,

David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, August 2021. Number: 7873 Publisher: Nature Publishing Group.

[29] Ingmar Kanitscheider and Ila Fiete. Training recurrent networks to generate hypotheses about how the brain solves hard navigation problems. *arXiv:1609.09059 [q-bio]*, December 2016. arXiv: 1609.09059.

[30] Ingmar Kanitscheider and Ila Fiete. Emergence of dynamically reconfigurable hippocampal responses by learning to perform probabilistic spatial reasoning. preprint, Neuroscience, December 2017.

[31] Alexander J. E. Kell, Daniel L. K. Yamins, Erica N. Shook, Sam V. Norman-Haignere, and Josh H. McDermott. A Task-Optimized Neural Network Replicates Human Auditory Behavior, Predicts Brain Responses, and Reveals a Cortical Processing Hierarchy. *Neuron*, 98(3):630–644.e16, May 2018.

[32] Seyed-Mahdi Khaligh-Razavi and Nikolaus Kriegeskorte. Deep Supervised, but Not Unsupervised, Models May Explain IT Cortical Representation. *PLoS Computational Biology*, 10(11):e1003915, November 2014.

[33] Timothy D. Kim, Thomas Z. Luo, Jonathan W. Pillow, and Carlos D. Brody. Inferring Latent Dynamics Underlying Neural Population Activity via Neural Differential Equations. In *Proceedings of the 38th International Conference on Machine Learning*, pages 5551–5561. PMLR, July 2021. ISSN: 2640-3498.

[34] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations*, January 2017. Number: arXiv:1412.6980 arXiv:1412.6980 [cs].

[35] Ashok Litwin-Kumar, Kameron Decker Harris, Richard Axel, Haim Sompolinsky, and L. F. Abbott. Optimal Degrees of Synaptic Connectivity. *Neuron*, 93(5):1153–1164.e7, March 2017.

[36] Jesse A. Livezey, Kristofer E. Bouchard, and Edward F. Chang. Deep learning as a tool for neural data analysis: Speech classification and cross-frequency coupling in human sensorimotor cortex. *PLOS Computational Biology*, 15(9):e1007091, September 2019. Publisher: Public Library of Science.

[37] Kevin Luxem, Falko Fuhrmann, Johannes Kürsch, Stefan Remy, and Pavol Bauer. Identifying Behavioral Structure from Deep Variational Embeddings of Animal Motion. Technical report, bioRxiv, October 2020. Section: New Results Type: article.

[38] A. Mathis, A. Herz, and M. Stemmler. Optimal population codes for space: grid cells outperform place cells. *Neural Comp.*, 24:2280–2317, 2012.

[39] Alexander Mathis, Andreas V. M. Herz, and Martin Stemmler. Optimal Population Codes for Space: Grid Cells Outperform Place Cells. *Neural Computation*, 24(9):2280–2317, September 2012.

[40] Alexander Mathis, Pranav Mamidanna, Kevin M. Cury, Taiga Abe, Venkatesh N. Murthy, Mackenzie Weygandt Mathis, and Matthias Bethge. DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience*, 21(9):1281–1289, September 2018. Number: 9 Publisher: Nature Publishing Group.

[41] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan

Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, February 2015.

[42] Aran Nayebi, Alexander Attinger, Malcolm G. Campbell, Kiah Hardcastle, Isabel I.C. Low, Caitlin S. Mallory, Gabriel C. Mel, Ben Sorscher, Alex H. Williams, Surya Ganguli, Lisa M. Giocomo, and Daniel L.K. Yamins. Explaining heterogeneity in medial entorhinal cortex with task-driven neural networks. preprint, Neuroscience, November 2021.

[43] J. O'Keefe and J. Dostrovsky. The hippocampus as a spatial map: Preliminary evidence from unit activity in the freely-moving rat. *Brain Research*, 34:171–175, 1971. Place: Netherlands Publisher: Elsevier Science.

[44] Chethan Pandarinath, Daniel J. O'Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D. Stavisky, Jonathan C. Kao, Eric M. Trautmann, Matthew T. Kaufman, Stephen I. Ryu, Leigh R. Hochberg, Jaimie M. Henderson, Krishna V. Shenoy, L. F. Abbott, and David Sussillo. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature Methods*, 15(10):805–815, October 2018. Number: 10 Publisher: Nature Publishing Group.

[45] Talmo D. Pereira, Diego E. Aldarondo, Lindsay Willmore, Mikhail Kislin, Samuel S.-H. Wang, Mala Murthy, and Joshua W. Shaevitz. Fast animal pose estimation using deep neural networks. *Nature Methods*, 16(1):117–125, January 2019. Number: 1 Publisher: Nature Publishing Group.

[46] P. D. Rich, H.-P. Liaw, and A. K. Lee. Large environments reveal the statistical structure governing hippocampal representations. *Science*, 345(6198):814–817, August 2014.

[47] Blake A. Richards, Timothy P. Lillicrap, Philippe Beaudoin, Yoshua Bengio, Rafal Bogacz, Amelia Christensen, Claudia Clopath, Rui Ponte Costa, Archy de Berker, Surya Ganguli, Colleen J. Gillon, Danijar Hafner, Adam Kepecs, Nikolaus Kriegeskorte, Peter Latham, Grace W. Lindsay, Kenneth D. Miller, Richard Naud, Christopher C. Pack, Panayiota Poirazi, Pieter Roelfsema, João Sacramento, Andrew Saxe, Benjamin Scellier, Anna C. Schapiro, Walter Senn, Greg Wayne, Daniel Yamins, Friedemann Zenke, Joel Zylberberg, Denis Therien, and Konrad P. Kording. A deep learning framework for neuroscience. *Nature Neuroscience*, 22(11):1761–1770, November 2019. Number: 11 Publisher: Nature Publishing Group.

[48] Muhammad Saif-ur Rehman, Robin Lienkämper, Yaroslav Parpaley, Jörg Wellmer, Charles Liu, Brian Lee, Spencer Kellis, Richard Andersen, Ioannis Iossifidis, Tobias Glasmachers, and Christian Klaes. SpikeDeeptector: a deep-learning based method for detection of neural spiking activity. *Journal of Neural Engineering*, 16(5):056003, July 2019. Publisher: IOP Publishing.

[49] Andrew Saxe, Stephanie Nelli, and Christopher Summerfield. If deep learning is the answer, what is the question? *Nature Reviews Neuroscience*, 22(1):55–67, January 2021. Number: 1 Publisher: Nature Publishing Group.

[50] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, January 2016. Number: 7587 Publisher: Nature Publishing Group.

[51] Trygve Solstad, Charlotte N. Boccara, Emilio Kropff, May-Britt Moser, and Edvard I. Moser. Representation of Geometric Borders in the Entorhinal Cortex. *Science*, 322(5909):1865–1868, December 2008. Publisher: American Association for the Advancement of Science.

[52] Ben Sorscher, Gabriel C Mel, Surya Ganguli, and Samuel A Ocko. A unified theory for the origin of grid cells through the lens of pattern formation. *Advances in Neural Information Processing Systems*, page 18, 2019.

[53] Ben Sorscher, Gabriel C. Mel, Samuel A. Ocko, Lisa Giocomo, and Surya Ganguli. A unified theory for the computational and mechanistic origins of grid cells. Technical report, bioRxiv, December 2020. Section: New Results Type: article.

[54] Bryan C. Souza, Rodrigo Pavão, Hindiael Belchior, and Adriano B. L. Tort. On Information Metrics for Spatial Coding. *Neuroscience*, 375:62–73, April 2018.

[55] Sameet Sreenivasan and Ila Fiete. Grid cells generate an analog error-correcting code for singularly precise neural computation. *Nat Neurosci*, 14(10):1330–7, Sep 2011.

[56] Hanne Stensola, Tor Stensola, Trygve Solstad, Kristian Frøland, May-Britt Moser, and Edvard I. Moser. The entorhinal grid map is discretized. *Nature*, 492(7427):72–78, December 2012. Number: 7427 Publisher: Nature Publishing Group.

[57] David Sussillo and Omri Barak. Opening the Black Box: Low-Dimensional Dynamics in High-Dimensional Recurrent Neural Networks. *Neural Computation*, 25(3):626–649, March 2013.

[58] David Sussillo, Rafal Jozefowicz, L. F. Abbott, and Chethan Pandarinath. LFADS - Latent Factor Analysis via Dynamical Systems. Technical Report arXiv:1608.06315, arXiv, August 2016. arXiv:1608.06315 [cs, q-bio, stat] type: article.

[59] S.G. Trettel, J.B. Trimper, E. Hwaun, I.R. Fiete, and L.L. Colgin. Grid cell co-activity patterns during sleep reflect spatial overlap of grid fields during active behaviors. *Nat Neurosci*, 22(4):609–617, 04 2019.

[60] George Tucker, Surya Bhupatiraju, Shixiang Gu, Richard E. Turner, Zoubin Ghahramani, and Sergey Levine. The Mirage of Action-Dependent Baselines in Reinforcement Learning. *arXiv:1802.10031 [cs, stat]*, November 2018. arXiv: 1802.10031.

[61] James C. R. Whittington, Timothy H. Muller, Shirley Mark, Guifen Chen, Caswell Barry, Neil Burgess, and Timothy E. J. Behrens. The Tolman-Eichenbaum Machine: Unifying Space and Relational Memory through Generalization in the Hippocampal Formation. *Cell*, 183(5):1249–1263.e23, November 2020.

[62] K.J. Yoon, M.A. Buice, R. Barry, C.and Hayman, N. Burgess, and I.R. Fiete. Specific evidence of low-dimensional continuous attractor dynamics in grid cells. *Nat Neurosci*, 16(8):1077–84, Aug 2013.

11 Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section **??**.
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]