CURRICULUM LEARNING AS A TOOL TO UNCOVER LEARNING PRINCIPLES IN THE BRAIN

Anonymous authors
Paper under double-blind review

ABSTRACT

We present a novel approach to use curricula to identify principles by which a system learns. Previous work in curriculum learning has focused on how curricula can be designed to improve learning of a model on particular tasks. We consider the inverse problem: what can a curriculum tell us about how a learning system acquired a task? Using recurrent neural networks (RNNs) and models of common experimental neuroscience tasks, we demonstrate that curricula can be used to differentiate learning principles using target-based and a representation-based loss functions as use cases. In particular, we compare the performance of RNNs using target-based learning rules versus those using representational learning rules on three different curricula in the context of two tasks. We show that the learned state-space trajectories of RNNs trained by these two learning rules under all curricula tested are indistinguishable. However, by comparing learning times during different curricula, we can disambiguate the learning rules and challenge traditional approaches of interrogating learning systems. Although all animals in neuroscience lab settings are trained by curriculum-based procedures called shaping, almost no behavioral or neural data are collected or published on the relative successes or training times under different curricula. Our results motivate the systematic collection and curation of data during shaping by demonstrating curriculum learning in RNNs as a tool to probe and differentiate learning principles used by biological systems, over conventional statistical analyses of learned state spaces.

1 INTRODUCTION

The brain is thought to be the ultimate learner as it learns from few examples, solves unstructured problems, and has an impressive task repertoire. Understanding how it achieves these learning feats could lead us to better artificial intelligence (AI) algorithms (Hassabis et al., 2017; Macpherson et al., 2021). Interrogating learning in the brain, however, poses significant challenges both experimentally and computationally. Experimentally, measuring the entire synaptic connectivity map or connectome–a primary substrate of learning–is only just approaching reality for small brains (Zador et al., 2012; Dorkenwald et al., 2020). Even so, the emerging connectome data provides just a snapshot in time whereas learning concerns connectome dynamics, and this temporal data is still far off. Computationally, it is unclear how a sequence of brain-wide connectivities would be registered in order to reveal how animals learn different tasks (Babai, 2015). Techniques to measure neural activity from the brains of experimental animals are in a golden age, measuring exponentially more neurons–and over longer periods of time–with each passing year (Stevenson & Kording, 2011). Consistently measuring the same population of active neurons over many trials during complex tasks remains challenging. Such neural data often require subsampling or are averaged over trials just to correlate with behavioral performance, which further complicates its relationship to connectivity Hong & Lieber (2019). Yet, even with perfect neural activity data throughout training, it is still an open computational problem to infer how the connectivity changes during training (Magrans de Abril et al., 2018). Uncovering the key substrates linking structural (connectome), dynamic (neural activity), and behavioral elements in biological brains could help us develop better AI algorithms with brain-like learning properties.

In this paper, we use behavioral dynamics to infer learning principles. Measuring behavior in animals is comparatively easier than neural recordings or synaptic strengths. The challenge, however, is for

theory to attribute individual behaviors to specific learning rules. We suggest the use of curricula to get information from behavioral dynamics about underlying learning principles.

A curriculum is traditionally a schedule for information to be presented to a student learner. Usually, a curriculum is designed to facilitate learning, e.g., by increasing the number or complexity of items taught during a course. In the machine learning (ML) framework we are considering, the student learner is a neural network, specifically an RNN. Recurrence in network models brings some key advantages we exploit here–briefly, ability to produce dynamics and analogy to the biological brain's ubiquitous feedback connections (Yang et al., 2019; Singer, 2021; Ehrlich et al., 2021). Although the benefits of curricula in ML have been shown (Graves et al., 2017; Weinshall et al., 2018; Saglietti et al., 2021), curriculum learning has not been widely adopted for practical applications, with notable exceptions in robotics (James et al., 2019), and reinforcement learning (Taylor & Stone, 2009). In contrast, curriculum learning is of the utmost practical importance to experimental neuroscience – most animals in neuroscience lab settings are trained using curricula, a process called *shaping* in the literature (Pinto et al., 2018; Koay et al., 2021; Guo et al., 2014). Yet, very little experimental data on curricula exists in neuroscience, as most data are collected from "expert" animals after shaping.

Our goal is to classify learners–here, RNNs–based on the rules they use to learn different tasks using a set of pre-designed curricula. We build and analyze RNNs trained on two common experimental neuroscience tasks using three different curricula inspired by shaping procedures. In particular, we use the evidence accumulation task (Pinto et al., 2018; Stine et al., 2020) and delayed decision task (Romo et al., 1999; Constantinidis et al., 2018; Liu et al., 2014). Our curricula are directly inspired by shaping procedures in those works. We show, using only behavioral dynamics during the execution of different curricula, that it is possible to distinguish learning rules solely on the basis of outcomes: target-based learning from learning through representational constraints. Importantly, we find that all RNNs, regardless of curriculum or learning rule, are indistinguishable post-shaping by standard statistical analyses applied to neural dynamics in their trained state. Our results emphasize the importance of studying animals during shaping and the value of curriculum learning in RNNs as a hypothesis test-bed for probing learning principles in the biological brain.

1.1 Related work

Curriculum learning has long been relevant to the fields of AI/ML and neuroscience Wang et al. (2021).In the late 80s and 90s, curricula were used to learn difficult control problems or tasks in robotics and reinforcement learning (Selfridge et al. (1985), Schmidhuber (1991), Sanger (1994)). Around then, Elman (1993) noted that humans and animals utilize curricula and asked how it could benefit machines. Later, Bengio et al. (2009) formalized curriculum learning and related it to input complexity in a key paper on how learning could be optimized.

Despite the presence of curricula in ML literature for several decades, our work is the first we are aware of that uses curricula to characterize learning principles used by neural network models to learn tasks, analogously to experimental animals in lab settings. Recently, the question of identifying learning rules was studied by Nayebi et al. (2020). There, the authors utilize the full knowledge of network activations available to artificial systems to infer learning rules. Their work anticipates the quickly improving electrical recordings from neuroscience experiments. Here, however, we are interested in being able to glean learning principles solely from behavioral data which is presently measurable to neuroscientists. Ultimately, these two approaches may be used in complement to further improve resolution of learning principles.

Ashwood et al. (2020) also link behavioral data and learning rules using large-scale data from a left-right task performed by mice in the International Brain Lab (IBL). Importantly, the IBL task considered includes little to no time dependence. The mice are able to learn the task easily without a curriculum. The lack of time dependence in the task is reflected in the model presented in Ashwood et al. (2020), which does not consider history or state dependence. Our work is distinct in that we target complex tasks with temporal dependencies for which a) curricula are appropriate, and b) the analogous experiments require shaping procedures for the lab animal to learn. We therefore encompass history and state effects in our approach.

In this paper, we demonstrate that different learning rules and curricula converge to similar solutions in all cases (Figue 5). In Maheswaranathan et al. (2019), the authors claim that there is a universal solution for a given task that all learning algorithms converge to. Their approach does not incorporate

curricula; we posit that our work demonstrates a different, complementary invariance. Here, our goal is not to prove universality, but rather to identify learning mechanisms that may be inaccessible by only studying the trained or post-shaping state.

2 Models and Methods

2.1 Recurrent Neural Networks (RNNs)

In all experiments we use neural networks with $N = 350$ recurrently connected continuous firing-rate type leaky integrating model neurons. A neuron with index n connects to a neuron with index m through the recurrent weight w^{rec}_{nm} and input channel index j through input weight w^{inp}_{nj}. The internal state of neuron n, $\mathbf{x}_n$, is determined by:

$$\tau \frac{d\mathbf{x}_n}{dt} = w^{\text{rec}}_{nm}\mathbf{a}_m(t) + w^{\text{inp}}_{nj}\mathbf{v}_j(t) - \mathbf{x}_n \tag{1}$$

where $\mathbf{v}(t)$ is a time-dependent, task-relevant input at time t and $\mathbf{a}(t)$ is the activation function applied to the internal state–here, the hyperbolic tangent function: $\mathbf{a_n}(t) = \tanh(\mathbf{x}_n(t))$. τ is the time constant of the neuron, 10ms in this paper. We use the Euler method to calculate neural states with $dt = 1$ms.

We define the linear readout of a network at time t, $\mathbf{z}(t)$, as the weighted sum of the activations $\mathbf{a}(t)$ via weights w^{out}_n:

$$\mathbf{z}(t) = w^{\text{out}}_n \mathbf{a}_n(t) \tag{2}$$

Recurrent weights W^{rec} are initialized i.i.d from a Gaussian with mean 0 and variance $\frac{g^2}{N}$. The parameter setting the scale of the recurrent weights, $g = 1$. Input W^{inp} and linear readout weights W^{out} are each drawn from a uniform distribution from -1 to 1. Internal states, $\{\mathbf{x}\}$, are initialized from a Gaussian with mean 0 and variance 1.

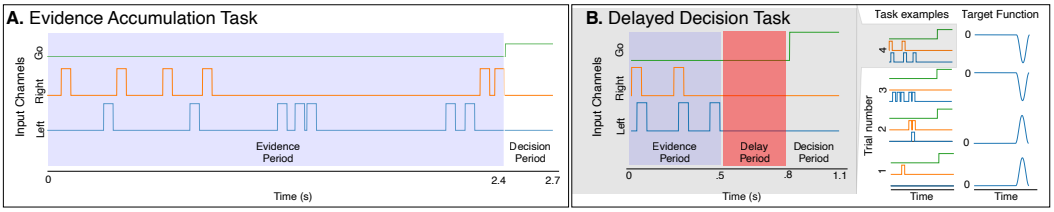

Figure 1: **Task Design.** Both tasks utilize three input channels. Two correspond to "left" (blue line) or "right" (orange line). The third (green line) sends the go cue, signaling the start of a decision period. Both evidence accumulation (abbreviated EA, left) and delayed decision (abbreviated DD, right) tasks contain cue periods and decision periods. During the cue period, input events or "cues" occur at times generated by a Poisson process in the left/right channels. In the decision period, a target bump reflects the channel with more events during the cue period. The EA task contains a 2.4s cue period, compared to 0.5s for the DD task. The DD task contains a delay period. No events are presented during this period.

2.2 Tasks

Our tasks are motivated by those commonly used in neuroscience for studying working memory and decision making in rodents, non-human primates, and humans (Pinto et al., 2018; Stine et al., 2020; Romo et al., 1999; Constantinidis et al., 2018; Liu et al., 2014). In the rodent version of this task, for instance, a mouse runs down a corridor with cues presented along the left and right. The number of left or right cues inform the mouse which of two turns it should make when the corridor ends, and is rewarded if it turns toward the side with more cues.

In our model tasks, we have a cue period similar to the corridor in the mouse task. Two input channels, corresponding to left and right cues in the mouse task, send pulsatile input events into the RNN units,

each with an amplitude of 0.25 and duration of 50ms. Cue start times for each channel are Poisson distributed. We use use Poisson rates λ of 150ms^{-1} and 300ms^{-1} for the two channels throughout. On each simulated trial, the λ for input channels are randomly swapped. The network also has a third input channel which produces a 0.25 amplitude "go cue" lasting 50ms in the decision period. The target function, z^{tar} for the linearly read out RNN-unit activations is 0 during the cue period. During the decision period, the target function is:

$$z^{\text{tar}}(t) = 2p\left[\frac{1+\cos\left(\frac{2\pi(t-t_{\text{start}})}{t_{\text{fin}}-t_{\text{start}}}\right)}{2}\right], \quad t \in (t_{\text{start}}, t_{\text{fin}}) \tag{3}$$

where t_{start} and t_{fin} mark the start and end of the decision period, and p is 1 or -1, based on which channel accumulated more cues.

We use two task variants: First is an evidence accumulation (EA) task (Figure 1A) to challenge primacy and recency biases. RNNs on this task, similar to lab animals, can sometimes overweigh early or recent evidence at the cost of running counts and relative discrepancy of cues between the two channels. Our second task is a delayed decision (DD) task (Figure 1B), featuring a delay period between the cue and decision periods to challenge temporal credit assignment, which is often problematic for network models to span.

Evidence Accumulation (EA) tasks consist of a 2.4s cue period followed by a 0.25s decision period.

Delayed Decision (DD) tasks consist of a 0.5s cue period, a 0.5s delay period, and a 0.25s decision period. The target function during the delay is 0 with no cues during this time.

2.3 Discrepancy

A key characterization of the two task variants is a quantity we refer to as *Discrepancy*. We define discrepancy as the instantaneous difference between the number of cues in the left and the right input channels. Formally, the set of all left event or "cue" times is $\{t_i^{\text{left}}\}_{i=1}^{K^{\text{left}}}$ where K^{left} is the total number of left events in the trial and t_i^{left} is the time of the ith left event. Discrepancy at time t is defined as:

$$D(t) = \sum_{i=1}^{K^{\text{left}}} H(t - t_i^{\text{left}}) - \sum_{i=1}^{K^{\text{right}}} H(t - t_i^{\text{right}}) \tag{4}$$

$D(t) < 0$ when there are more left cues than right. We define *absolute discrepancy* as $|D(t)|$ and the absolute discrepancy of a trial, $|D(t_{\text{final}})|$. For all trials, we enforce $|D(t_{\text{final}})| > 0$.

2.4 Learning principles

The goal of this paper is to use curriculum learning in RNN models of decision-making tasks as an analogy to shaping in animal neuroscience experiments, in order to identify learning principles employed by the brain. A key distinction for neuroscience is whether the brain learns mostly by rewarding and punishing behavioral outputs or by enforcing internal representations on its neural dynamics (Saxe et al., 2021; Bhand et al., 2011; Yamins et al., 2014). Here, in RNNs trained by backpropagation through time, we design separate loss functions mimicking either *output- or target-based learning* or *representational learning*. For output- or **target-based backpropagation (BPT)**, we define an L2 error between a target function for an idealized "behavior-like" output of the task, $\mathbf{z}^{\text{tar}}(t)$ and the RNN's overall output, $\mathbf{z}(t)$ (Eq. 2):

$$\mathcal{L}^{\text{tar}}(t_{\text{T}}) = \sum_{s=0}^{T} \left[\mathbf{z}(t_s) - \mathbf{z}^{\text{tar}}(t_s)\right]^2, \tag{5}$$

where t_T is the weight-update time and t_0 is the trial start time. $t_T = t_{\text{final}}$, i.e., weights are updated only at the ends of trials. For **representational backpropagation (BPR)**, we incorporate discrepancy into the loss function, forcing the network to directly represent in its activations the instantaneous difference between the cues in the two input channels throughout a trial. To do this, we first use

representational weights W^{rep} with elements w_n^{rep} that linearly read out discrepancy $\hat{D}(t)$ from the RNN units $\mathbf{a}_n$, $\hat{D}(t) = w_n^{\text{rep}}\mathbf{a}_n$. Our representational loss function is then:

$$\mathcal{L}^{\text{rep}}(t_{\text{T}}) = \sum_{s=0}^{T} \left[\mathbf{z}(t_s) - \mathbf{z}^{\text{tar}}(t_s)\right]^2 + \lambda(t)\left[\hat{D}(t_s) - D(t_s)\right]^2, \tag{6}$$

where $\lambda(t)$ weights the representational loss. $\lambda(t)$ is 0 during delay and decision periods; representational constraints are only applied during the cue period, during which $\lambda(t_{\text{cue}}) = 0.01$.

For both loss functions, we use autograd in PyTorch (Paszke et al., 2017). Gradients are calculated and accumulated at every time step in parallel trials presented in batch sizes of 32. Weights are updated after each trial using Adam with $\beta_0 = 0.9$ and $\beta_1 = 0.999$ (Kingma & Ba, 2014). Learning rates are 0.01 for readout or output weights, W^{out} and representational weights W^{rep}, and 0.0003 for recurrent weights, W^{rec}.

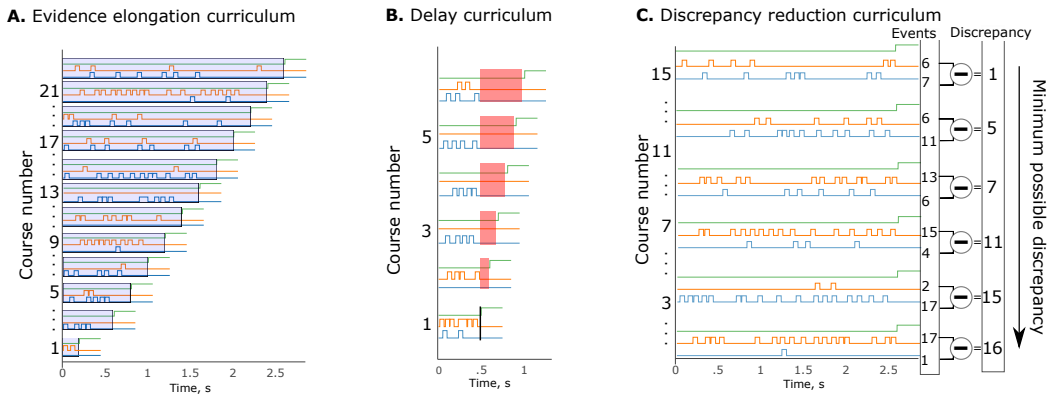

Figure 2: **Curriculum Design.** Each curriculum is presented in "courses", blocks of trials with the same task parameters. The number of trials in a given course is determined by the performance of the network. **A** *Evidence elongation curriculum.* Cue period is elongated at 0.1s per course for both DD and EA tasks (EA schematized here). **B** *Delay elongation curriculum.* Delay period is elongated by 0.1s per course in DD tasks. **C** *Discrepancy reduction curriculum.* The minimum allowable discrepancy is decreased by 1 for each new course, until all discrepancies are presented.

2.5 Curricula

We define curricula using a hierarchical categorization: A curriculum is an ordered set of courses; a course is an ordered set of batches; a batch is a set of trials. A trial is an instantiation of the input space, here, either a DD or an EA task. Parametrization of the input space as task parameters lets us conceptualize each course as a subset of the input space as well. For example, if we consider all possible trials of the DD task, one subset–or course–could consist of all trials with a 1s delay. Our goal is not to learn the entire input space, but rather a subset, which we refer to as "desired task" When there is only one course (i.e., when the desired task is trained directly), we call this the *null curriculum*.

We extend the above definition of a curriculum to also include administration rules for different courses. These rules determine the number of batches in a course presented to the RNN during the curriculum as well as the transition between courses. For example, each course in a curriculum could be administered for 50 batches before transitioning to the next course in the ordered set. However, here we also include a test set for each course that allows us to evaluate performance after each batch. Our test set is designed to have balanced discrepancy, with 10 examples of each allowable discrepancy. The final course of all curricula for a given task (EA and DD) share the same test set. If the RNN correctly solves 75% of tasks in the test set, it "graduates" to the next course. An RNN output is correct if the integral over the decision period is within 50% of the integral of the target function.

All curricula are inspired by real world shaping procedures in experimental neuroscience, like those seen in Pinto et al. (2018); Duan et al. (2015); Stine et al. (2020); Romo et al. (1999); Constantinidis et al. (2018). We use the following three curriculum types:

Delay elongation curricula: The first course has no delay period; delay elongates by 0.1s per course until it reaches the length of the desired task's delay. Only applies to DD tasks in which the final delay is $.5$s.

Evidence elongation curricula: The first course has a cue period of 0.1s, which elongates by 0.1s per course until it reaches the length of the desired task's cue period. In EA tasks, this is 2.4 seconds; in DD tasks this is .5 seconds.

Discrepancy reduction curricula: The first course has a discrepancy threshold of 15 in EA tasks and 5 in DD tasks, which decreases by 1 with each course.

2.6 Curriculum completion time

We calculate curriculum completion time (CCT) by counting the number of batches a network sees during all the courses in a curriculum. This number also equals the number of weight updates during a curriculum. Given the difficulty of our tasks, not all networks learn to solve the task in a reasonable time, and sometimes not at all, especially under a null curriculum (Figure 3). For comparing CCTs between such "partially undefined" sets, we use a rank-ordered Mann-Whitney U test (Hettmansperger & McKean, 2010). We set a weight update limit of 500 iterations. RNNs that fail to learn can still included because it is known that their CCT is greater than the CCT of all that do learn within the weight update limit.

2.7 State space analyses

In neuroscience, dimensionality reduction techniques, e.g., Principal Component Analysis (PCA) are commonly used to infer dominant features of neural dynamics and to evaluate complexity.(Cunningham & Byron, 2014) We similarly characterize and compare the evolution of state spaces of our RNNs under different curricula and loss functions, and measure their effective dimensionality using PCA. We use all RNN activations from the final test set of each curriculum for computing the covariance matrix for PCA. The covariance matrix has trial time-steps $\times$ number of trials in the rows and neurons $N = 350$ in the columns. The resulting eigenvectors or PCs have N components, and points along this PC space represent times during a trial as a "trajectory". As each time in a trial has a discrepancy, per equation 4, we can color PC trajectories accordingly.

We fit the first PC as the hyperbolic tangent of discrepancy. Formally, our eigendecomposition gives the N-dimensional state $S = \{s_i\}_i^N$ as a function of time $s_i(t) = p_{ij}\mathbf{a}_j(t)$, where P_i is the ith PC vector with elements p_{ij}, j iterates neurons, and $\mathbf{a}_j(t)$ is the activation of neuron j at time t. We find parameters (a, b) to maximize the correlation between s_1 and $\hat{s_1} = b + a\tanh(D(t))$.

To measure dimensionality, we follow the convention of measuring the inverse participation ratio of PC eigenvalues (Rajan et al., 2010).

3 Results

3.1 Null curricula

Both learning rules are able to learn the desired EA task without curricula. BPR makes learning faster and more reliable, with a median CCT of 200 and all 50 RNNs sucessfully solving >75% of the test set trials. BPT had a median CCT of 325 and only 32 of 50 RNNs learned within the 500 weight-updates limit. In Figure 3, we show that training and testing losses in the decision period largely overlap, with less noise in test loss. This is representative of the balanced discrepancy in our test set, where as individual training batches are generated with random discrepancy. The decision period loss curve for BPR is steep (BPT's is not), demonstrating the learning benefit from the representational loss.

For the desired DD task in null curricula, both learning rules are unable to learn the task within 500 weight updates. Loss functions during the decision period, Figure 3, indicate little to no improvement

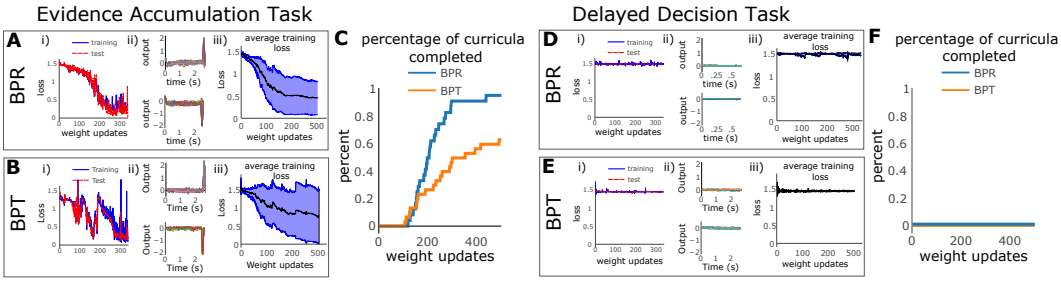

Figure 3: **Learning Without Curricula:** "Null curricula". **A, B:** Performance of BPR and BPT during training with the null curriculum on the EA task. (i) Red loss curves are the mean loss on the test set, where as blue is the mean loss on the training batch. (ii) Sample network output on the test set. Left trials and right trials are separated by upper and lower panels respectively. (iii) Average training loss over 50 RNNs trained. **C:** Cumulative density of curriculum completion times of 50 networks of BPR (blue) and BPT (orange) with the null curriculum. **D, E, F:** Performance of BPR and BPT during training with the null curriculum on the DD task.

to performance over training. Errors from the decision period cannot propagate through the long delay to the relevant directional cues.

3.2 Curriculum Learning

Curriculum learning increases learning speed and reliability in the EA task for BPT (Figure 4) []. However, there is no obvious benefit from curricula for BPR in the same task (Figure 4) []. For the DD task, only the delay elongation curriculum successfully rescues BPT. In BPR, learning is significantly faster during both discrepancy reduction and delay elongation curricula (Figure 4). Evidence elongation curricula are unable to rescue performance in RNNs trained by BPR, failing to learn in 500 weight update limit.

3.3 Learned state spaces invariant to task variants, curricula, and learning rule

Despite learning with different loss functions and under diverse curricula, the state spaces of all RNNs that successfully learn the desired task have the same key features. In particular, all are low dimensional, with two PCs explaining $\geq 90\%$ of the variance. Effective dimensionality is 1.5 in all cases (Figure: 5). Further, the first PC of all state spaces is strongly correlated with, and best fit by, the hyperbolic tangent of the discrepancy, $\mathbf{tanh}(\text{discrepancy})$. The correlation between the data and our fit is >0.9R in all solutions. We argue that given the similarity of the spaces in the learned state, differentiating between learning principles is non-trivial and will represent a key advance for neuroscience.

3.4 Relative curriculum completion time (CCT) is indicative of learning rule

In the EA task, evidence elongation and discrepancy reduction curricula significantly change the distribution of CCT only for BPT. Evidence elongation curricula significantly speeds up learning with a median CCT of 190 vs 325 weight updates (Z: 2.9 from U Test). Discrepancy reduction curricula make learning even faster with a median CCT of 134 (Z: 3.7). For BPR, learning is not significantly improved by any of the curricula tested.

For the DD task, we observe that delay elongation curricula benefit both BPT and BPR, but only BPR benefits from discrepancy reduction curricula. For discrepancy reduction curricula, BPR has a median CCT of 226, which significantly faster than under null curricula that fail to converge in our 500 weight-updates limit (Z: 3.9). Delay elongation curricula were even faster with a median CCT of 150 (Z: 5.4) with BPR. Similarly, for BPT, the median CCT was 154 with significance of Z: 4.16.

Our results in Figure 5, suggest that using the CCT distribution provides enough information to differentiate our target-based (i.e., BPT) versus representational learning (i.e., BPR) models. It is

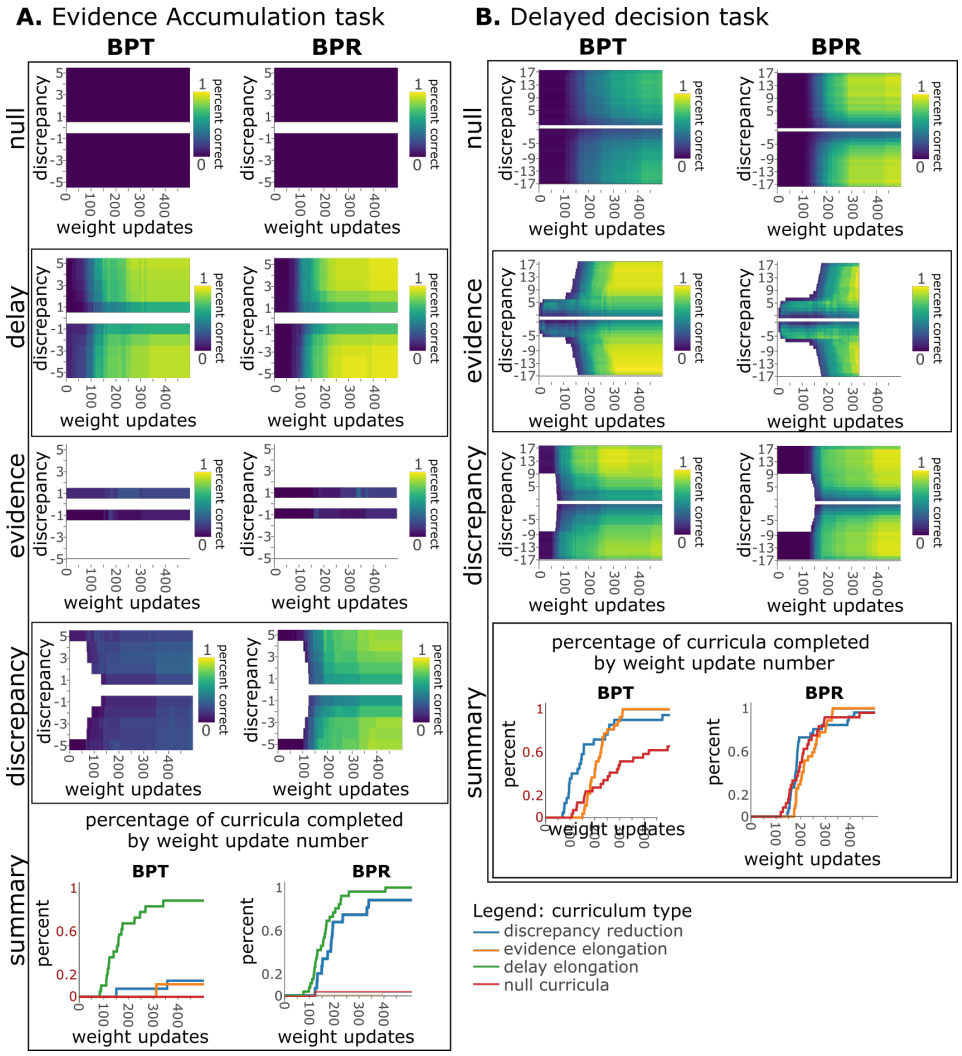

Figure 4: Curriculum Learning Performance. **A:** Evidence Accumulation (EA) task performance of BPR and BPT under all curricula. Each row shows a different curriculum and columns show BPT vs BPR learning rules. **B:** Delayed Decision (DD) tasks performance of BPR and BPT under all curricula. All heatmaps show average performance as a function of discrepancy and weight update. Each heatmap pixel corresponds to the average performance 50 networks on a particular discrepancy. Pixels are excluded if there are fewer than three RNNs tested on that discrepancy. As RNNs may be in different courses at any given time, if a discrepancy is not yet in a particular RNN's test set, that RNN's performance is assumed to be zero on that discrepancy. Bottom graphs show the cumulative density of curriculum completion times under all curricula. Each use 50 RNNs and show what percentages have completed a curriculum by a particular weight update.

non-trivial to classify learning models given the learned state spaces under all twelve conditions, but here we posit that the CCT distribution under just one learning rule or the other is sufficient to reliably determine which learning principle was operational. This is important because our goal is to identify learning principles from neuroscience experiments where individual subjects or lab animals like mice are often assumed to be the same.

3.5 Discussion

In this paper, we proposed a novel use for curriculum learning as a method for uncovering the learning principles of a system. We demonstrated using models of common neuroscience tasks and

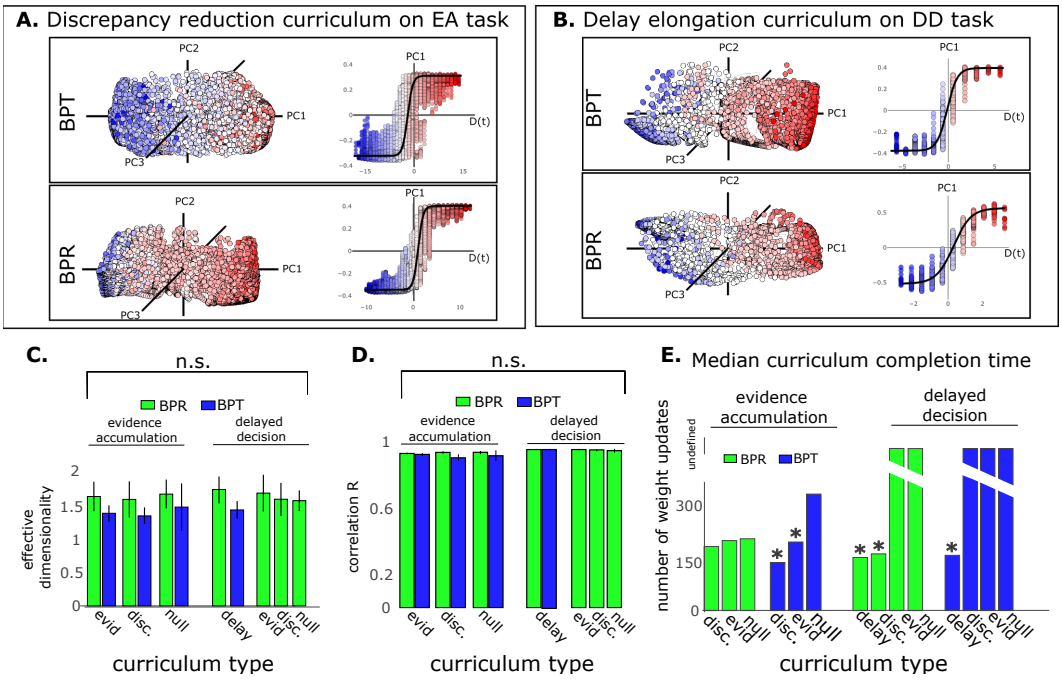

Figure 5: Discriminating rules by learned state versus curriculum completion time. **A:** (left) Low dimensional state spaces of networks trained by BPT and BPR with a discrepancy curriculum on EA task. Principal component scores of each time during the test set are shown on the first three PCs. Each point is colored by the discrepancy $D(t)$, with blue being the most negative discrepancy and red being the most positive. (right) Fit between $\tanh(D(t))$ and PC1. **B:** Low dimensional state spaces on delay elongation curriculum on DD task. **C:** Effective dimensionality of BPR (green) and BPT (blue) under each curriculum (each bar) for EA and DD tasks. **D:** Correlation R between the first principal component and $\tanh(D(t))$ for BPR (green) and BPT (blue) under each curriculum (each bar) for EA and DD tasks.

curricula inspired by standard shaping protocols that information about learning principles may not be accessible by studying fully trained systems. However, we suggest that using only behavioral dynamics during a set of curricula, we can uncover this information. This motivates the collection and curation of behavioral data (and eventually, concomitant neural data) during broad range of shaping procedures already being used by experimental neuroscientists on mice and other lab animals.

Our approach focuses on an easily accessible metric of performance–the time to complete a curriculum. It is likely that other behavioral metrics such as a moving average of performance during curricula could also be valuable or provide better resolution in uncovering learning principles, particularly when combined with neural findings. While we demonstrated that global features of learned neural activations were largely invariant to our selected tasks, curricula, and learning rules (Figure 5, tracking the evolution of activations during the execution of different curricula is likely to be a rich source of information about learning principles. In Nayebi et al. (2020), the authors successfully use information from activations during training without curricula. The addition of curricula to their approach could further improve learning rule discrimination. Furthermore, we suggest that relative predictions, such as the ones we make from curriculum completion times in Figure 5, are invariant to shifts in their absolute quantity might that arise from extending a model to a real brain. For example, if the effective dimensionality predicted by our models was significantly higher after curriculum A than after curriculum B, then such a relative prediction would still hold even if the absolute dimensionalities in the mouse were very different from the model.

We hope that this paper will encourage other AI/ML and computational neuroscience researchers to expand our approach to more tasks and more curricula to further discriminate learning principles. We expect more such effort to provide higher-resolution insights into learning principles in the biological brain and motivate better data collection from different behavioral experiments.

REFERENCES

Zoe Ashwood, Nicholas A. Roy, Ji Hyun Bak, and Jonathan W Pillow. Inferring learning rules from animal decision-making. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 3442–3453. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/234b941e88b755b7a72a1c1dd5022f30-Paper.pdf.

László Babai. Graph isomorphism in quasipolynomial time. *CoRR*, abs/1512.03547, 2015. URL http://arxiv.org/abs/1512.03547.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 41–48, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605585161. doi: 10.1145/1553374.1553380. URL https://doi.org/10.1145/1553374.1553380.

Maneesh Bhand, Ritvik Mudur, Bipin Suresh, Andrew Saxe, and Andrew Ng. Unsupervised learning models of primary cortical receptive fields and receptive field plasticity. *Advances in neural information processing systems*, 24:1971–1979, 2011.

Christos Constantinidis, Shintaro Funahashi, Daeyeol Lee, John D Murray, Xue-Lian Qi, Min Wang, and Amy FT Arnsten. Persistent spiking activity underlies working memory. *Journal of neuroscience*, 38(32):7020–7028, 2018.

John P Cunningham and M Yu Byron. Dimensionality reduction for large-scale neural recordings. *Nature neuroscience*, 17(11):1500–1509, 2014.

Sven Dorkenwald, Claire McKellar, Thomas Macrina, Nico Kemnitz, Kisuk Lee, Ran Lu, Jingpeng Wu, Sergiy Popovych, Eric Mitchell, Barak Nehoran, Zhen Jia, J. Alexander Bae, Shang Mu, Dodam Ih, Manuel Castro, Oluwaseun Ogedengbe, Akhilesh Halageri, Zoe Ashwood, Jonathan Zung, Derrick Brittain, Forrest Collman, Casey Schneider-Mizell, Chris Jordan, William Silversmith, Christa Baker, David Deutsch, Lucas Encarnacion-Rivera, Sandeep Kumar, Austin Burke, Jay Gager, James Hebditch, Selden Koolman, Merlin Moore, Sarah Morejohn, Ben Silverman, Kyle Willie, Ryan Willie, Szi-chieh Yu, Mala Murthy, and H. Sebastian Seung. Flywire: Online community for whole-brain connectomics. *bioRxiv*, 2020. doi: 10.1101/2020.08.30.274225. URL https://www.biorxiv.org/content/early/2020/08/30/2020.08.30.274225.

Chunyu A. Duan, Jeffrey C. Erlich, and Carlos D. Brody. Requirement of prefrontal and midbrain regions for rapid executive control of behavior in the rat. *Neuron*, 86(6):1491–1503, 2015. ISSN 0896-6273. doi: https://doi.org/10.1016/j.neuron.2015.05.042. URL https://www.sciencedirect.com/science/article/pii/S089662731500481X.

Daniel B. Ehrlich, Jasmine T. Stone, David Brandfonbrener, Alexander Atanasov, and John D. Murray. Psychrnn: An accessible and flexible python package for training recurrent neural network models on cognitive tasks. *eNeuro*, 8(1), 2021. doi: 10.1523/ENEURO.0427-20.2020. URL https://www.eneuro.org/content/8/1/ENEURO.0427-20.2020.

Jeffrey L. Elman. Learning and development in neural networks: the importance of starting small. *Cognition*, 48(1):71–99, 1993. ISSN 0010-0277. doi: https://doi.org/10.1016/0010-0277(93)90058-4. URL https://www.sciencedirect.com/science/article/pii/0010027793900584.

Alex Graves, Marc G Bellemare, Jacob Menick, Remi Munos, and Koray Kavukcuoglu. Automated curriculum learning for neural networks. In *international conference on machine learning*, pp. 1311–1320. PMLR, 2017.

Zengcai V Guo, S Andrew Hires, Nuo Li, Daniel H O'Connor, Takaki Komiyama, Eran Ophir, Daniel Huber, Claudia Bonardi, Karin Morandell, Diego Gutnisky, et al. Procedures for behavioral experiments in head-fixed mice. *PloS one*, 9(2):e88678, 2014.

Demis Hassabis, Dharshan Kumaran, Christopher Summerfield, and Matthew Botvinick. Neuroscience-inspired artificial intelligence. *Neuron*, 95(2):245–258, 2017.

Thomas P Hettmansperger and Joseph W McKean. *Robust nonparametric statistical methods*. CRC Press, 2010.

Guosong Hong and Charles M Lieber. Novel electrode technologies for neural recordings. *Nature Reviews Neuroscience*, 20(6):330–345, 2019.

Stephen James, Paul Wohlhart, Mrinal Kalakrishnan, Dmitry Kalashnikov, Alex Irpan, Julian Ibarz, Sergey Levine, Raia Hadsell, and Konstantinos Bousmalis. Sim-to-real via sim-to-sim: Data-efficient robotic grasping via randomized-to-canonical adaptation networks. In *CVPR*, 2019.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Sue Ann Koay, Adam S. Charles, Stephan Y. Thiberge, Carlos D. Brody, and David W. Tank. Sequential and efficient neural-population coding of complex task information. *bioRxiv*, 2021. doi: 10.1101/801654. URL https://www.biorxiv.org/content/early/2021/01/18/801654.

Ding Liu, Xiaowei Gu, Jia Zhu, Xiaoxing Zhang, Zhe Han, Wenjun Yan, Qi Cheng, Jiang Hao, Hongmei Fan, Ruiqing Hou, et al. Medial prefrontal activity during delay period contributes to learning of a working memory task. *Science*, 346(6208):458–463, 2014.

Tom Macpherson, Anne Churchland, Terry Sejnowski, James DiCarlo, Yukiyasu Kamitani, Hidehiko Takahashi, and Takatoshi Hikida. Natural and artificial intelligence: A brief introduction to the interplay between ai and neuroscience research. *Neural Networks*, 2021. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2021.09.018. URL https://www.sciencedirect.com/science/article/pii/S0893608021003683.

Ildefons Magrans de Abril, Junichiro Yoshimoto, and Kenji Doya. Connectivity inference from neural recording data: Challenges, mathematical bases and research directions. *Neural Networks*, 102: 120–137, 2018. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2018.02.016. URL https://www.sciencedirect.com/science/article/pii/S0893608018300704.

Niru Maheswaranathan, Alex H. Williams, Matthew D. Golub, Surya Ganguli, and David Sussillo. Universality and individuality in neural dynamics across large populations of recurrent networks. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 15603–15615, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/5f5d472067f77b5c88f69f1bcfda1e08-Abstract.html.

Aran Nayebi, Sanjana Srivastava, Surya Ganguli, and Daniel LK Yamins. Identifying learning rules from neural network observables. *arXiv preprint arXiv:2010.11765*, 2020.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. *NIPS-W*, 2017.

Lucas Pinto, Sue A. Koay, Ben Engelhard, Alice M. Yoon, Ben Deverett, Stephan Y. Thiberge, Ilana B. Witten, David W. Tank, and Carlos D. Brody. An accumulation-of-evidence task using visual pulses for mice navigating in virtual reality. *Frontiers in Behavioral Neuroscience*, 12:36, 2018. ISSN 1662-5153. doi: 10.3389/fnbeh.2018.00036. URL https://www.frontiersin.org/article/10.3389/fnbeh.2018.00036.

Kanaka Rajan, L Abbott, and Haim Sompolinsky. Inferring stimulus selectivity from the spatial structure of neural network dynamics. In *Advances in Neural Information Processing Systems*, pp. 1975–1983, 2010.

Ranulfo Romo, Carlos D Brody, Adrián Hernández, and Luis Lemus. Neuronal correlates of parametric working memory in the prefrontal cortex. *Nature*, 399(6735):470–473, 1999.

Luca Saglietti, Stefano Sarao Mannelli, and Andrew Saxe. An analytical theory of curriculum learning in teacher-student networks. *arXiv preprint arXiv:2106.08068*, 2021.

T.D. Sanger. Neural network learning control of robot manipulators using gradually increasing task difficulty. *IEEE Transactions on Robotics and Automation*, 10(3):323–333, 1994. doi: 10.1109/70.294207.

Andrew Saxe, Stephanie Nelli, and Christopher Summerfield. If deep learning is the answer, what is the question? *Nature Reviews Neuroscience*, 22(1):55–67, 2021.

J. Schmidhuber. Curious model-building control systems. In *[Proceedings] 1991 IEEE International Joint Conference on Neural Networks*, pp. 1458–1463 vol.2, 1991. doi: 10.1109/IJCNN.1991.170605.

Oliver G. Selfridge, Richard S. Sutton, and Andrew G. Barto. Training and tracking in robotics. In *Proceedings of the 9th International Joint Conference on Artificial Intelligence - Volume 1*, IJCAI'85, pp. 670–672, San Francisco, CA, USA, 1985. Morgan Kaufmann Publishers Inc. ISBN 0934613028.

Wolf Singer. Recurrent dynamics in the cerebral cortex: Integration of sensory evidence with stored knowledge. *Proceedings of the National Academy of Sciences*, 118(33), 2021. ISSN 0027-8424. doi: 10.1073/pnas.2101043118. URL https://www.pnas.org/content/118/33/e2101043118.

Ian H Stevenson and Konrad P Kording. How advances in neural recording affect data analysis. *Nature neuroscience*, 14(2):139–142, 2011.

Gabriel M Stine, Ariel Zylberberg, Jochen Ditterich, and Michael N Shadlen. Differentiating between integration and non-integration strategies in perceptual decision making. *eLife*, 9:e55365, apr 2020. ISSN 2050-084X. doi: 10.7554/eLife.55365. URL https://doi.org/10.7554/eLife.55365.

Matthew E Taylor and Peter Stone. Transfer learning for reinforcement learning domains: A survey. *Journal of Machine Learning Research*, 10(7), 2009.

Xin Wang, Yudong Chen, and Wenwu Zhu. A survey on curriculum learning, 2021.

Daphna Weinshall, Gad Cohen, and Dan Amir. Curriculum learning by transfer learning: Theory and experiments with deep networks, 2018.

Daniel L. K. Yamins, Ha Hong, Charles F. Cadieu, Ethan A. Solomon, Darren Seibert, and James J. DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014. ISSN 0027-8424. doi: 10.1073/pnas.1403112111. URL https://www.pnas.org/content/111/23/8619.

Guangyu Robert Yang, Michael W Cole, and Kanaka Rajan. How to study the neural mechanisms of multiple tasks. *Current opinion in behavioral sciences*, 29:134–143, 2019.

Anthony M. Zador, Joshua Dubnau, Hassana K. Oyibo, Huiqing Zhan, Gang Cao, and Ian D. Peikon. Sequencing the connectome. *PLOS Biology*, 10(10):1–7, 10 2012. doi: 10.1371/journal.pbio.1001411. URL https://doi.org/10.1371/journal.pbio.1001411.