A Sequence Learning Model for Decision Making in the Brain

Zhewei Zhang[1,2], Huzi Cheng[1], Zhongqiao Lin[1,2], and Tianming Yang[1*]

[1] Institute of Neuroscience, Key Laboratory of Primate Neurobiology, CAS Center for Excellence in Brain Science and Intelligence Technology, Shanghai Institutes for Biological Sciences, Chinese Academy of Sciences

[2] University of Chinese Academy of Sciences, Beijing 100049, China

*Address correspondence to:

Tianming Yang (tyang@ion.ac.cn)

Institute of Neuroscience

320 Yue Yang Rd.

System Neuroscience Building, Room 302

Shanghai, China 200031

Phone: +86-21-54921737, Fax: +86-21-54921735

Abstract

Decision making is often modelled as a competition between options. Currently, a great number of popular models to explain the accuracy and speed in decision making are based on variations of drift diffusion models (DDM), in which the options compete by accumulating evidence toward decision bounds. Attractor-based recurrent neural networks have been proposed to explain the underlying neural mechanism. Yet, it is questionable that either the DDM or attractor network is the brain's general solution for decision making. Here, we propose an alternative recurrent neural network modeling approach based on gated recurrent units and sequence learning. Our network model is trained to learn the statistical structure of temporal sequences of sensory events, action events, and reward events. We demonstrate its learning with a reaction-time version of the weather prediction task previously studied in monkey experiments, in which both the animals' behavior and the neuronal responses were consistent with the DDM. The network model's performance is able to reflect the accuracy and reaction time pattern of the animals' choice behavior. The analyses of the unit responses in the network reveal that they match important experimental findings. Notably, we find units encoding the accumulated evidence and the urgency signal. We further identify two groups of units based on their connection weights to the choice output units. Simulated lesions of each group of units produce doubly-dissociable effects on the network's choice and reaction time behavior. Graph analyses reveal that these two groups of units belong to one highly inter-connected sub-network. Finally, we show that the network is capable of making predictions consistent with the predictive coding and Bayesian inference framework. Our work offers experimentally testable predictions of how decision making is achieved in the brain. It provides an approach that may piece together experimental findings of decision making,

reinforcement learning, and predictive coding. In particular, it suggests that the DDM may be a manifestation of a more general computational mechanism in the brain.

Introduction

Decision making based on noisy or probabilistic evidence is often solved with a sequential sampling strategy, in which noisy evidence is collected and accumulated until a decision criterion is met in favor of one of the decision options. Such decisions involve the consideration of the balance between accuracy and speed. A large body of studies have proposed many models to explain the choice behavior, including the drift diffusion model (Stone, 1960; Ratcliff, 1978), the linear ballistic accumulator model (Brown and Heathcote, 2008), and the leaky competing accumulator model (Usher and McClelland, 2001), which differed on the details of how evidence is accumulated. Experimental works have revealed neurons in many different brain areas, including the prefrontal cortex, the parietal cortex, and the basal ganglia, behave as accumulators during decision making (Kim and Shadlen, 1999; Gold and Shadlen, 2000; Roitman and Shadlen, 2002; Cisek and Kalaska, 2005; Yang and Shadlen, 2007; Ding and Gold, 2010; Heitz and Schall, 2012; Hanks et al., 2015; Yartsev et al., 2018). Models based on attractor neural networks have been proposed to explain at the circuit level how decision making is carried out in the brain (Wang, 2002; Wong and Wang, 2006). The units in attractor networks show response patterns resembling the experimental findings and the choice behavior of these networks matches the predictions from the DDM.

Despite its success, the potential of the attractor network model as a general model for decision making is marred by its lack of cognitive flexibility, which is a signature of the full decision-making capacity in more advanced animals like primates including humans. If a task uses complicated a stimulus set (Yang and Shadlen, 2007; Cisek et al., 2009; Kira et al., 2015), contains multiple decision steps (Gläscher et al., 2010), or requires integration of past history to update the stimulus-response contingencies (Walton et al., 2010; Jang et al., 2015), attractor network models would require significant modifications if they are applicable at all (Soltani and Wang, 2010). Yet, so far there is a lack of alternative neural network models that may account well for the DDM in the field.

Here, we propose an alternative approach that starts from a simple assumption: the brain needs to make appropriate responses at appropriate time based on the past sequence of events. Therefore, instead of modeling the decision making as a competition between alternatives, we consider decision making as a problem of learning the statistical structure of a task's event sequence. Importantly, the task event sequence includes not only the sensory events, but also the corresponding decisions and hence the motor outputs and the reward events as the decision outcome. We build a recurrent network model to learn the statistical structure of a task's temporal sequence that comprises of sensory inputs, actions, and reward events. The model is based on gated recurrent units (GRU), which has been shown to be suitable for learning sequences (Cho et al., 2014; Chung et al., 2014). The network takes inputs in the form of a temporal sequence of a combination of sensory events, action events, and reward events, and generates predictions of future sensory events, action events, and reward events. A supervised learning algorithm is used to train the network to reproduce the

input sequences. The predicted action events are treated as the model's responses and are assessed for the behavior performance. The predictions of sensory and reward events are not directly exhibited in behavior, but are part of the learning.

We demonstrate the model's capability with a sophisticated probabilistic reasoning task, which we previously studied in monkey and human experiments and contains a complicated statistical structure in its task sequence comprising sensory cues, actions, and reward outcomes (Yang and Shadlen, 2007; Kira et al., 2015). The model replicates the animals' choice accuracy and reaction time behavior. The activities of the units in the network mimic the neural activity previous reported in many structures in the brain. Finally, we analyze the network structure and show that how units in the model contribute to the choice and reaction time behavior and how they are connected to each other. Our results suggest that the DDM may be a manifestation of a more general computational mechanism that the brain employs to solve a broader class of decision-making problems. Although the model is presented here as a solution to the decision-making problem, several features of the model allow it to be a much more general model for the brain. Notably, the essence of our model, which is to generate predictions, coincides with the predictive coding framework.

Results

Network

Our model is a network that contains three layers: the input layer (IL), the hidden layer based on gated recurrent units (GRUL), and the output layer (OL) (Fig 1a). The input layer contains units (N_{IL}=20) that carry the information about the sensory, motor, and reward events. Each unit's activity represents the presence or the absence of the corresponding event on the time line continuously. When an event is present, which is the case when a stimulus is turned on, an action is initiated, or a reward is given, the corresponding unit's activity is set to 1; otherwise it is 0. The input units are fully connected with the next layer. We use vector x_t of length N_{IL} to describe the activities of the IL units.

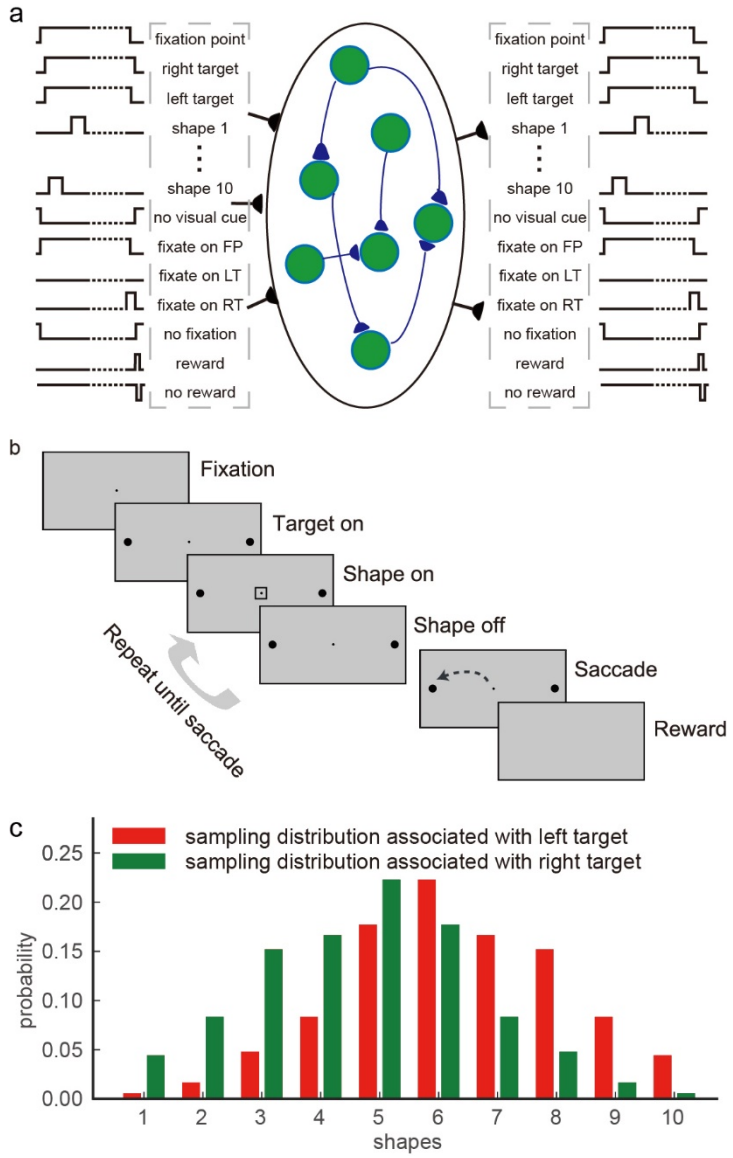


Figure 1. **a**. The model diagram. The network has three layers: the input layer (IL), the gated recurrent unit layer (GRUL), and the output layer (OL). The IL receives the input sequence of sensory events, action events, and reward events. The GRUL has 128 gated recurrent units. The OL has units mirroring the IL units and produces the prediction of the future events. **b**. The reaction-time version of probabilistic reasoning task. The subject fixates at a central point and views a sequence of shapes to make a decision by moving the eye toward one of the two choice targets in the peripheral. **c**. The sampling distributions. Shapes are sampled from the sampling distribution associated with the correct target in each trial.

The hidden layer (GRUL), which is the core of our model, is based on gated recurrent units. There are N_h=128 units in the GRUL. The state vector h_t of length N_h describes the responses of these units at time *t*, which is updated with the information from the IL as follows:

$$z_t = \text{Sigmoid}(w_z x_t + u_z h_{t-1} + b_z)\,, \tag{1}$$

$$r_t = \text{Sigmoid}(w_r x_t + r_r h_{t-1} + b_r)\,, \tag{2}$$

$$h_t = (1 - z_t) \bullet h_{t-1} + z_t \bullet \tanh(w_h x_t + u_h (r_t \bullet h_{t-1}) + b_h)\,, \tag{3}$$

$$h_t = [h_t]_+, \tag{4}$$

where z_t and r_t represent the update gate vector and the reset gate vector, respectively; w_z, w_r and w_h are the input connection weight matrices for the update gates, the reset gates, and the gated units; u_z, u_r and u_h are the recurrent connection weight matrices for the update gates, the reset gates, and the gated units; b_z, b_r and b_h are the bias vectors for the update gates, the reset gates, and the gated units; and • indicates the element-wise multiplication. The rectified-linear function $[\bullet]_+$here keeps the activities non-negative.

The GRUL units project to the output layer with full connections. The OL is composed of an array of units that mirror the IL units ($N_{OL} = N_{IL}$= 20), representing the network's predictions of the corresponding sensory events, action events and reward events for the next time step. The activity of the OL units o_t (a vector of length N_{OL}) is:

$$o_t = \text{Sigmoid}(w_o h_t), \tag{5}$$

where w_o is the output connection weight matrix. The prediction y is a function of response o_t:

$$y_t = \begin{cases} 1 & o_t > 0.5 \\ 0 & o_t < 0.5 \end{cases} \quad (6)$$

The network predicts the corresponding event would happen in the next time step when y_t^i is 1. If y_t^i corresponds to an action, the probability of an action is carried out at the next time step is a softmax function of o_t:

$$P(a_t = a_i \mid o_t^i) = exp(\beta * o_t^i)/\sum_{j=1}^{4} exp(\beta * o_t^j), \quad (7)$$

where temperature parameters β is 8 in the simulation. The chosen action is evaluated for the network's task performance for testing purposes.

Behavior Task

We train our model with a reaction-time version of probabilistic reasoning task described previously (Kira et al., 2015). The task is illustrated in Fig 1b. In this task, a subject has to make decisions between a pair of eye movement targets. Initially, the subject needs to fixate on a central point on a computer screen. Then a stream of shapes appears sequentially near the fixation point. There are totally 10 possible shapes. Each shape conveys information on the correct answer. The subject needs to integrate information of the shapes to form a decision and move the eyes to look at the choice target whenever ready.

In each trial, the computer randomly decides the correct target. Each target is associated with a distribution of the shapes in the sequence (Fig 1c). Each shape in the sequence is independently sampled from the distribution associated with the correct target. Because the likelihoods of observing a particular shape under the two distributions are different, each shape provides a bit of information on which target is more likely to be correct. It has been shown that the sequential probability ratio test (SPRT) is an optimal strategy to solve the task in the sense that it requires the least number of observations to achieve a desired performance (Wald and Wolfowitz, 1948). In the SPRT, one needs to accumulate the log likelihood ratio (logLR) associated with each piece of evidence, which is the log ratio between the conditional probabilities of observing the evidence given the two testing hypotheses. In our task, the logLR associated with each shape ranges from -0.9 to 0.9 (base 10). We define positive and negative logLRs as evidence supporting the left and the right target, respectively.

The task features a sophisticate statistical structure containing the shapes, the choice, and the reward. In addition, choices have to be made at appropriate time. None of the task knowledge is explicitly modeled in our network. By simply training the network with task sequences, we are going to show how our model learns to perform the task and generates behavior consistent with the DDM. Furthermore, we will show how neurons in the network encode information related to the task and they resemble previous experimental findings.

Network training

The training dataset is created with simulated trials in which choices are generated with a drift-diffusion model with collapsing bounds. In the beginning of each trial, the correct target is randomly selected, and the shapes are generated with the corresponding distribution (Fig 1c). A choice is triggered when the accumulated logLR reaches to either of the two opposite collapsing bounds. The bounds start at ±1.5 and linearly collapse toward 0 at the rate of 0.1 per shape epoch. The left target is chosen if the positive bound is reached, and the right target chosen when the negative bound is reached. If the choice matches the pre-determined outcome, a reward is given, and the trial sequence is added into the training dataset. Therefore, only rewarded trials are fed into the network for training.

The goal of the training is for the network to learn to predict events and generate sequences that lead to a reward. The loss function L is defined as the sum of mean squared error between elements in the output o_t and actual event sequence e_{t+1} for all time points t.

$$L = \sum_{t=0}^{t=T-1} (o_t - e_{t+1})^2 \Big/ N_o \tag{8}$$

where T is the length of each trial. The parameter vector θ (including w_*, u_* and b_*) of the network is updated with the stochastic gradient descent (SGD) and the gradient clipping is applied to avoid exploding gradients as follows:

$$\theta_t = \theta_{t-1} - \eta\, g, \tag{9}$$

$$g = \min\left(clip = 0.25,\ \frac{\partial L}{\partial \theta}\bigg|_{\theta = \theta_{t-1}}\right), \tag{10}$$

where η=0.6 is the learning rate.

The results presented below is based on 20 simulation runs using a training set containing 2×10^5 trial sequences.

Behavior Performance

After training, the network is able to make appropriate choices based on shape sequences. When the total logLR associated with the shape sequence is positive, the network tends to choose the left target, and when the total logLR is negative, it tends to choose the right target (Fig 2a). A logistic regression reveals that the logLR assigned to each shape correlates well with its leverage on the choice (Fig 2b). The average number of shapes that the model uses for decision is 6.94$\pm$ 0.04 (mean $\pm$sem). Both the distribution of the reaction time, which is quantified as the number of the shapes presented before the choice in each trial, and the total logLR at the time of choice suggest that the model behavior is consistent with a DDM with collapsing bounds (Fig 2c). We use another logistic regression to examine the effects of shape order on the choice. We observe that even the very first shapes in the sequence exert leverages on the choice similar to the rest of the shapes, except for the last shape, suggesting all shapes are used in decision making (Fig 2d). The last shape's sign is consistent with the choice in more than 99% of all trials. This is consistent with the DDM, in which the last shape should bring the total logLR over the bound. Overall, the network performance resembles the behavior of the macaque monkeys trained with a very similar task (Kira et al., 2015).

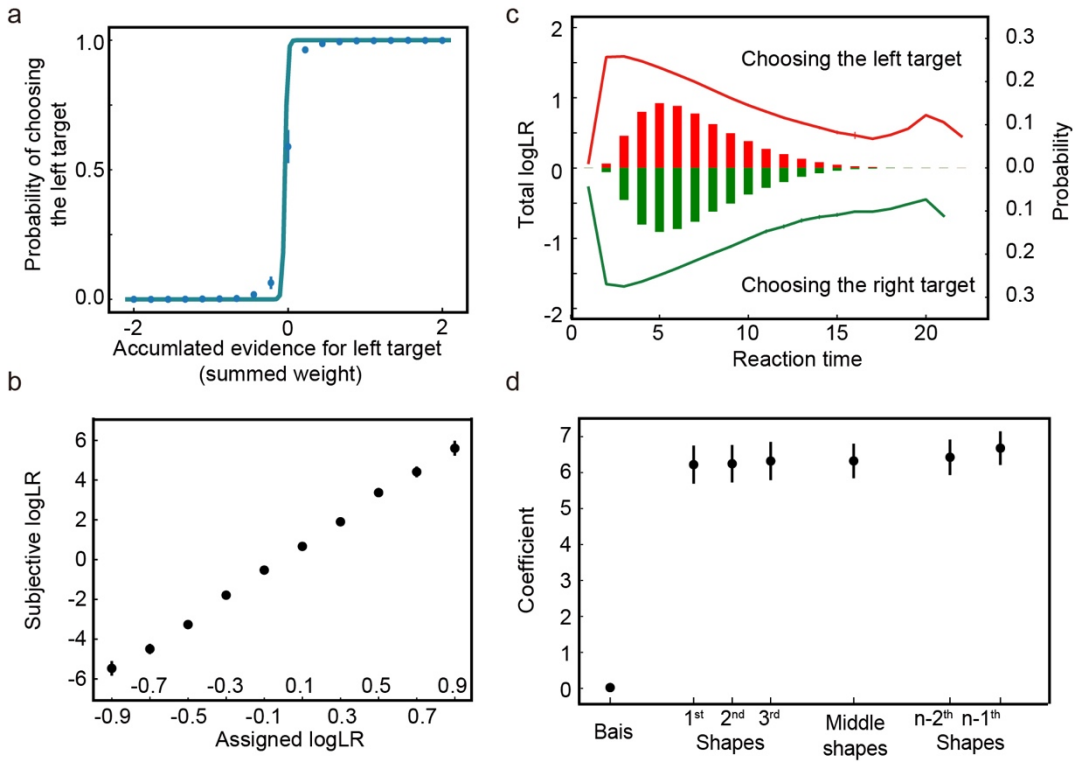


Figure 2. **a**. The psychometric curve. The model chooses the left target more often when the accumulated evidence supports the left target. The blue curve is the fitting curve from the logistic regression. **b.** The leverage of each shape on choice revealed by the logistic regression is consistent with the assigned logLR. **c.** The lines indicate the mean total logLR (left y-axis) for trials grouped by reaction time, quantified with the number of shapes. The bars show the reaction time distribution (right y-axis). Red and green indicate the left and right choices, respectively. **d**. The leverage of the first 3, the second and third from the last, and the middle shapes on the eventual choices. Only trials with more than 6 shapes are used in this plot. No significant differences are found between any two coefficients of shape regressors (two-tailed t test with Bonferroni correction). The error bars in all panels indicate the standard error across runs. Some error bars are smaller than the data points and not visible as the result.

The good performance of the network is not because the network remembers all the training sequences. The network is able to generalize beyond the training sequences. Actually, the network can be trained with a training dataset that contains only 1000 unique sequences and achieves a good performance (Supplementary Figure 1). The learning of the statistical structure of the task is the key to the model's performance.

Network analyses – Evidence and Choice encoding

Next, we examine how the evidence and the choice are encoded in the network to understand how the network performs the task and find out if the units in the network match the experimental findings in the brain.

We use a linear regression to quantify the selectivity of each units (see Methods) of three parameters: the accumulated evidence, quantified with the total logLR, the choice outcome, and the urgency. Here, the urgency is quantified as the number of shapes that have been displayed as a measure of passing time. Many units are found to encode the total logLR (N = 67.01 ± 2.36), the choice outcome (N = 81.45 ± 1.21) and the urgency (N= 82.70 ± 1.52). The example unit in Fig 3a shows a classical ramping-up activity pattern that has been reported in neurons from the prefrontal cortex (Kim and Shadlen, 1999), the parietal cortex (Roitman and Shadlen, 2002; Yang and Shadlen, 2007; Kira et al., 2015), and the striatum (Ding and Gold, 2010). Its activity increases when the total logLR grows to support its preferred choice and decreases when the total logLR is against its preferred choice. In contrast, the example neuron in Fig 3b has a ramping-up activity pattern independent of the choice, which is a signature of urgency signal.

Neurons with the similar activity pattern have been reported in the global pallidus (Thura and Cisek, 2017).

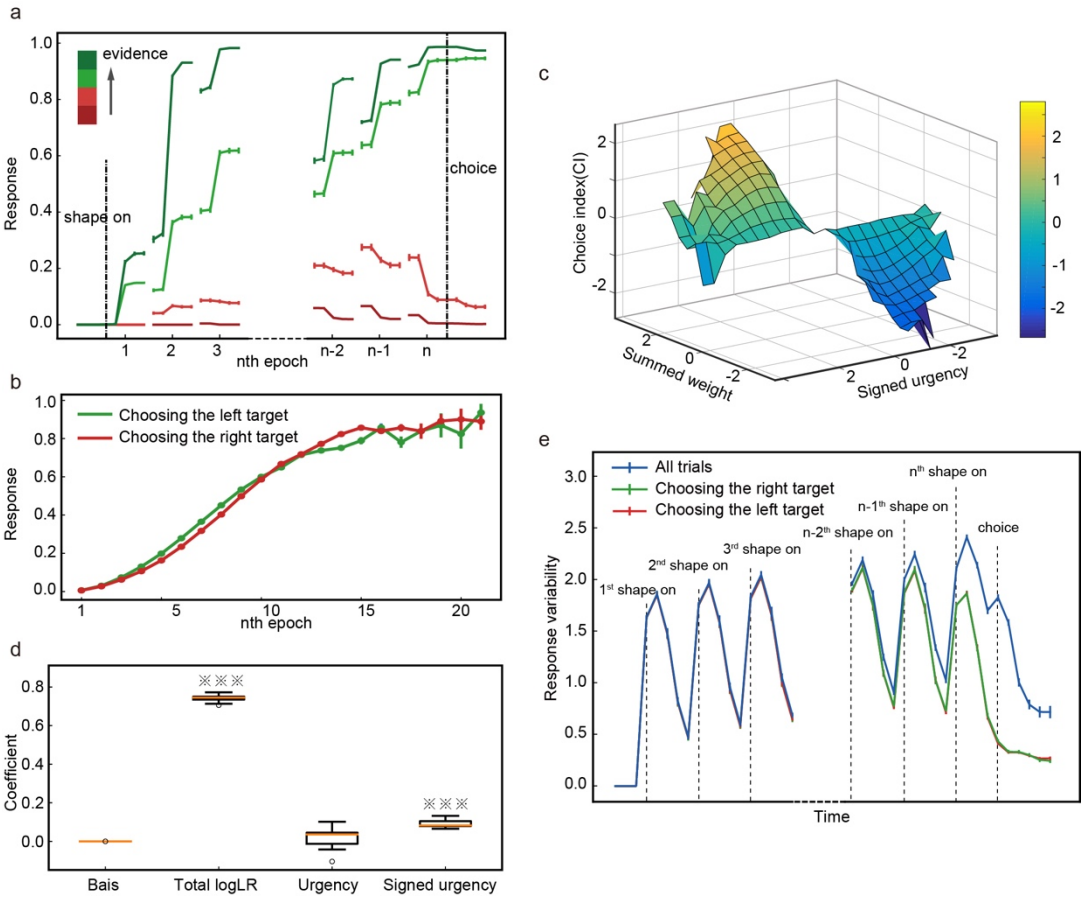


Figure 3. **a**. An example neuron that accumulates evidence and reflects eventual choice. The trials are grouped into quartiles by the accumulated evidence in each epoch. **b**. An example neuron that encodes the urgency signal. The error bars in panels a and b indicate the S.E. across trials. **c.** The normalized choice index (CI, magnitude indicated by color) varies with the total logLR and urgency signal. **d**. The linear regression shows that the normalized CI is affected by the total logLR and the signed urgency signal. ※※※ indicates $p<0.001$(Two-tailed t test with Bonferroni correction). The boxplot shows the median (organ lines), lower and upper quartile values (boxes) and the maximum and minimum of values (horizontal black lines). The hollow dots are outliers. **e.** Network unit response variability. Response variability is aligned to the 1st shape onset (left half of the plot) and to the choice

(right half of the plot) in all trials (blue), the trials in which the left target is chosen (red), and the trials in which the right target is chosen (green). The error bars indicate the standard error across runs.

We further examine how the population responses evolve during the decision-making process. By using the targeted dimension reduction analysis, we identify the dimension which captures the largest variance due to the choice. We project the population responses in each shape presentation epoch onto this choice-related dimension. Thus, the population responses of the hidden layer are mapped onto a one-dimensional choice index (CI). We normalize the CI by subtracting its mean and dividing it by its variance. The sign of the normalized CI indicates the choice based on current evidence and the magnitude of CI indicates the accumulation of the evidence. The normalized CI is correlated with the total logLR and the signed urgency signal, which is the product of the sign of the total logLR and the urgency signal (Fig 3cd). It suggests that the total logLR is the main factor that affects the population response along the choice axis. The urgency signal amplifies the CI without affecting its sign.

Finally, we study the response variability of the units in the GRUL to understand the dynamics of network (Fig 3e). We find that the units' activities diverge in the beginning of a trial and the divergence grows as more shapes are presented. However, when it's close to the time of choice, the population activities become homogeneous across the trials of the same choice, leading to a decrease of response variability. The response variability reaches its minimum around the time of choice. The overall pattern is very similar to that of the intraparietal neurons observed in the brain (Churchland et al., 2011).

Network analyses – When and Which

The choice and the urgency signal have been reported to be combined together at the level of single neuron (Heitz and Schall, 2012; Hanks et al., 2014). Yet, other studies have suggested that there are populations of neurons that contribute to choice and reaction time separately (Thura and Cisek, 2017). To study how units in our model encode these two signals and contribute to the choice behavior, we examine how they connect to the output units.

The choice of the network is reflected by the output units related to the eye movement of the network. They include one unit for the fixation (FP) and two units for the saccades toward the left (LT) and the right (RT) target. For each GRUL unit, we estimate how it contributes to these outputs by defining two indices, I_{when} and I_{which}:

$$I_{when} = w^{i}_{ho-LT} + w^{i}_{ho-RT} \quad (11)$$

$$I_{which} = w^{i}_{ho-LT} - w^{i}_{ho-RT} \quad (12)$$

where w^{i}_{ho-LT}, and w^{i}_{ho-RT} are the connection weights between the GRUL unit *i* and the OL units LT and RT, respectively.

We then use I_{when} and I_{which} to select two groups of GRUL units: the *when* units and the *which* units. Within each group, we further divide them into positive and negative units depending on their signs of I_{when} or I_{which}. The +*when* neurons are selected with the

following procedure. We first sort all units with positive I_{when} values by their I_{when} values from the largest to the smallest. We calculate the accumulative I_{when} along this axis and select the top units that together contribute more than 50% of the sum of the positive I_{when}. These units are defined as the +*when* units (7.70±0.25 out of 42.55±1.02 units with positive I_{when}). They have largest connection weights to the two saccade output units than to the fixation output unit (Fig 4a). We use similar procedures to select the -*when*, the +*which*, and the -*which* units (n=19.10±0.35 out of 85.45±1.02 units with negative I_{when}, 9.00±0.25 out of 62.50±1.36 units with positive I_{which}, and 9.35±0.23 out of 65.50±1.36 units with negative I_{which}, respectively). The -*when* units have smaller connection weights to the saccade units than to the fixation unit, the +*which* units have larger connection weights to the left saccade than to the right saccade output unit, and the -*which* units have smaller connection weights to the left saccade than to the right saccade output unit (Fig 4a). With these selection criteria, the *when* units contribute more to the decision of when a response should be made, while the *which* units contribute more to the decision of which target should be chosen. Together they constitute 35.27±0.29% of all the units in the GRUL.

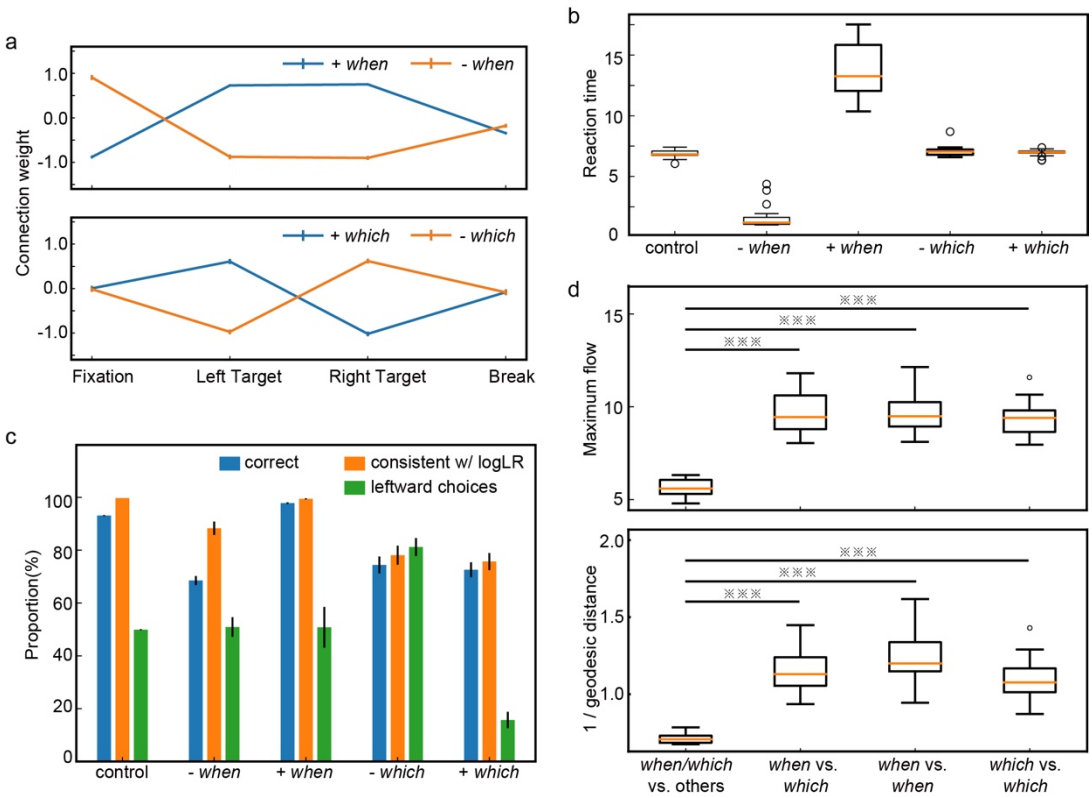


Figure 4. **a**. The connection weights between the eye movement output units and the *when* and *which* units. **b**. The effect of lesions to the *when* and *which* units on the reaction time. **c.** The effect of lesions to the *when* and *which* units on the choice. The blue dots indicate the proportion of correct trials. The orange dots indicate the proportion of trials in which the choice is consistent with the sign of the accumulated evidence at the time of choice. The green dots indicate the percentage of trials choosing the right target. **d**. The maximum flow (upper panel) and the inverse of geodesic distance (lower panel) between different unit groups. The smaller maximum flow and the larger geodesic distance between *when/which* units and other units suggest the relatively tight connections between the *when* and *which* units. ※※※ indicates $p < 0.001$ (Two tailed t-test with Bonferroni correction). The error bars in all panels indicate the standard error across runs.

We further examine the causality between the activities of the *when* and *which* units and the network's performance. The link can be demonstrated by simulated lesions that selectively inactivate the connections of each group of units to the output layer. With only the outputs inactivated, the hidden layer itself is not disturbed. When we inactivate the +*when* units, the RTs become longer while the accuracy remains intact (Fig 4bc). In contrast, when we inactivate the output of the -*when* units, the network's RTs become smaller (Fig 4b). Although the network performance has an apparent drop, the network's choice still accurately reflects the accumulated logLR, suggesting the performance drop is due to the fact that the network has to work with a smaller total logLR because of the shorter RT (Fig 4c). In comparison, when *which* neurons are manipulated, only the choice accuracy is affected but the RT remains the same (Fig 4bc). More specifically, inactivating +*which* units leads to a bias toward the right target, which is associated with the negative logLR in the task design, while inactivating -*which* units leads to a bias toward the left target (Fig 4c). These results suggest there are two distinct populations of GRUL units contributing to the choice and the reaction time.

The distinct functional roles of *when* and *which* neurons may suggest they also have different connection patterns. We use the connection matrix of the GRUL units to construct a directed weighted graph and calculate the geodesic distance and the maximum flow between the GRUL units (see Methods). To account for units who are not connected to each other, we use the average of the inverse of geodesic distance in the analysis. Interestingly, we find the inverse of geodesic distances between the *when* neurons and the *which* neurons are significantly larger than the average inverse of distance between the *when*/*which* units and others. The maximum flows between them are significantly higher than the network average (Fig 4d). These results suggest that

the *when* and the *which* neurons belong to a tightly connected network and are not topologically separated.

Network analyses – Predictive Coding

So far, we have shown that our network is able to generate appropriate choices based on shape sequences. However, the network is constructed and trained in the way that the sensory events are treated exactly the same as the action events. Therefore, the activities of the output units that correspond to the sensory events should provide predictions the sensory events. In other words, our network may also serve as a model for the predictive coding framework.

In the probabilistic reasoning task, each choice is associated with a shape distribution. Therefore, the past shape sequence is informative of the probability distribution of the upcoming shapes. To examine how well the network predicts the shapes, we plot the subthreshold activity, o_t, of the ten shape output units in Figure 5 at the time point right before the onset of the next shape. When the current evidence is in favor of the left target, the activities of the OL shape units resemble the distribution from which the shapes are drawn when the left target is the correct target. When the evidence is in favor of the right target, the activities of the OL shape units shift to resemble the distribution from which the shapes are drawn when the right target is correct (Fig 5a).

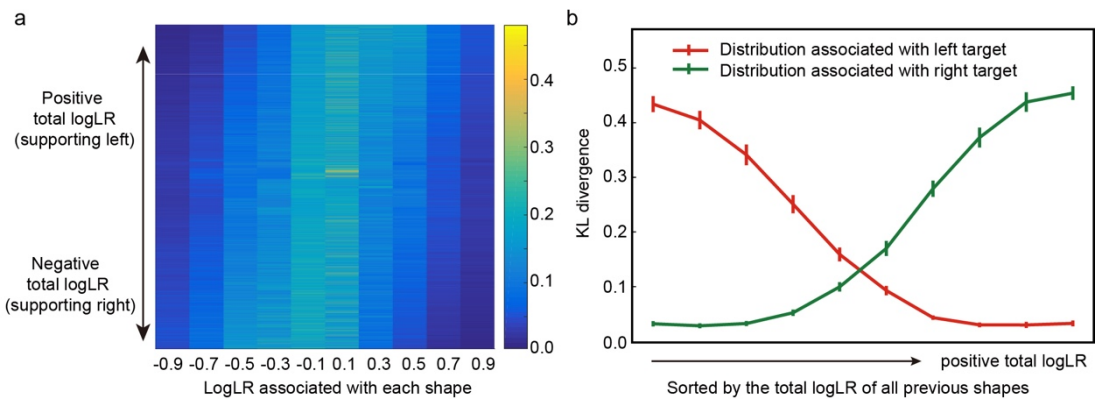


Figure 5. **a**. The subthreshold activities of 10 shape output units, plotted in 10 columns with their assigned logLR indicated below. We calculate the 10 shape units' activities at the time step immediately before each shape onset for all epochs in all trials in the test dataset (96270 trials and 668162 epochs from 20 runs), sort them by the total logLR before the shape onset, and plot them in rows of the heatmap. **b**. The Kullback-Leibler (KL) divergence between the subthreshold activities and the sampling distributions showing in the Fig.1c. Data is grouped by the total logLR. The error bars indicate the standard error across runs.

We quantify the similarity between the activity patterns of the OL shape units and the shape distributions by calculating the Kullback-Leibler (KL) divergence (Fig 5b). The KL divergence between the OL unit activities and the shape distribution associated with the left target decreases as the total logLR supporting the left target gets larger and grows when the total logLR is smaller. The opposite trend is observed on the KL divergence between the OL unit activities and the shape distribution associated with the right target. These results suggest the activities of the OL shape units reflect the probability distribution of the next shape based on the current evidence.

460 **Discussion**

461

462 Here, we present a neural network model based on gated recurrent units. The network
463 is trained to learn the statistical structure of a probabilistic reasoning task. We show
464 that the network choice and reaction time behavior resemble those of animals and are
465 consistent with the DDM. The neurons in the network exhibit signature behavior of the
466 evidence-accumulating neurons previous described experimentally. Two groups of
467 neurons, the *which* and the *when* neurons, may be identified by their output weights.
468 Their contributions to the choice and the reaction time may be doubly-dissociated with
469 simulated lesions. Finally, the network may also serve as a neural network model for
470 predictive coding framework.

471

472 Relationship to the previous modeling work

473

474 Our modeling effort is based on a distinct approach from the previous work of attractor
475 models (Wang, 2002; Wong and Wang, 2006). Instead of modeling decision making as a
476 competition between different options, the current model considers it as a statistical
477 learning problem. Although our network does show a smaller response variability at the
478 time of choice (Fig 3e), which is a signature of attractor network, our emphasis is not on
479 how the activity of the network converges to an attractor-state to form a decision. The
480 decision making is an exemplar behavior of the network's overall ability to predict
481 events.

482

483 Compared to other sequence learning models (Lotter et al., 2018), one unique feature of
484 our model is that we include not only the sensory events, but also the motor events and

the reward events into the sequences that the model has to learn. Not only the choice, but also the timing the choice can be generated by learning the statistical structure of the sequences. This gives the model an advantage over the standard reinforcement learning algorithms.

As the result, our model is a flexible framework for behavior including decision making. The particular behavior paradigm we use to illustrate our points contains sequences that may have more than 10 events, which constitute a sequence set containing astronomical number of combinations. Considering that most of the behavior tasks studied in the modeling literature are much simpler (Yang et al., 2019), our model should be applicable to them. In addition, the model can be used for tasks that include multiple action stages, such as the two-step task (Gläscher et al., 2010). Finally, the sequences of events used to train the model do not have to be individual trials. By feeding multiple trials as a single sequence into the model, we can train the model to learn the contingencies across trials so that the model may perform tasks that depends on trial history information, such as many variations of reversal learning (Walton et al., 2010; Jang et al., 2015).

Training

To train the model, it is necessary to feed the model with sequences of sensory, action, and reward events. This means there has to be a separate mechanism to generate appropriate responses and form these sequences in the first place for the brain to learn. One potential solution is based on the fact that the animals may possess innate or other

established stimulus-action associations relevant to the task. These responses should contain a certain degree of variations that lead to different reward outcomes (Neuringer, 2002). For example, when we train monkeys to perform complicated behavior task such as the probabilistic reasoning task, we usually start from the most basic delayed saccade task. Monkeys show an innate tendency of making saccades toward unexpected visual stimuli with variable delays. By selectively rewarding saccades with longer delays, we can train the monkeys to hold the fixation for an extended period before making saccades. More components of the task then can be introduced into the task gradually. We may train our network with a similar strategy. Starting with simple tasks that the network is capable of doing, we can selectively reward the network and use its output sequence for next stage training, until we reach to the final version of the task.

Reinforcement learning

By learning statistical relationships between sensory, action, and reward events, our model can be trained to perform many of the behavior paradigms that involve the learning of the contingencies between stimuli, actions, and rewards, which have been mostly modelled in the field with the reinforcement learning framework.

The biggest conceptual difference between our model and the reinforcement learning model is the role of reward. In reinforcement learning, the system is trained to maximize the reward. The discrepancy between the actual and the expected rewards is used to update the value of states or the policies that determine actions based on the

states. In contrast, our model treats the rewards as part of an event sequence just as other sensory or motor events. The learning in our model is driven by the prediction errors calculated with sensory, action, and reward events. There are no explicit states or policies that have to be evaluated during learning, and choices are not made based on the value of these states.

The findings about dopamine neurons have been used to support the existence of a reinforcement learning system in the brain (Schultz et al., 1997). Yet, recent findings have started to reveal a wider role of dopamine neurons in signaling not just reward prediction error, but also predictions of sensory stimuli or actions (Jin and Costa, 2010; Wassum et al., 2012; Wood et al., 2017; Engelhard et al., 2018). These findings suggest that the dopamine system may play important roles in sequence learning in general instead of its more limited role suggested by reinforcement learning models.

Basal Ganglia

It has been long noticed that the key feature of GRU networks, which is a gate controlled by an inhibitory signal that determines the maintenance of information, resembles the circuitry in the basal ganglia (Frank et al., 2001; O'Reilly and Frank, 2006). The basal ganglia receive broad input from all over the cortex, including sensory cortices and motor cortex (Engelhard et al., 2018), which comprises all the necessary information for learning sequences.

Our model may apply to the existing field of the basal ganglia research in several distinct directions, which have been largely studied separately. There is a lack of

frameworks that may successfully account for the studies from different approaches altogether.

First of all, our model prediction can be directly compared against previous studies of the basal ganglia's role in decision making. For example, Ding and Gold found caudate neurons represented accumulated information in a random dot motion discrimination task (Ding and Gold, 2010, 2012). Cisek and colleagues observed a group of neurons in the globus pallidus showed activity patterns similar to the urgency signal (Thura and Cisek, 2017). In rodent experiments, it has been reported lesions in the striatum led to deficits in evidence integration and produced a bias in decision making (Wang et al., 2018; Yartsev et al., 2018). Our model may account for the results from these studies.

Second, our model, which is trained to learn sequences, naturally explains the basal ganglia's role in performing sequential actions and in procedure memory (Jin and Costa, 2010; Wood et al., 2017; Geddes et al., 2018). It is conceivable that simply setting the input and output units according to the task requirement, we may train the model to perform sequential actions.

Third, the basal ganglia have also been indicated to play a central role in habitual and goal-directed behavior, both may be explained with our model (Graybiel, 1995; Graybiel and Grafton, 2015). On the one hand, our model may be used to generate responses of habitual behavior, which fits best a stable environment where the contingency between sensory, response, and reward events is fixed. For example, the probabilistic reasoning task we used to test the model was used in monkey studies in which the monkeys were extensively trained. The over-training of the task may allow the animals to perform the

task efficiently without the involvement of the brain areas of higher cognitive functions. On the other hand, our model may also contribute goal-directed learning. A key component of goal-directed behavior is the ability of assessing which actions may lead to a desired goal. Such assessment depends on the capability of making predictions based on the past sequence of events, which is exactly what our model can do.

Lesion Experiments

The simulated lesions may lead to future experimental work for testing the model. However, many of the lesion experiments, especially those done in monkeys, are done with direct manipulations of the activities of the neurons under investigation. Because the nature of the recurrent network structure, this type of the lesion, although appearing to be selective, may disturb the whole network and stop us from learning the actual functional role of each group of neurons. It is important to manipulate only the outputs to the downstream structures to avoid such problems. Recent developments of tools that allow manipulations of projection-specific neurons may eventually be useful in verifying the effects of our simulated lesions.

Predictive Coding and Bayesian Inference

Because of the symmetric structure of our model in which the action events are treated in the same way as sensory events, our model provides prediction of these events, which makes it a potential model for predictive coding (Friston, 2005). In a predictive coding framework, perceptual inference rests upon the brain's capability of making

predictions. The combination of the prediction and the external input may be solved with Bayesian inference, in which the prediction serves as the prior knowledge and new inputs are used to update the prior knowledge and generate behavior. In particular, the DDM can be regarded as a form of Bayesian inference. It has been shown that how neurons may implement Bayesian inference both theoretically and experimentally (Ma et al., 2006; Walker et al., 2018). Our model provides a solution of how the predictions may be generated from past experience. Although this is not the main focus of the current study, future studies may reveal the full potential of the model in this regard.

Summary

Our model in the current form is crude. We have only done limited investigation in the context of decision making. Yet, by illustrating the key features of the model, we hope that we have provided a proof of principle with this novel approach of modeling to generate interests in the field to explore its full potential.

Funding

This work was supported by Shanghai Municipal Science and Technology Major Project (Grant No.2018SHZDZX05) and by Strategic Priority Research Program of Chinese Academy of Science,Grant No. XDB32070100.

Acknowledgements

We thank Yang Xie, Chechang Nie, Wenyi Zhang for their help in all phases of the study, and Shan Yu for providing comments and advice. The authors declare no competing financial or nonfinancial interests.

Methods

Network

Our model is a network that contains three layers: the input layer (IL), the hidden layer based on gated recurrent units (GRUL), and the output layer (OL) (Fig 1a). The input layer contains units that carry the information about the sensory, motor, and reward events. (N_{IL}=20).

There is a total of 14 sensory input units. 10 of them represent the 10 shapes in the task. There are 3 units indicating the eye movement targets, including the fixation point, the left target, and the right target. We also include a unit that indicates the absence of any visual stimuli.

There are 4 action input units that represent the efference copies of the motor commands: 1 for fixation on the fixation point, 1 for saccading to and then fixating on the left target, 1 for saccading to and then fixating on the right target, and 1 for fixation breaks.

Finally, 2 reward input units are included: one for reward and one for the absence of reward.

The OL includes also 20 units that mirror the inputs.

We set the time step to 100 ms in the model training and testing. Our analyses and conclusions remain valid for smaller time steps.

Behavior Analysis

The network is fixed during testing. We generate 5000 random shape sequences of 25 shapes long as the testing data and feed them into the network model. A shape sequence is stopped whenever the output units associated with the saccades are triggered. If the network does not make a response before all 25 shapes have been presented, the trial is aborted (186.50 $\pm$ 3.42 (3.73 $\pm$ 0.68 %) trials in each run, excluded in further analyses).

<u>Subjective weights (Fig 2b):</u>

We perform a logistic regression to assess how each kind of shape affects the choice. The probability of choice is a function of the sum of leverages, Q, provided by each kind of shape:

$$P(choice = left) = \frac{1}{1+10^{-Q}}, \tag{13}$$

$$Q = \beta_0 + \sum_{m=1}^{10} \beta_m N_m , \tag{14}$$

where N_m represents the how many times shape m appears in a trial. β_0 is the bias term, $\beta_{1\sim10}$ are the estimates of how much weight the network model assigns to shapes 1 to 10, and are termed as the subjective weights of evidence (Yang and Shadlen, 2007). Since the regressors are not independent to each other, here we use the ridge

regression to minimize the variation of the estimation. The hyperparameter controls the trade-off between the cross- entropy loss and the L2-norm of the coefficients is used through a ten-fold cross validation for each fitting.

Shape order (Fig 2d):

To test how the shape order affects the network's choice, we perform a logistic regression on the trials with more than six epochs:

$$Q = \beta_0 + \sum_{n=1}^{3} \beta_n \mathrm{logLR_n} + \beta_4 \sum_{n=3}^{N-3} \mathrm{logLR_n} + \sum_{n=1}^{2} \beta_{n+4} \mathrm{logLR_{N-n}}, \quad (15)$$

where $\mathrm{logLR_n}$ is the logLR of the shape in epoch n and N is the total number of epochs in a sequence. β_0 is the bias term, β_1, β_2, β_3 are the fitting coefficients of the shapes in the first three epochs, β_4 the middle epochs, and β_5, β_6 the second and the third epochs to the last. The regression is done without the final shape (n-th). This is because the last shape is almost always (in more than 99 percent of trials) consistent with the choice. In DDM, the last shape brings the total logLR over the bound. The fitting procedure is similar to the procedure that we use to estimate the subjective value of each shape.

Analysis of Unit Responses:

Selectivity (Fig 3ab):

We test whether each neuron's activity in the GRUL is modulated by the total logLR, absolute value of the total logLR, the urgency, and the choice outcome with a linear

regression. We align the hidden unit state h to the shape onset in each epoch and use a simple linear regression to characterize the unit's selectivity:

$$h_t = \beta_0 + \beta_1 \sum \mathrm{logLR_t} + \beta_2 \left|\sum \mathrm{logLR_t}\right| + \beta_3 u_t + \beta_4 c\,, \qquad (16)$$

where h_t is the unit's response at time t aligned to the shape onset, $\sum \mathrm{logLR_t}$ is the total $\mathrm{logLR_t}$ of the shapes that have been presented by time t, u_t represents the urgency, which is quantified as the number of shape epochs, c represents the eventual choice of the network, which is set to 1 when the left target is chosen, -1 when the right target is chosen, and 0 if fixation is still maintained by the end of the epoch. The regression is performed on every time step during the shape representation. The significance of each variable is determined at $p<0.05$ with Bonferroni correction. We define a unit as selective to a variable if the variable shows a significant effect on the unit's activity at every time step in an epoch.

<u>Targeted dimension reduction (Fig 3cd)</u>

The analysis was previously described (Mante et al., 2013). Briefly, we first assess the selectivity of the hidden state $h_{i,t}$ of unit i at time t after the shape onset in each epoch to the eventual choice with a linear regression. Vectors $\beta_{c,t}$ parallel to the direction that captures the largest variance caused by the variation in choices at time t in the state space:

$$h_{i,t} = b_{i,t} + \beta_{i,t}\, c\,. \qquad (17)$$

Then we perform the principal component analysis (PCA) to decompose the units' response. The first N_{pca} principal components pc_a that together capture more than 80% of the variance are used to construct the de-noise matrix D:

$$D = \sum_{a=1}^{a=N_{pca}} pc_a \, pc_a^T. \tag{18}$$

Next, we de-noise the vectors $\beta_{i,t}$ by projecting them into the denoise subspace D and define the de-noise regression vector $\beta_{i,t}^{pca}$ with largest norm as time-independent de-noise regression vector:

$$\beta_{c,t}^{pca} = D\beta_{c,t}\,,$$

$$\beta_{c,t}^{max} = \beta_{c,t_{max}}^{pca}\,, \text{with } t_{max} = argmax_t \left|\beta_{c,t}^{pca}\right|. \tag{19}$$

Finally, to understand how the population activities are regulated by the eventual choices, we project the hidden unit state h to the vector $\beta_{c,t}^{max}$.

<u>Response Variability (Fig 3e)</u>

The response variability is computed in the PCA space. We first construct the task-relevant subspace S using the first N_{pca} principal components that together capture more than 80% of the variance. For each time step t in each trial, we project the hidden unit state h_t to V_t in the subspace S. The response variability is defined as the mean distance between V_t and the vector average of V_t across the selected trials.

Network analysis (Fig 4cd):

Geodesic distance and max flow are used to quantify how much information may be transferred between two nodes in a graph. We use weight matrix u_h, which represents the connection strengths between units in the GRUL, to construct a directed weighted graph. Each neuron in the GRUL corresponds to a node in the graph. Connections with highest 30% of the absolute value of the connection strength are turned into the edges in the graph. The results hold if we keep the highest 10% or 50%. The weight and capacity of each edge are defined as the absolute value of the original connection strength.

We define the geodesic distance from node A to target B as the inverse of the weight of the edge from node A to B (Newman, 2001). The geodesic distance between any pairs are calculated with Dijkstra's algorithm (Ahuja et al., 1993). The distance between the pairs that are not connected is defined as infinite. To account for these pairs, we calculate the mean of the inverse of geodesic distance across all unit pairs in each group.

References

Ahuja RK, Magnanti TL, Orlin JB (1993) Network Flows: Theory, Algorithms, and Applications | Pearson. Pearson.

Brown SD, Heathcote A (2008) The simplest complete model of choice response time: Linear ballistic accumulation. Cogn Psychol 57:153–178.

Cho K, van Merrienboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation.

Chung J, Gulcehre C, Cho K, Bengio Y (2014) Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling.

Churchland AK, Kiani R, Chaudhuri R, Wang X-J, Pouget A, Shadlen MN (2011) Variance as a Signature of Neural Computations during Decision Making. Neuron 69:818–831.

Cisek P, Kalaska JF (2005) Neural Correlates of Reaching Decisions in Dorsal Premotor Cortex: Specification of Multiple Direction Choices and Final Selection of Action. Neuron 45:801–814.

Cisek P, Puskas GA, El-Murr S (2009) Decisions in changing conditions: the urgency-gating model. J Neurosci 29:11560–11571.

Ding L, Gold JI (2010) Caudate encodes multiple computations for perceptual decisions. J Neurosci 30:15747–15759.

Ding L, Gold JI (2012) Separate, causal roles of the caudate in saccadic choice and execution in a perceptual decision task. Neuron 75:865–874.

Engelhard B, Finkelstein J, Cox J, Fleming W, Jang HJ, Ornelas S, Koay SA, Thiberge S, Daw N, Tank D, Witten I (2018) Specialized and spatially organized coding of

808 sensory, motor, and cognitive variables in midbrain dopamine neurons.
809 bioRxiv:456194.
810 Frank MJ, Loughry B, O'Reilly RC (2001) Interactions between frontal cortex and basal
811 ganglia in working memory: a computational model. Cogn Affect Behav Neurosci
812 1:137–160.
813 Friston K (2005) A theory of cortical responses. Philos Trans R Soc B Biol Sci 360:815–
814 836.
815 Geddes CE, Li H, Jin X (2018) Optogenetic Editing Reveals the Hierarchical Organization
816 of Learned Action Sequences. Cell 174:32–43.e15.
817 Gläscher J, Daw N, Dayan P, O'Doherty JP (2010) States versus rewards: dissociable
818 neural prediction error signals underlying model-based and model-free
819 reinforcement learning. Neuron 66:585–595.
820 Gold JI, Shadlen MN (2000) Representation of a perceptual decision in developing
821 oculomotor commands. Nature 404:390–394.
822 Graybiel AM (1995) Building action repertoires: memory and learning functions of the
823 basal ganglia. Curr Opin Neurobiol 5:733–741.
824 Graybiel AM, Grafton ST (2015) The striatum: where skills and habits meet. Cold Spring
825 Harb Perspect Biol 7:a021691.
826 Hanks T, Kiani R, Shadlen MN (2014) A neural mechanism of speed-accuracy tradeoff in
827 macaque area LIP. Elife 3.
828 Hanks TD, Kopec CD, Brunton BW, Duan CA, Erlich JC, Brody CD (2015) Distinct
829 relationships of parietal and prefrontal cortices to evidence accumulation. Nature
830 520:220–223.
831 Heitz RP, Schall JD (2012) Neural mechanisms of speed-accuracy tradeoff. Neuron.
832 Jang AI, Costa VD, Rudebeck PH, Chudasama Y, Murray EA, Averbeck BB (2015) The

Role of Frontal Cortical and Medial-Temporal Lobe Brain Areas in Learning a Bayesian Prior Belief on Reversals. J Neurosci 35:11751–11760.

Jin X, Costa RM (2010) Start/stop signals emerge in nigrostriatal circuits during sequence learning. Nature 466:457–462.

Kim J-N, Shadlen MN (1999) Neural correlates of a decision in the dorsolateral prefrontal cortex of the macaque. Nat Neurosci 2:176–185.

Kira S, Yang T, Shadlen MNN (2015) A neural implementation of Wald's sequential probability ratio test. Neuron 85:861–873.

Lotter W, Kreiman G, Cox D (2018) A neural network trained to predict future video frames mimics critical properties of biological neuronal responses and perception. arXiv.org.

Ma WJ, Beck JM, Latham PE, Pouget A (2006) Bayesian inference with probabilistic population codes. Nat Neurosci 9:1432–1438.

Mante V, Sussillo D, Shenoy K V., Newsome WT (2013) Context-dependent computation by recurrent dynamics in prefrontal cortex. Nature 503:78–84.

Neuringer A (2002) Operant variability: Evidence, functions, and theory. Psychon Bull Rev 9:672–705.

Newman MEJ (2001) Scientific collaboration networks . II . Shortest paths , weighted networks , and centrality. 64:1–7.

O'Reilly RC, Frank MJ (2006) Making working memory work: A computational model of learning in the prefrontal cortex and basal ganglia. Neural Comput 18:283–328.

Ratcliff R (1978) A theory of memory retrieval. Psychol Rev 85:59–108.

Roitman JD, Shadlen MN (2002) Response of Neurons in the Lateral Intraparietal Area during a Combined Visual Discrimination Reaction Time Task. J Neurosci 22:9475–9489.

858 Schultz W, Dayan P, Montague PR (1997) Neural Substrate of Prediction and. Science
859 (80-) 275:1593–1599.
860 Soltani A, Wang X-J (2010) Synaptic computation underlying probabilistic inference. Nat
861 Neurosci 13:112–119.
862 Stone M (1960) Models for choice-reaction time. Psychometrika 25:251–260.
863 Thura D, Cisek P (2017) The Basal Ganglia Do Not Select Reach Targets but Control the
864 Urgency of Commitment. Neuron 95:1160–1170.e5.
865 Usher M, McClelland JL (2001) The time course of perceptual choice: The leaky,
866 competing accumulator model. Psychol Rev 108:550–592.
867 Wald A, Wolfowitz J (1948) Optimum Character of the Sequential Probability Ratio Test.
868 Ann Math Stat 19:326–339.
869 Walker EY, Cotton RJ, Ma WJ, Tolias AS (2018) A neural basis of probabilistic
870 computation in visual cortex. bioRxiv:365973.
871 Walton ME, Behrens TEJ, Buckley MJ, Rudebeck PH, Rushworth MFS (2010) Separable
872 learning systems in the macaque brain and the role of orbitofrontal cortex in
873 contingent learning. Neuron 65:927–939.
874 Wang L, Rangarajan K V., Gerfen CR, Krauzlis RJ (2018) Activation of Striatal Neurons
875 Causes a Perceptual Decision Bias during Visual Change Detection in Mice. Neuron
876 97:1369–1381.e5.
877 Wang X-J (2002) Probabilistic decision making by slow reverberation in cortical
878 circuits. Neuron 36:955–968.
879 Wassum KM, Ostlund SB, Maidment NT (2012) Phasic Mesolimbic Dopamine Signaling
880 Precedes and Predicts Performance of a Self-Initiated Action Sequence Task. Biol
881 Psychiatry 71:846–854.
882 Wong K-F, Wang X-J (2006) A Recurrent Network Mechanism of Time Integration in

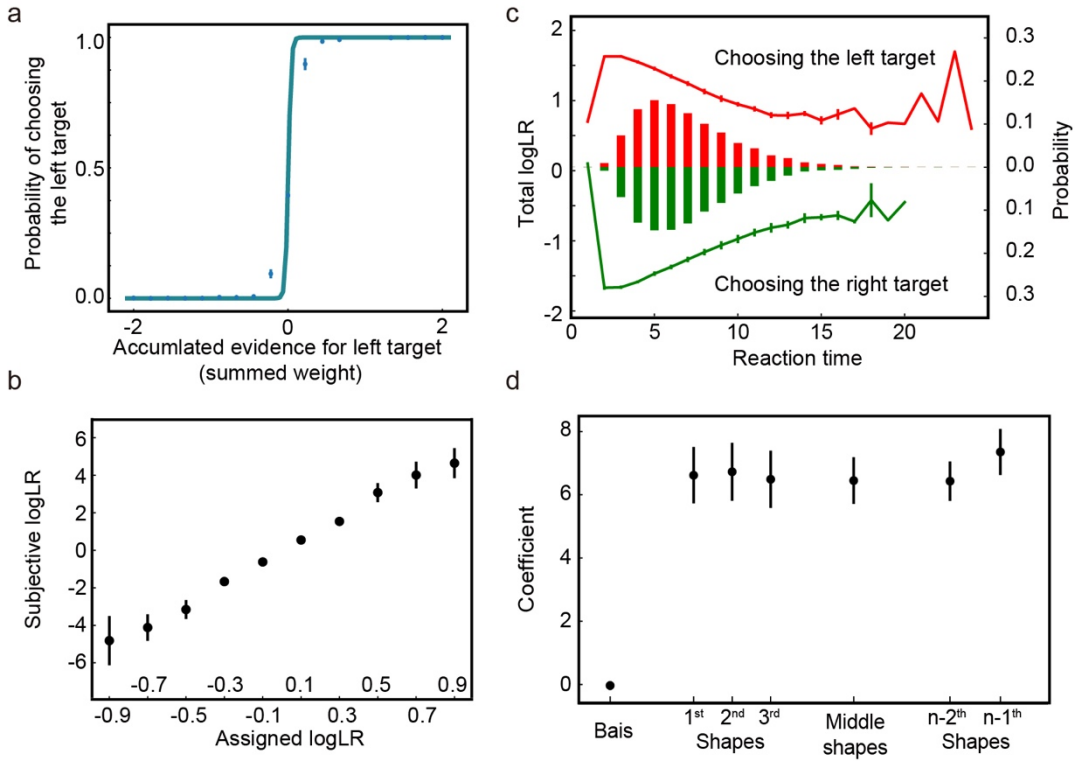
Perceptual Decisions. J Neurosci 26:1314–1328.

Wood J, Simon NW, Koerner FS, Kass RE, Moghaddam B (2017) Networks of VTA Neurons Encode Real-Time Information about Uncertain Numbers of Actions Executed to Earn a Reward. Front Behav Neurosci 11:140.

Yang GR, Joglekar MR, Song HF, Newsome WT, Wang X-J (2019) Task representations in neural networks trained to perform many cognitive tasks. Nat Neurosci:1.

Yang T, Shadlen MN (2007) Probabilistic reasoning by neurons. Nature 447:1075–1080.

Yartsev MM, Hanks TD, Yoon AM, Brody CD (2018) Causal contribution and dynamical encoding in the striatum during evidence accumulation. Elife 7:1–24.



Supplementary Figure 1. The network performance after training with only 1000 unique sequences. Same format as in Figure 2.

900