Research Strategy

(a) Significance

Artificial intelligence and neuroscience. The field of artificial intelligence (AI) has made remarkable advances over the last several years[1–4]. Historically neurobiology and AI have had fruitful interplays, providing key insights to one another[5]. Indeed, a major tool of AI, the neural network, was inspired by the brain's neural circuits[6]; and the architecture of deep learning networks was inspired by the visual system[1,7]. Furthermore, the learning algorithms used in another branch of AI, reinforcement learning (RL), were rooted in learning theories in animal psychology[8–10]. The mathematical algorithms for RL, in turn, facilitated our understanding of the midbrain dopamine system in the brain[11]. Although many AI ideas have been rooted in neurobiology and psychology, the field of AI has recently made independent progress in algorithms and network architectures that proved efficient *in silico*[1–4]. These rapid advances in AI open up new opportunities in neurobiology. Are these new, state-of-the-art AI algorithms actually used in the brain?

Reinforcement learning (RL) and its recent development in AI. RL is the branch of AI that provides algorithms by which an agent learns to maximize its future rewards by trial-and-error[10]. In one common approach in RL, an agent learns values associated with different states or state-action combinations. In RL, learning of values is typically driven by reward prediction errors (RPEs). REPs are defined as the discrepancy between the actual and predicted reward, which, in temporal difference (TD) learning, is the discrepancy between value predictions at consecutive time points. An agent learns a value function so as to *minimize* prediction errors.

Recent remarkable developments in AI have been, in part, driven by the incorporation and further development of RL algorithms within deep neural network architectures. In Mnih et al. (2015), an artificial neural network called **deep Q-network** (**DQN**) was shown to reach human-level performance in complex video games (Atari games), out-performing other state-of-the-art[2]. A DQN combines deep neural networks with Q-learning. A deep neural network is a multi-layer neural network and was used to process complex pixel inputs of video games. Q-learning is a class of RL algorithms that learns the value of each state-action combination[12]. In DQNs, synaptic weights of a network are modified so as to minimize the cost function, akin to RPEs used in conventional Q-learning algorithms[12].

Distributional RL. Since Mnih et al. (2015), various new algorithms have been developed to improve DQNs[13]. One promising such algorithm is a new type of RL algorithm, called **distributional RL**[14]. DQNs that implemented distributional RL, instead of traditional RL, featured significant performance gains[13,14]. Their key difference to traditional RL algorithms is how they predict future rewards. In environments in which rewards are probabilistic with respect to its occurrence and size, traditional RL algorithms learn to predict a single quantity, the *average* over all potential rewards, weighted by their respective probabilities (**Fig. 1A**). Distributional RL, by contrast, learns to predict the *entire distribution* over rewards (**Fig. 1C**).

In a specific type of distributional RL called **quantile distributional RL** (or distributional RL with quantile regression), each value predictor learns to predict a particular *quantile* (x^{th} quantile) of the target distribution (quantiles are points in a distribution that split the total population into equally-sized bins)[15] (**Fig. 1C**). For example, the 50% quantile corresponds to the median of the distribution, and the 75% quantile corresponds to the point that divides the distribution into 75% and 25% of the total number of samples. If there are enough value predictors that correspond to many different quantiles (ideally, equally spaced but

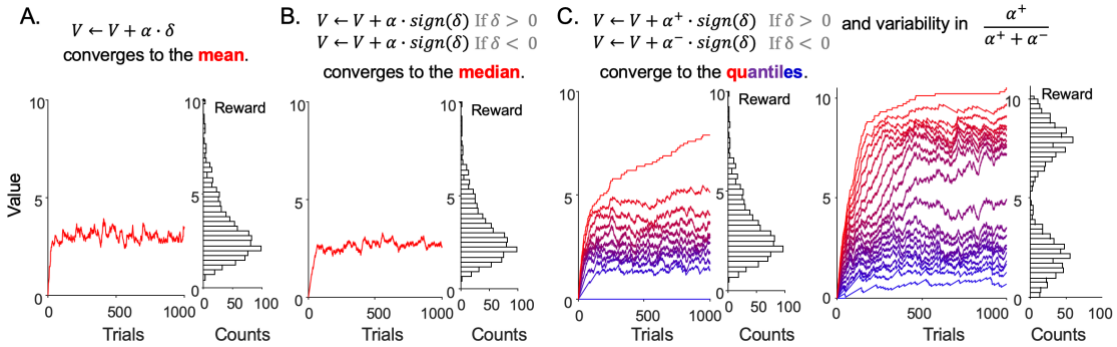

Fig. 1. Quantile distributional RL. A. Traditional RL. In each trial, the agent receives a particular amount of reward drawn from the target distribution (right). The predicted value (red) converges to the mean value (left). *V*, value; α, learning rate; δ, RPE. **B. RL with a binarized update rule**. The value prediction now converges to the *median* where the agent experiences the same *number* of positive and negative RPEs. **C. Quantile distributional RL**. Learning of two example distributions are shown. The variability in 'optimism' in learning rate, $\alpha^{+}/(\alpha^{+} + \alpha^{-})$, causes each value predictor to converge to $\alpha^{+}/(\alpha^{+} + \alpha^{-})$*th quantile*. The density of predicted values forms a population code that represents the probability density of the target distribution.

sufficiently large number of random quantiles suffice), the density of predicted values from different value predictors will correspond to the density of samples in the target distribution. In essence, a population of value predictors form a population code that represents the shape of the target distribution (**Fig. 1C**).

Distributional RL can arise from simple modifications of traditional RL. Remarkably, theoretical work has shown that quantile distributional RL can arise out of simple modifications of traditional RL (**Fig. 1, 2**)[15]. The first modification is **binarization** of the update rule (**Fig. 1B**). In traditional RL, fine-grained RPEs are used to update value prediction. This update rule drives the value prediction to convergence to the *average* of the target distribution. In quantile distributional RL, value predictions are updated by a predetermined, fixed amount. The consequence of this binarized update rule is that the update rule now only cares about the *number of times* that a positive or negative RPE it experiences, regardless of the magnitude of the RPEs. The value predictor eventually reaches the equilibrium where positive and negative RPEs occur the same number of times – i.e. the median of the distribution. The second modification is the addition of structured **variability** to RPE signals (**Fig. 1C**). Learning of value predictors is driven by a diverse set of RPEs that differ in terms of their 'optimism', with some emphasizing positive RPEs ('optimistic' RPEs) while others emphasizing negative RPEs ('pessimistic' RPEs). This RPE variability causes different value predictors to converge to different value levels. Combining the two ingredients – binarized update and diversity in the 'optimism' in RPEs – each value predictor now converges to a particular quantile.

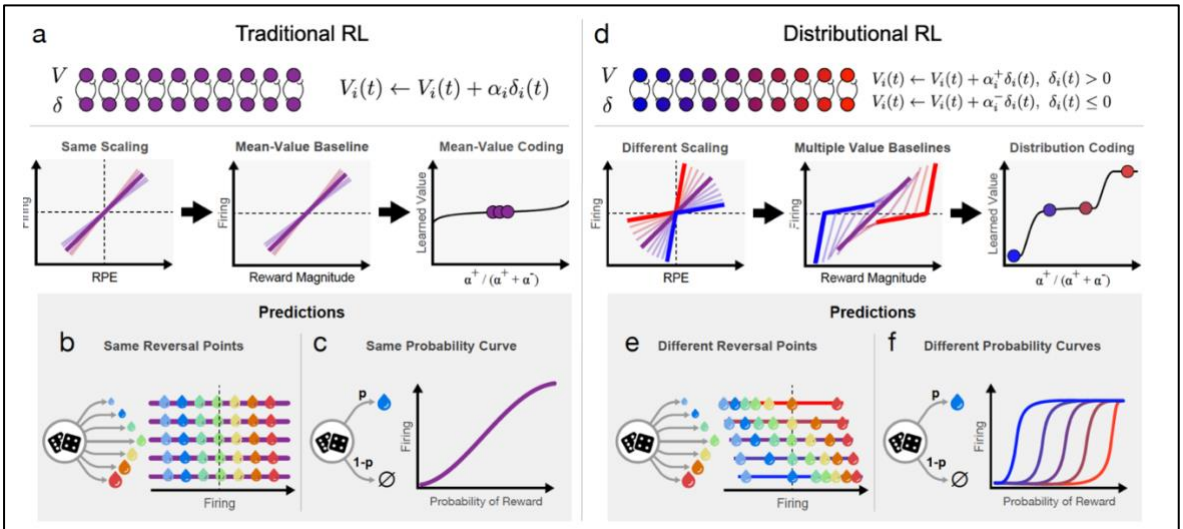

Fig. 2. Distributional RL can arise from a diversity in dopamine RPEs. a. In traditional RL, all value predictors learn the same value *V*. Each dopamine cell is assumed to have the same relative scaling for positive and negative RPEs (lower left). This causes each value prediction (or value baseline) to be the mean of the outcome distribution (lower middle). Collectively, these value predictions code only for the mean of the distribution (lower right). Dotted lines indicate zero RPE or pre-stimulus firing. **b**, Consider a situation where there is a chance of receiving different possible reward magnitudes (the variable magnitude task). Traditional RL predicts that for all cells, at the time of reward delivery, a reward larger than the expected magnitude (in this case, the reward volume shaded in green) should produce a positive RPE, while a smaller reward should produce a negative RPE. Each row is the responses of one hypothetical cell, and the dotted line indicates baseline firing. **c**, If the reward magnitude is held constant but the probability of reward is varied, with probability indicated by a cue preceding the outcome, all cells' responses to the cue should follow the same curve as a function of probability. **d**, In distributional RL, different channels have different relative learning rates for positive RPEs (α^+) and negative RPEs (α^-). Red colors indicate $\alpha^+ > \alpha^-$ (optimistic), and blue colors indicate $\alpha^+ < \alpha^-$ (pessimistic). An imbalance between α^+ and α^- causes each channel to learn a different value prediction. This set of value predictions collectively represents the distribution over possible rewards. **e**, In the variable magnitude task, different value predictions lead to a diversity of RPE reversal points, with some cells producing positive and other cells negative RPEs for the same reward magnitude. **f**, In the probabilistic reward task, distributional RL predicts that different cells will have response curves shifted left or right as a function of reward probability.

A formal expression of the learning rule and relaxation of binarization requirement for distributional RL. More formally, the update rule of traditional RL can be written as:

$$V(x) \leftarrow V(x) + \alpha \cdot \delta \quad (1)$$

where x is a state, $V(x)$ is the state value, δ is RPE, and α is the learning rate. For quantile distributional RL, the learning rate parameter might differ for positive and negative RPEs (α^+ and α^-, respectively):

$$V(x) \leftarrow V(x) + \alpha^+ \cdot sign(\delta) \quad for\ \delta > 0 \quad (2)$$

$$V(x) \leftarrow V(x) + \alpha^- \cdot sign(\delta) \quad for\ \delta < 0 \quad (3)$$

The sign function binarizes RPEs (δ) into -1 or 1 (modification #1). The optimism in RPEs discussed above corresponds to the ratio between α^+ and α^-. The modification #2 (variability), thus, corresponds to having

diversity in the quantity $\frac{\alpha^+}{\alpha^+ + \alpha^-}$. The learning rule described in equations 2 and 3 can be further generalized as follows:

$$V(x) \leftarrow V(x) + \alpha^+ \cdot f(\delta) \quad for\ \delta > 0 \tag{4}$$

$$V(x) \leftarrow V(x) + \alpha^- \cdot f(\delta) \quad for\ \delta < 0 \tag{5}$$

where f is a function that transforms the RPE (δ). Depending of the function f, value predictions converge to different quantities (**Fig. 3**). First, if f is a binary function (e.g. $f(x) = sign(x)$), the value predictions will converge to the $\frac{\alpha^+}{\alpha^+ + \alpha^-}$*th* quantiles of the target distribution as discussed in the previous section. Second, if f is a linear function (e.g. $f(x) = x$), the value predictions will converge to *expectiles*. This form of value representation still reflects the shape of the target distribution. Third, if f is a κ-saturating function (or sigmoid), value predictions will be intermediate between the two. More generally, for any non-decreasing f, the value predictions from different value predictors will together reflect the shape of the target distribution. In this sense, distributional RL does not necessarily require modification #1 (binarization).

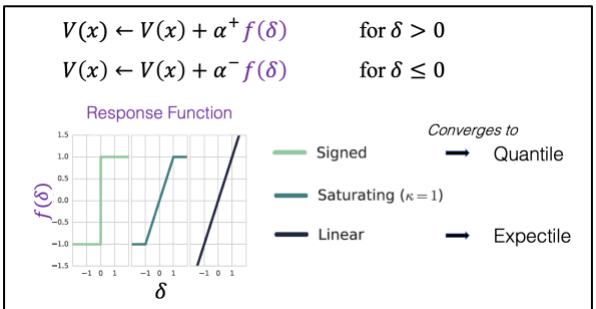

Fig. 3. Different response functions cause value predictors to converge to different quantities.

Terminology. Distributional RL (or distributional coding) can be divided into multiple types, based on how it represents a distribution[14,15]. In the first, 'categorical' type (similar to a histogram), the support of value representation is fixed (e.g. 'bins' in a histogram). The second type assumes a particular parametric form of the distribution (e.g., a Gaussian), and it represents this distribution by a set of parameters, such as its mean and variance. The third, '**quantile-like**' type represents the distribution by specific quantities that partition the target distribution. In this scheme, each unit of a population represents a unique quantity such as a quantile, an expectile, or something in-between. In categorical representations, each unit always represents the same value (e.g. activity directly represents 'probability' of a fixed value 'bin'). By contrast, in quantile-like representations, each unit changes the value that it represents during learning while maintaining what statistics (quantile, expectile etc.) it represents. **In this proposal, we focus mostly on quantile-like representations, and 'distributional RL' will refer to this representation generally unless further specified**.

Alternative neural population codes to encode probability distributions. If neural populations represent reward distributions with quantile-like representations, they do so by encoding their quantiles (or expectiles, etc.) in neural activity. We will refer to such an encoding as a **quantile representation code**.

Quantile representation codes are a novel type of population code, and differs from previously suggested population codes that encode probability distributions by their parameters[16]. The two best-established such codes are probabilistic population codes (PPCs[17,18]; we focus on a variant called linear invariant PPCs[18]) and distributed distributional codes (DDCs[19,20]). They both assume that the encoded distributions are members of a family known as *the exponential family of probability distributions*[21]. This family includes a wide range of regularly used distribution types, such as the Gaussian, Poisson, and Gamma distributions. PPCs assume that neural population activity is a linear function of the *natural parameters* of these distributions. For example, for a Gaussian distribution with mean μ and variance σ^2, the activity of each neuron would be the weighted sum of the Gaussian's natural parameters, μ/σ^2 and $1/\sigma^2$, with different weights across neurons. DDCs, in contrast, assume that neural population codes linearly encode the *expectation parameters*, which, for a Gaussian, are μ and $\mu^2 + \sigma^2$. Therefore, for the same type of encoded distributions, they make different predictions about how neural population activity ought to vary with the distribution's parameters. This makes them in principle experimentally distinguishable, but no previous work has attempted to do so.

As quantile representation codes do not a-priori make any assumption about the distribution type, they are neither strictly PPCs nor DDCs. However, given the flexibility of each of these neural coding schemes, what might appear to us as a quantile representation code might in fact be a PPC or a DDC. Telling them apart is challenging, as these different codes have never been fully characterized and compared. Further elucidating how these coding schemes differ, and combining high throughput imaging techniques with the relative simplicity of dopamine neuron (DAN) response signals provides a unique opportunity to formally distinguish these competing models.

Does the brain use distributional RL? The above framework of distributional RL provides a set of predictions and assumptions that can be tested experimentally in animals. In this project, we aim to test these predictions

and assumptions experimentally in mice. In parallel with experiments, we will theoretically examine the nature of population codes for distribution. The theoretical investigations will, in turn, provide crucial insights into how to optimize experimental designs, and subsequent interpretations.

Overall significance. If we can show that the brain uses distributional RL, this is quite significant. First, how the brain represents a distribution (probability density function) remains elusive. Second, quantile-like distributional RL is a totally novel 'format' of population coding. One big advantage is that, unlike other distributional coding schemes, it does not assume a particular distribution (i.e. it is non-parametric, allowing for substantial flexibility in neural encoding of distributions). Third, distributional RL also provides a mechanism by which a distributional code can arise through RPE signals. Fourth, although diversity of dopamine signals is often taken as evidence against the RPE hypothesis of dopamine signals, distributional RL provides a normative perspective on a certain type of diversity of dopamine neurons (DANs). Overall, distributional RL provides a novel theoretical perspective on neural representations, population coding, and RL in the brain. Finding that the brain uses distributional RL will have a major impact on both AI and neuroscience.

(b) Innovation

Distributional RL. This project combines a cutting-edge theory in artificial intelligence (machine learning) with high-quality neurobiological experiments. Distributional RL is a novel idea which is regarded as a major advance in RL algorithms. The concept that the midbrain dopamine system uses distributional RL is innovative.

Integration of new techniques. This project combines various novel neurobiological techniques. First, we will use two-photon calcium imaging with GRIN lens to image single neurons in behaving animals. This technique allows not only for imaging ensemble of neurons in a cell-type specific manner but also allows for recording the same neurons across different days and tasks, which is important for testing specific hypotheses. We will also use novel high-density electrophysiological recording methods (Neuropixels[22]). We will be able to record many neurons simultaneously, and from different brain areas, to test the nature of population coding more directly than if we were pooling data across sessions.

(c) Approach

Overall strategy

Aims. The main question of this proposal is whether and how a population of neurons represent a distribution of values and how this relates to RL. The framework of distributional RL provides a particular way of representing distributions and learning them through RPE signals. We will first compare the type of population code used in quantile-like RL against other coding schemes used to represent probability distributions theoretically. We will then conduct neurophysiological recording experiments to test a set of predictions regarding (1) how a population of neurons represent distributions (probability density function) over reward amounts, and (2) whether the experimental data support quantile-like distributional RL.

The main experiments will involve recording the activity of neurons that signal RPEs and expected values. A body of evidence indicates that DANs in the ventral tegmental area (VTA) signal RPEs[11,23–28]. It has also been shown that there are neurons that represent reward expectation in various regions of the brain[25,29,30]. For RPE signals, we will focus on **VTA DANs.** For value representations, we will examine the **ventral striatum** (**VS**, nucleus accumbens)[31–37] and the lateral **orbitofrontal cortex** (**OFC**)[30,38–45] in which reward expectation-related signals have been reported previously.

Why mice? We will use mice because using mice allows us to take advantage of various cutting-edge techniques to monitor and manipulate cell-type specific neuronal activities[46,47]. The Uchida lab has extensive experience in characterizing VTA DANs in mice[37,48–58]. These studies have strongly indicated that VTA DANs in mice signal RPEs both during phasic responses[27,28,48,51,52] and seconds-timescale slow fluctuations[59]. The Uchida lab has also recorded from the striatum[35] and OFC[38,60] in rodents.

Collaboration. This work is a collaboration between three groups: the **Uchida lab** (Harvard University), the **Drugowitsch lab** (Harvard Medical School) and a group in **DeepMind** (Drs. Matthew Botvinick, Will Dabney, and Zeb Kurth-Nelson; see the support letter from Dr. Botvinick). We have started analyzing the data that the Uchida lab has collected[51,52] and the first set of analyses has revealed that DAN activity signatures are compatible with the idea that they implement quantile-like distributional RL (Dabney, Kurth-Nelson, Uchida, Starkweather, Hassabis, Munos and Botvinick, "A distributional code for value in dopamine-based reinforcement learning", *under review*). The project proposed in this grant extends this initial effort in two significant ways. First, the data that we have analyzed were collected for a different purpose[51,52] and do not

support addressing important issues that are at the heart of this proposal. We therefore aim to perform new experiments specifically designed to test distributional RL. Second, to facilitate experimental designs and data analysis, we aim for deeper understanding of the nature of population codes used in distributional RL. The **Drugowitsch lab** is an expert in probabilistic inference and probabilistic population codes (PPCs) and has started analyzing the difference between quantile-like population codes against pre-existing coding schemes (PPCs and DDCs). The Drugowitsch lab will explore these theoretical foundations and will use these findings to inform experimental designs and data analysis. The **DeepMind** group will continue to support our data analysis while exploring the issues in RL theories and AI more generally. The **Uchida lab** will perform experiments in mice.

Specific Aim 1: To characterize distributional reward codes, and to develop methods to identify them.

Rationale. Our ultimate goal is to identify the neural code underlying the representation of reward expectations and reward prediction errors. In particular, we want to distinguish between the representation of a single reward estimate, and representations of whole reward distributions. To achieve this goal, it is crucial to be able to distinguish the different potential representations in neural data. We will do so by characterizing these representations in more detail, and by developing methods to distinguish them in neural data.

Hypothesis 1.1: we can identify probabilistic reward coding in neural population activity, and the type of code used, by methods that combine Generalized Linear Models (GLMs) with Artificial Neural Networks (ANNs).

Hypothesis 1.2: Quantile-like representation codes support reward-related computations better than other parametric population codes.

Contemporary population coding theories (see background) suggest that reward distributions are encoded in neural population activity through multiple statistics of these distributions. This stands in contrast to current reward coding theories, that assume that the mean reward is the only encoded statistic (e.g., **Fig. 1A**). We hypothesize that the population activity in reward-coding areas encodes additional statistics, like the reward variance, or quantiles of the reward distribution. Different probabilistic coding types differ in which set of statistics they encode (**Fig. 4A**), the full breath of which remains to be characterized in detail. Therefore, we will initially remain agnostic about the exact set of encoded statistics that is encoded, by using ANNs to learn the encoded statistics (**Fig. 4C**). Instead, we aim to identify only its number, without making specific, potentially biasing, assumptions about which code is used. Finding support for more than one encoded statistic already indicates the use of a probabilistic code. As a next step, we will distinguish between different types of probabilistic codes, by identifying which statistics the neural populations encode (**Fig. 4D**). Lastly, we will characterize the computational benefits of the different distributional codes, in order to identify why the nervous system might choose one representation over another.

Methods

We will initially focus on the encoding of reward expectations. These might be encoded by VS and orbitofrontal cortex, or by VTA DANs at cue onset. Later, we will identify the associated coding of reward prediction errors upon receiving reward.

Identifying the number of encoded statistics. To identify the number of encoded statistics irrespective of their exact nature, we will model them by ANNs. We will use a small ANN with a single hidden layer that, for each cue, takes the reward distribution as inputs, and outputs a set of statistics (**Fig. 4C**). These statistics are fed into a GLM to model neural population activity (**Fig. 4B**; see below). To identify the number of encoded statistics, we will fit ANN/GLM models with different numbers of statistics to neural population activity (deconvolved calcium traces or spike counts, summed over relevant time period within each trial) by simultaneously fitting their ANN and GLM parameters, and will evaluate their fit quality on hold-out data. If the population encodes a single statistic, then a model that fits a single statistic should yield the best fit quality. We will identify the number of encoded statistic by selecting the model that has a significantly higher fit quality than models with fewer statistics.

Identifying the distributional code type. Once we have established that neural populations encode more than a single statistic, we can distinguish between different distributional code types. These types differ qualitatively in the set of encoded statistics (**Fig. 4A**). Each code can furthermore use a different number of statistics. For quantile-like codes, the statistics are a set of reward distribution quantiles, expectiles, or in-between. For PPCs and DDCs, the statistics depends on the assumed type of reward distribution (e.g., Gaussian, Laplace, …). We will characterize the statistics of different coding types, and – again using GLMs – evaluate in simulations of

neural population activity how well we can distinguish them. Finally, we will identify the code used by a neural population by comparing the fit quality of models assuming different code types to neural data.

Characterizing prediction errors. RPEs are thought to encode the difference between expected and observed rewards. In the context of distributional RL, we predict these RPEs to encode the difference between the distribution statistics informing reward expectations, and those associated with the observed reward. For quantile regression, for example, this leads to the RPEs shown in **Fig. 2D**. To predict the RPE signatures of different distributional code types, we will develop a unified mathematical framework that assumes that RPEs provide the learning signal to update the statistics of the encoded reward distribution. Applied to quantile regression codes, this framework yields the RPEs previously described (**Fig. 4E**). Furthermore, it allows us to predict how RPEs would differ across the different set of statistics that constitute the different types of codes. Again relying on GLM population coding, we will test in simulations how well we can distinguish between RPEs arising from different coding schemes, and finally apply the method to recorded neural population activity to identify different distributional coding schemes.

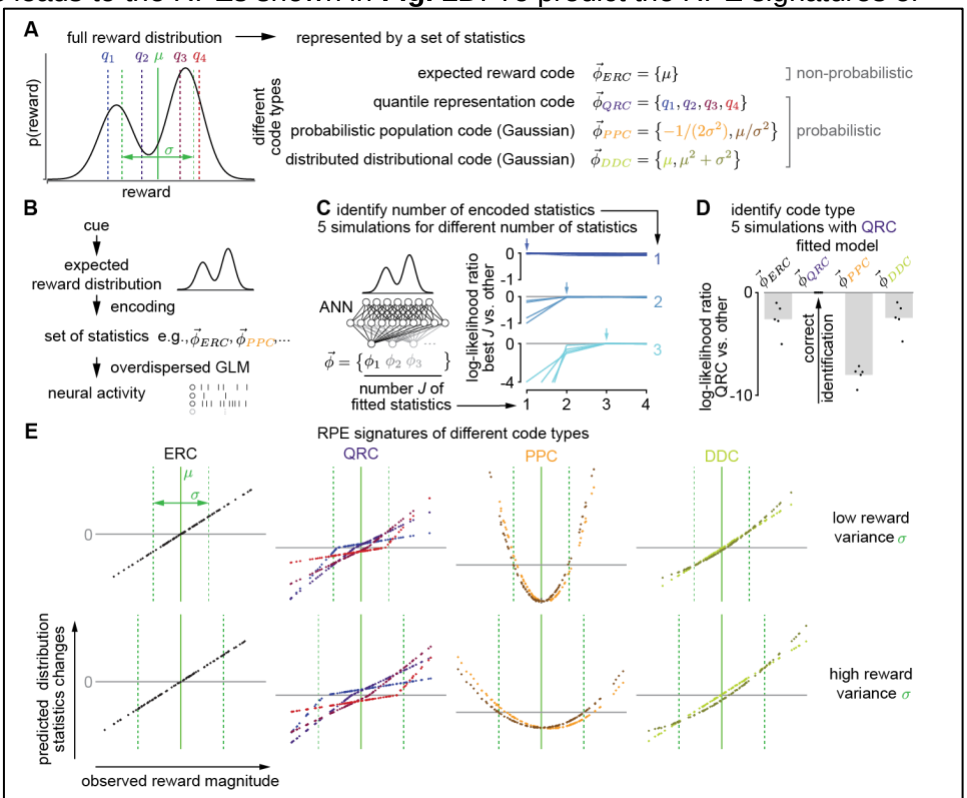

Fig. 4. A Different neural population codes assume the reward distribution to be represented by different sets of statistics (for PPCs/DDCs: Gaussian distribution type used as example). **B** GLM model for encoding reward distribution statistics in neural population. **C** Our ANN/GLM method can correctly recover the number of encoded statistics (arrows) from simulated population data while remaining agnostic to the code type. **D** It can furthermore distinguish between different distributional code types (dots = per-simulation test-set LLR, bars = across-simulations average). **E** Different code types predict different RPE signatures (each dot = predicted statistic change in single trial, dot color indicates specific statistic, μ/σ = mean/SD of actual reward distribution).

Using over-dispersed GLMs to link encoded statistics to neural population activity. We will link predicted across-trial variability in the set of statistics to observed across-trial variability in population activity by over-dispersed linear-nonlinear-Poisson models. These models assume that each neuron's activity is given by a weighted sum of encoded statistics, followed by a non-linear activation function, and a draw from a Poisson distribution (**Fig. 4B**). We will capture additional variability (e.g., additional non-reward-related variables, fluctuating statistics around steady-state) by instead using an over-dispersed Poisson (i.e., Negative Binomial[61]) distribution. Fitting such models is by now standard (e.g.,[62]), and requires fitting one set of weights and one dispersion parameter per neuron.

Characterizing the benefits of different distributional codes. Neural population codes of different formats pose different computational benefits. Linear PPCs excel at Bayesian inference[17,63], whereas DDCs are good at parameter learning[20]. In contrast, quantile representation code have so far only been investigated in the context of quantile-like distributional RL[64]. Identifying the computational benefits of different code types informs us about why the nervous system might use different codes in different circumstances, e.g., for the coding of reward. We will characterize the benefits of different code types by identifying the neural operations (e.g., summing input, divisive normalization) required to achieve specific kinds of computations. We will focus on reward-related computations, such as computing risk-sensitive reward expectations, comparing rewards, and combining reward expectations with state uncertainty. This will lead to predictions for the neural mechanisms at play when performing these computations, and how they differ for different types of codes.

Preliminary results

1. We confirmed in simulations that we can identify the number of encoded statistics in an experimentally plausible scenario (i.e., 12 simultaneously recorded neurons, 50 trials per reward cue with the reward statistics of task 2 (see Aim 2)). We simulated neural activity satisfying the assumptions underlying GLMs for different numbers of encoded statistics, and then applied the described method to the simulated data, fitting all parameters. As **Fig. 4C** shows, the cross-validated fit quality increases with the number of fitted statistics until the encoded number of statistics is reached, and slightly decreases thereafter due to over-fitting (Hypothesis 1.1).

2. Comparable simulations demonstrated that we can also identify the distributional code type. We simulated population activity with quantile regression codes, and assessed the model fit quality for different assumed codes. The cross-validated fit quality was highest for code type that generated the data (**Fig. 4D**). Changing the generative type yielded comparable results (not shown) (Hypothesis 1.1).

3. We have developed a uniform mathematical framework to predict RPEs, resulting in different, distinguishable RPE shapes for different code types. For rewards drawn from a Gaussian distribution, we show how the RPEs (illustrated in **Fig. 4E** in terms on predicted statistics changes) depend on the observed reward magnitude, and how this dependency changes with the reward distribution's variance (top vs. bottom). These RPEs shapes provide the basis for distinguishing different codes in DAN population activity (Hypothesis 1.1).

4. Decision-makers might prefer certain over uncertain rewards. It turns out that such risk-sensitive reward comparisons only require a weighted sum of neural activity for quantile regression codes, but significantly more complex neural operations for PPCs and DDCs (Hypothesis 1.2).

Expected results and interpretation

1. For PPCs and DDCs, our simulations have so far been restricted to assuming the encoding of Gaussian reward distributions. We will explore a larger set of distributions, their associated distributional statistics, and how easily they can be discriminated across different codes, and thus in neural population activity. We expect there to be an upper limit on the number of statistics that are discriminable across codes. Crossing this limit would imply that the same population activity effectively encodes multiple code types simultaneously.

2. In GLM simulations we will determine the largest set of different statistics we can distinguish in neural data for different population sizes and numbers of trials. This will characterize our method's ability to distinguish different distributional codes in neural population activity. Discriminability should increase with trial number, but, due to an increasing number of model parameters, might saturate with population size.

3. We will further characterize learning in distributional codes to provide more general predictions for the RPE signatures in neural activity. We will do so by considering larger sets of encoded statistics, and different learning schemes (e.g., maximum likelihood, Bayesian learning, etc.). This will yield a thorough characterization of the RPE signatures we predict for different kinds of distributional codes, which support fine-tuning the design of our experiments.

4. We will take similar approaches to previous work[17,63] to describe the neural mechanisms required to perform reward processing-related computations in the different code types. We expect quantile-like representation codes to yield simpler mechanisms to implement these computations than PPCs and DDCs.

Potential pitfalls and solutions

1. Due to the use of GLMs, we can only distinguish code types whose associated set of statistics is not co-linear. Furthermore, we cannot identify more statistics than the size of the recorded population. Therefore, we will start with small number of statistics for all codes, and increase them successively until they become close-to-colinear or we reach the recorded population size. If we approach close-to-colinearity, we attempt to increase statistical discrimination power by pooling trials across multiple days.

2. The mapping between objective and subjective reward magnitudes might be non-linear[65,66]. ANNs can capture these non-linearities (**Fig. 4C**), but distinguishing between different codes and their associated statistics (which relate to subjective, not objective rewards; **Fig. 4D**) requires us to explicitly model these non-linearities, for which we will use simple single-parameter functions to capture the shape of the utility function. Task 1 (Aim 2) support a direct measure of the reward response curve, providing additional constraints.

3. Additional non-reward-encoding variables, e.g., sensory or movement-related information, might introduce shared variability across simultaneously recorded neurons – variability not captured by over-dispersed GLMs. If we find that such shared variability perturbs our model fits, we will attempt to model it explicitly.

Specific Aim 2: To monitor RPE and value signals in the brain and test predictions of distributional RL.

Rationale. Whether and how a neuronal population represents a distribution remains elusive. We will start by testing the hypotheses agnostic to particular forms of distributional code using the insights obtained in Aim 1:

Hypothesis 2.1: We can decode various statistics of a distribution (not just the mean but also variance, quantiles or expectiles) or even the distribution shape from ensemble neuronal activities.

We will then use the insights obtained in Aim 1 to test the following hypothesis:

Hypothesis 2.2: We can identify the distributional code type from neuronal ensemble activities.

Next, we will more specifically examine distributional RL which makes specific predictions. We will start by testing hypotheses regarding dopamine RPE signals – the driving force of quantile-like distributional codes:

Hypothesis 2.3: There is substantial diversity in 'optimism' $(\alpha^+/(\alpha^+ + \alpha^-))$ **in dopamine RPEs.**

Hypothesis 2.4: The optimism biases of individual DANs are stable across days and tasks.

Hypothesis 2.5: The reversal points (zero-crossing points) of dopamine RPE signals are positively correlated with the 'optimism' $(\alpha^+/(\alpha^+ + \alpha^-))$ **of individual DANs.**

One key factor for distributional RL is diversity in the activity of DANs and in value predictors. Finding a significant extent of variability does support distributional RL, but we would like to test non-trivial predictions, like those of Hypotheses 2.4 and 2.5, that are unlikely to occur merely from a natural variability or noise.

Distributional RL also makes predictions regarding cue-evoked value-related activities in DANs as well as in value-coding neurons. We will therefore test these hypotheses:

Hypothesis 2.6. The population activity evoked by reward predictive cues is sensitive to the shape of distributions in accord with distributional RL.

Hypothesis 2.7. We can decode the shape of the distribution using a method that is derived from distributional RL.

Methods.

Animals. We will use male and female mice. If we encounter any differences between them, we will increase the sample size to examine whether the basic results hold for both sexes. Mice will be either C57BL/6J mice obtained from the Jackson Laboratory, or backcrossed with C57BL/6J mice for at least for 5 generations. We will use transgenic mice that express Cre recombinase under the control of the dopamine transporter gene (DAT-Cre[67]), Drd1-Cre and Adora2a-Cre (MMRRC Stock# 030989-UCD and 036158-UCD, respectively). For cell-type-specific calcium imaging, we will use a GCaMP6f reporter line, Ai148 (Jackson Lab Stock# 030328).

Biohazards. Experiments in this proposal use adeno-associated viruses (AAVs). They are used to exogenously express genes of interest in specific neurons in the brain. The viruses are handled using appropriate containment, and with the appropriate approvals from the Institutional Animal Care and Use Committee at Harvard University and from Environmental Health and Safety.

Scientific rigor and reproducibility. We will report all details of our experimental and analysis procedures in our papers, and will make the analysis code freely available on Github or MATLAB Central. We plan to make the original datasets freely available by formatting them using Neurodata Without Borders[68], and by depositing the data on a commonly used server (CRCNS). Sample size (number of cells, fields of view, animals) for each experiment will be estimated by power analysis using our preliminary data to define expected means and standard deviations. Statistical tests will be chosen based on the design of the experiments, fully taking into account the data distribution (normality, variance, etc.).

Two-photon calcium imaging using GRIN-lens. Our previous study used tetrode recording with optogenetic tagging to record the activity of VTA DANs. This method has allowed us to obtain typically 1-2 identified DANs in a session and occasionally a small ensemble (~4-5 identified DANs). **Optical imaging can provide recording of a larger neural ensemble and recording from the same neurons across multiple days and from different behavioral tasks**. A recent study has shown that the activity of 10-30 DANs can be simultaneously imaged from the VTA using a gradient-index (GRIN) lens[47]. To perform two-photon imaging of VTA DANs, we will use DAT-Cre mice crossed with the GCaMP6f reporter line Ai148 mice. For recording from direct or indirect pathway medium spiny neurons (MSNs) in the VS, we will use Drd1-Cre and Adora2a-Cre mice, respectively, crossed with Ai148 mice. Alternatively, we will inject AAV5-CAG-FLEX-GCaMP6f or AAV5-CAG-FLEX-GCaMP7f (UPENN vector core) in the respective Cre mice. We will implant a 0.6-mm diameter GRIN lens (NEM-060-25-10-920-S-1.5p, GrinTech) in the VTA, in the VS or in the lateral OFC. During the surgery, a headplate will be also implanted for head-fixation under the microscope. After full recovery, mice will be water deprived and gradually habituated to the imaging setup and head-fixation. We will use a custom-built two-photon microscope equipped with Ti:sapphire laser, GaAsP photomultipliers and ScanImage 4.0 software

to control the image acquisition. Frames with 512x512 pixel will be acquired at 15 or 30 Hz. The power at the exit of the objective (Nikon 20x, 0.5 NA) will be adjusted controlling a Pockel´s cell and will range from 15 to 70mW. We will perform motion correction and identification of single neurons/source extraction using available analysis pipelines (Suite2P[69] and/or CaImAn[70]). Calcium signals will be deconvolved to estimate underlying spike rates using OASIS[71] and registration will be performed across sessions.

Electrophysiological recording in behaving mice. First, to compare imaging data with spike recording, we will also conduct **tetrode recording with optogenetic identification of cell types** as described previously[37,48,50–52]. The Uchida lab has extensive experience in this technique[37,48,50–52]. Although this technique allows for simultaneous recording of only a small number of neurons, it provides a quantitative data that can aid interpretation of calcium imaging data. Second, we will use a high-density probe (**Neuropixels**[22]) to record from a dense population of neurons in the VS and OFC. The long shank of Neuropixels will allow us in some experiments to simultaneously record the activity of both OFC and VS. During surgery, mice will be implanted with a head-plate and a craniotomy will be made. During experiments, one Neuropixels probe will be lowered through the craniotomy using a micromanipulator. The data will be preprocessed and automatically spike sorted with Kilosort[72] as describe elsewhere[73].

Behavioral paradigms. We will train each mouse in the following two tasks (Task 1 and then Task 2). During imaging sessions, we will alternate between the two tasks and aim to obtain the activity of the same neurons across both tasks. A separate group of mice will be trained in Task 3. We will aim to collect 10-30 simultaneously recorded DAN ensembles from 5-10 animals in each task condition. We will aim to collect 20-50 simultaneously recorded direct and indirect pathway MSNs in the VS as well as 20-50 simultaneously recorded OFC neurons from 5-10 animals.

Task 1. Variable magnitude task. To examine how dopamine responses scale with RPEs, we will use the variable magnitude task similar to that used in our previous study[51,52]. This task allows us (1) to estimate the slope parameters for positive and negative RPEs and (2) to estimate the RPE reversal point for each neuron. We will perform a surgery to implant a head plate. After recovery, mice will be water-deprived. During experiments, mice will be head-fixed and trained to receive reward by licking a water spout. In 10% of trials an odor cue will be delivered that indicates that no reward will be delivered on that trial. In the remaining 90% of trials, one of the following reward magnitudes will be delivered, at random: 0.1, 0.3, 1.2, 2.5, 5, 10, 20 µL. In half of these trials, a reward will be provided 2 s after an odor cue (which indicates that a reward is forthcoming but does not disclose its magnitude). In the other half, it will be unsignaled.

Task 2. Variable distribution task. To examine whether neuronal responses are sensitive to the reward distribution shape, or more generally, how a population of neurons represents these reward distributions, we will devise a task in which different odor cues (conditioned stimuli or CSs) predict rewards with different distributions over reward amounts. In particular, we will include trial types which differ (1) by their variance (σ^2) but not by the mean (µ) or (2) by their shape but neither σ^2 nor µ (**Fig. 5**). In each trial the animal first experiences one of the odor cues for 1 s, followed by a 2 s pause, followed by either a reward of a particular amount drawn randomly from a predefined distribution. Odor meanings will be randomized across animals.

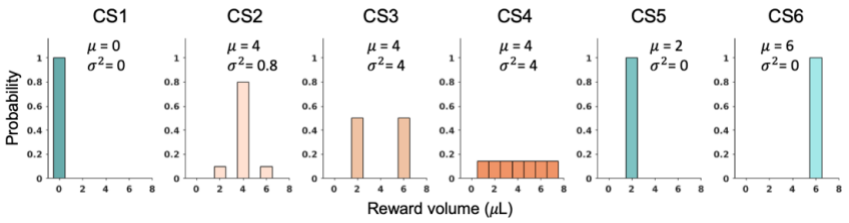

Fig. 5. Variable distribution task (Task 2). Each odor cue (CS, conditioned stimulus) is associated with a unique distribution of reward volumes.

Task 3. Variable distribution reversal task. A separate set of mice will be trained in Task 3. The goal of this task is to examine how ensemble neurons change their population activity patterns when distributions are changed. Two odor cues will be used, and one odor (CS-) will be associated with no outcome. The other odor (CS+) will predict a reward whose distribution is switched between two distributions across blocks of trials. Because switching distributions is expected to be more difficult than learning a predefined, steady distribution for each cue, we will use only two conditions (e.g. CS2 and CS3 or CS2 and CS4) in each animal. Distribution of reward will be reversed every 40 - 50 trials.

Preliminary results

1. We have established a method for GRIN-lens based 2-photon calcium imaging of VTA DANs (**Fig. 6**).
2. Hypotheses 2.1 & 2.2: we have performed simulations in Aim 1 (**Fig. 4**).

3. Hypotheses 2.3 & 2.5: we have analyzed our previous data[51,52] (**Fig. 6**). Testing hypothesis 2.4 requires recording from the same neuron from multiple days and tasks, which will be done using two-photon imaging.

4. Hypothesis 2.6: we have analyzed our previous data[51,52] (**Fig. 7**). Although this data does not allow us to directly address this question, the analysis demonstrated that the response to probabilistic reward (P=0.5) showed substantially greater variability across neurons that would be expected from linear interpolation.

Expected outcome and interpretation

1. Using methods from Aim 1, we will decode the number of statistics encoded in the population (see **Fig. 5C,D**) and we will be able to formally test whether DANs encode the full distribution of expected rewards or a summary statistic (mean, mean + variance, etc.).

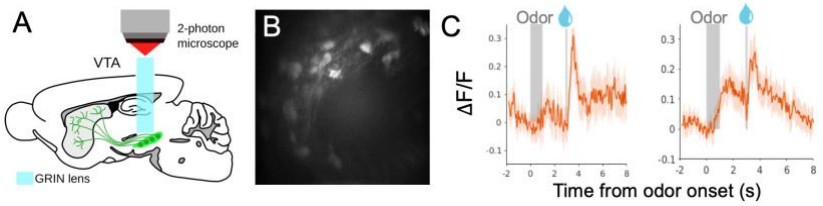

Fig. 6. 2-photon calcium imaging of VTA dopamine neurons using GRIN lens. A. Experiment. **B**. An image of GCaMP-labeled VTA DANs through GRIN lens. **C**. Example calcium trace of VTA DANs in a classical conditioning task. Odor cue at 0-1 sec (gray), reward at 3 sec.

2. We will obtain α^+ and α^- from dopamine neuron activities in Task 1. We will 'linearize' the dose response curve of each DAN by normalizing the x-axis based on the average dose response curve (akin to plotting response against RPEs in 'utility') (**Fig. 7A,B**). α^+ and α^- will be obtained as the slopes for positive and negative RPEs (the right and the left to the RPE reversal point) (**Fig. 7C,D**). We will first examine that $\alpha^+/(\alpha^+ + \alpha^-)$ provides a stable measure by comparing the values between early and late periods in a session. We will then examine whether $\alpha^+/(\alpha^+ + \alpha^-)$ tiles widely between 0 and 1 across neurons (Hypothesis 2.3) (**Fig. 7E**). We will examine both pooled data as well as data from single animals. The latter will ensure that the observed variability is not due to variation in the animal's state or recording conditions. Alternatively, all clustering close to 0.5 will be more consistent with conventional RL. We will also estimate α^+ and α^- using reward responses in CS4 trials in Task 2. To test Hypothesis 2.4, we will examine whether $(\alpha^+/(\alpha^+ + \alpha^-))$ is stable across days and Tasks 1 and 2 (Hypothesis 2.4). To test Hypothesis 2.5, we will examine whether $(\alpha^+/(\alpha^+ + \alpha^-))$ correlates with RPE reversal points of each neuron (**Fig. 7F**).

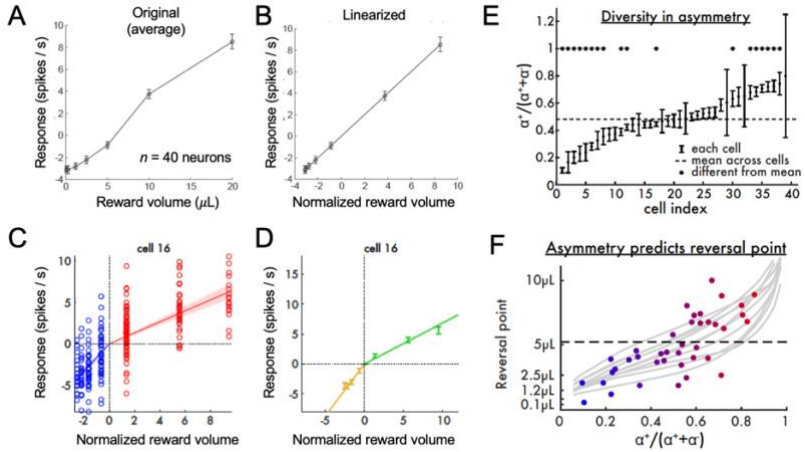

Fig. 7. Preliminary analysis of dopamine neurons' responses to predicted reward in Task 1 (Data from Eshel et al., 2015, 2016). A, B. The x-axis (reward volume) was transformed from A to B to linearize the dose response curve obtained in Task 1. **C. D.** Single neuron example. The slopes for positive and negative responses (RPEs) were then obtained (α^+ and α^-). **E**. The diversity in relative scaling of positive and negative responses in dopamine cells is statistically reliable, showing 95% confidence intervals for $\boldsymbol{\alpha^+/(\alpha^+ + \alpha^-)}$**. F**. The relative scaling of positive and negative responses predicts that cell's reversal point (each point is one dopamine cell). Dashed line is the mean over cells. Light gray traces show reversal points measured in a distributional TD simulation on the same task. Each trace is an independent run.

3. One key prediction of quantile-like distributional RL is that the variance of value-related activities across neurons will increase as the variance of the target distribution increases (**Fig. 8**). We will therefore examine whether the variance of cue-evoked responses across neurons in DANs, VS, and OFC will increase as the variance of the target distribution increases while the mean remains the same (CS2 versus CS3 or CS2 versus CS4) (Hypothesis 2.6). We will also examine whether the patterns of population activity changes when the shape of distribution is altered while the mean and variance remained the same (CS3 versus CS4) as predicted by quantile-like distributional RL models.

4. We will examine whether we can recover the shape of distributions based on cue-evoked responses across neurons based on a distributional RL scheme. For example, we will use $\alpha^+$and α^- from Task 1 to estimate the 'optimism' of each DANs. In Task 2, the magnitude of cue-evoked response in combination with the optimism of each neuron will be used to decode the exact shape of distribution over reward amounts predicted by each cue (Hypothesis 2.7).

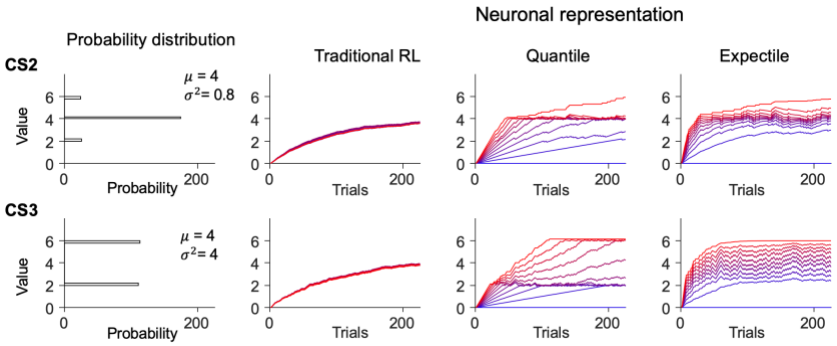

Fig. 8. Prediction. CS2 and CS3 trials differ in terms of reward volume distributions. CS3 trials have more variance in reward volumes than CS2 trials do. Accordingly, quantile- and expectile-distributional codes will have higher variance in neuronal representations than traditional RL.

Potential pitfalls and solutions.

1. The diversity that characterizes distributional RL must be distinguished from various kinds of 'noise' in the system, fluctuation due to uncontrolled parameters (e.g. thirst), between-session variability, or measurement noise. Hypotheses 2.4 and 2.5 (**Fig. 7F**) are non-trivial predictions that are unlikely to occur with these types of 'noise'. To further distinguish meaningful diversity from noise, we will also examine whether the diversity we observe is stable within a session by comparing the data from early and late periods in each session, compare the neuronal diversity with behavioral data (e.g., the vigor of licking), and examine whether the diversity exists in a population of simultaneously-recorded neurons. Furthermore, finding support for Hypothesis 2.7 will provide further evidence for distributional RL.

2. Negative RPEs may be coded by the length of pausing in dopamine neuron firing[74]. We will average firing rates over a relatively long time window to account for both firing rate changes and durations of pauses.

Specific Aim 3: To experimentally test tenets of distributional RL using causal manipulations.

Rationale. Quantile-like distributional RL makes some important assumptions. For one, it assumes a relative independence between the loops connecting value predictors and their corresponding DANs to maintain the variety of 'optimism' (see **Fig. 2d**). However, DANs tend to arborize their axons in the striatum (particularly in the dorsal striatum)[75]. Second, dopamine signals are the driving force of value representations downstream, and accordingly, these signals determine the RPE reversal points (zero-crossing points) of DANs (**Fig. 2**). These assumptions make testable experimental predictions. Our preliminary simulations have indicated that complete independence is not required to maintain a quantile-like distributional code (**Fig. 9**). Nonetheless, it is important to know whether different DANs maintain a certain level of independence.

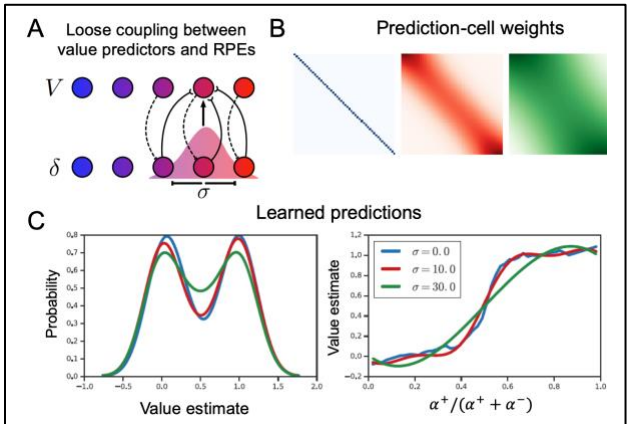

Fig. 9. Distributional code remains robust under loose coupling. A. Diagram illustrating Gaussian weighted topological mapping between RPEs and value predictors. **B**. Changing the standard deviation of the Gaussian induces different weights between the value predictors and the RPE cells. **C**. In a task with 50/50 chance of 1,0 or 0,0, we trained with distributional RL with different levels of coupling and observed that the distribution estimation is robust to the degree of coupling.

Optogenetics: We will optogenetically manipulate a small subset of DANs in VTA to make them more 'optimistic' or 'pessimistic' (transient optogenetic stimulation or inactivation during reward) while recording the activity of stimulated and unstimulated DANs.

Hypothesis 3.1: Optogenetic manipulation of a subset of DANs will alter the RPE reversal point of the activated neurons while leaving un-activated neurons relatively unaffected.

Manipulating habenula: We have previously shown that lesions of the habenula reduced the negative RPEs (or 'dip') caused by reward omission while positive RPEs remained relatively intact[50]. This raises the possibility that habenula lesions made α^- smaller relative to α^+ (thus more 'optimistic'). We will manipulate the activity of habenula neurons bidirectionally and will examine how this manipulation changes the responses of VTA DANs.

Hypothesis 3.2: Optogenetic manipulation of the lateral habenula (LHb) neurons will alter the learning rate parameters for negative RPEs (α^-) of VTA DANs.

Methods.

Manipulating DANs. We will express channelrhodopsin-2 (ChR2) or light-dependent chloride channel (GtACR) in VTA DANs. We will inject AAV5-FLEX-ChR2 or AAV5-FLEX-GtACR into the VTA of DAT-Cre mice unilaterally (*n* = 8 mice each). We will implant a tetrode drive[48] which houses two movable optical fibers (100μm diameter, 300μm apart) each of which is attached with 6 tetrodes. The two optical fibers will be targeted to the lateral and central regions of the VTA. After recovery, the animal will be trained in Task 1 (variable magnitude task). During recording sessions, we will activate/inactivate DANs through one of the fibers (30 Hz, 500 ms) during the delivery of a set of large rewards (5, 10, 20 μL). The activated fiber will be alternated every 4 days. We will record spikes of optogenetically-identified DANs. For control experiments, we will perform the same procedures after injecting AAV5-FLEX-EGFP into VTA (*n* = 8 mice each). In this control experiments, DANs will be identified based on their firing patterns[51,52].

Manipulating habenula. During surgery, we will inject AAV5-FLEX-ChR2 into the VTA of DAT-Cre mice unilaterally (this will be used for optogenetic identification during recording). We will also inject AAV5-Chrimson or AAV5-GtACR into the LHb ipsilateral to the VTA injection site (*n* = 8 mice each). We will implant a tetrode drive into the VTA (this will be used for manipulating habenula neuron activities). After recovery, the animal will be trained in Task 1 (variable magnitude task). During recording sessions, we will activate/inactivate LHb neurons in 25% of the total trials interleaved with unstimulated trials. We will record the spiking activity of optogenetically-identified DANs. For control experiments, we will perform the same procedures after injecting AAV5-FLEX-EGFP into the habenula (*n* = 8 mice each).

Preliminary results

1. We have extensive experience in manipulating the activity of VTA DANs[48,52]. We have also confirmed that VTA DANs can be inhibited by inhibitory opsins.

2. Our previous study showed that after habenula lesions, inhibitory responses caused by omission of predicted reward were greatly diminished while the excitatory responses remained relatively intact[50].

Expected outcome and interpretation

1. We will obtain the RPE reversal points of optogenetically-identified DANs in Task 1 as described in Aim 2 (**Fig. 7**). We will compare the reversal points between the groups of DANs recorded from the tetrodes at the manipulated and unmanipulated fibers. If Hypothesis 3.1 is true, the reversal points will be greater in the activated group and smaller in the inactivated group when compared to the unmanipulated group. These differences will not be observed in GFP control animals.

2. We will obtain the learning rate parameters (α^+ and α^-) in Task 1 as described in Aim 2 (**Fig. 7**). We will compare α^+ and α^- between trials with and without optogenetic manipulations in the habenula. If Hypothesis 3.2 is true, activation (or inactivation) of LHb neurons will increase (or decrease) α^-, but α^+ will remain relatively unchanged. These differences will not be observed in GFP control animals.

Potential pitfalls and solutions.

1. The number of DANs that we can identify in each session is not very high (1-4 neurons). Therefore, we will have to compare groups pooled across sessions and animals. For an alternative approach, we will consider combining GRIN-lens-based two-photon imaging and optogenetics[76].

2. Firing characteristics of DANs including the relevant parameters (reversal points, α^+ and α^-) might be different between the lateral and central VTA. We will counterbalance fibers with optogenetic manipulation by alternating the site of manipulation between the lateral and central VTA.

Timeline

All three aims will be performed in parallel. The Drugowitsch lab will simultaneously characterize in more detail the predictions of different code types, and their computational benefits (Aim 1), and perform theory-based data analysis (Aims 2 & 3), whenever required. In years 1-2, a stronger focus is on theory development. This focus will shift in years 3-5 to data analysis. The Uchida lab will perform experiments for Aim 2 and 3 in parallel. We will characterize VTA dopamine neuron activities in Task 1-2 (year 1-1.5) and Task 3 (year 1.5-3). We will then characterize the activity of VS and OFC neurons (year 4-5). For Aim 3, we will perform the manipulation of DANs first (year 1-2.5) and then LHb neurons (year 2.5-5).