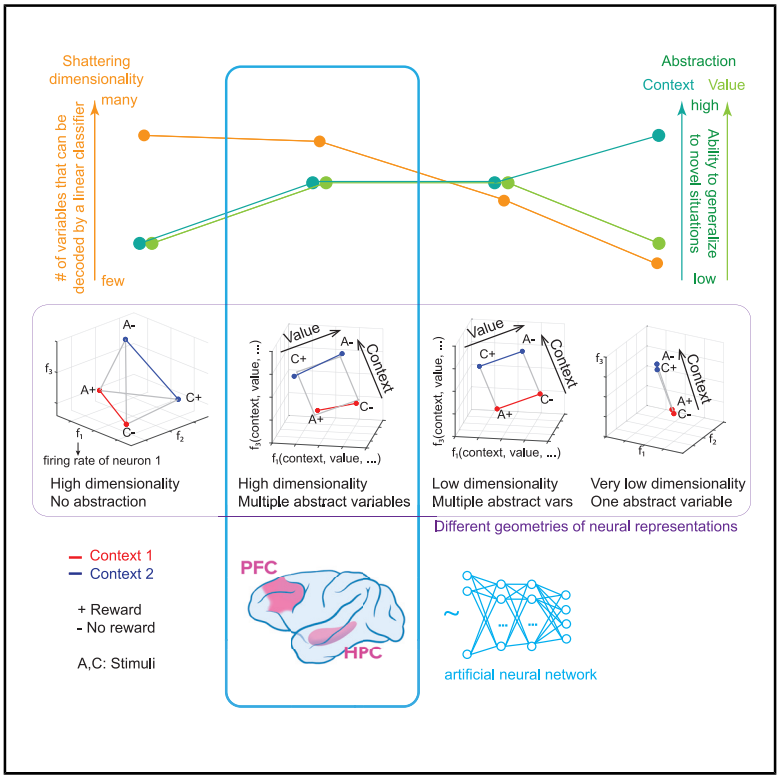
Cell

The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex

Graphical Abstract



Authors

Silvia Bernardi, Marcus K. Benna, Mattia Rigotti, Jérôme Munuera, Stefano Fusi, C. Daniel Salzman

Correspondence

sf2237@columbia.edu (S.F.),
cds2005@columbia.edu (C.D.S.)

In Brief

Different types of cognitive, emotional, and behavioral flexibility—generalization in novel situations and the ability to generate many different responses to complex patterns of inputs—place different demands on neural representations. This paper shows how the geometry of neural representations can be critical for elucidating how the brain supports these forms of flexible behavior.

Highlights

- The geometry of abstraction supports generalization
- Hippocampal and PFC representations are simultaneously abstract and high dimensional
- Multiple task-relevant variables are represented in an abstract format
- Representations in simulated neural networks are similar to recorded ones

Bernardi et al., 2020, Cell *183*, 1–14
November 12, 2020
https://doi.org/10.1016/j.cell.2020.09.031

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

Cell

CellPress

Article

The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex

Silvia Bernardi,[2,3,5,8,10] Marcus K. Benna,[1,4,5,9,10] Mattia Rigotti,[7,10] Jérôme Munuera,[1,10,12] Stefano Fusi,[1,4,5,6,11,*] and C. Daniel Salzman[1,2,5,6,8,11,13,*]

[1]Department of Neuroscience, Columbia University, New York, NY, USA
[2]Department of Psychiatry, Columbia University, New York, NY, USA
[3]Research Foundation for Mental Hygiene, Menands, NY, USA
[4]Center for Theoretical Neuroscience, Columbia University, New York, NY, USA
[5]Mortimer B. Zuckerman Mind Brain Behavior Institute, Columbia University, New York, NY, USA
[6]Kavli Institute for Brain Sciences, Columbia University, New York, NY, USA
[7]IBM Research AI, Yorktown Heights, NY, USA
[8]New York State Psychiatric Institute, New York, NY, USA
[9]Neurobiology Section, Division of Biological Sciences, University of California, San Diego, La Jolla, CA, USA
[10]These authors contributed equally
[11]Senior author
[12]Present address: Sorbonne Université, Institut du Cerveau - Paris Brain Institute - ICM, Inserm, CNRS, APHP, Paris, France
[13]Lead Contact
*Correspondence: sf2237@columbia.edu (S.F.), cds2005@columbia.edu (C.D.S.)
https://doi.org/10.1016/j.cell.2020.09.031

SUMMARY

The curse of dimensionality plagues models of reinforcement learning and decision making. The process of abstraction solves this by constructing variables describing features shared by different instances, reducing dimensionality and enabling generalization in novel situations. Here, we characterized neural representations in monkeys performing a task described by different hidden and explicit variables. Abstraction was defined operationally using the generalization performance of neural decoders across task conditions not used for training, which requires a particular geometry of neural representations. Neural ensembles in prefrontal cortex, hippocampus, and simulated neural networks simultaneously represented multiple variables in a geometry reflecting abstraction but that still allowed a linear classifier to decode a large number of other variables (high shattering dimensionality). Furthermore, this geometry changed in relation to task events and performance. These findings elucidate how the brain and artificial systems represent variables in an abstract format while preserving the advantages conferred by high shattering dimensionality.

INTRODUCTION

When encountering a new situation, the ability to determine right away what to do is a hallmark example of cognitive flexibility. This ability relies on the fact that the world is structured and new situations often share features with previously experienced ones. These shared features provide a compact representation that uses a small number of variables to describe the environment. This representation can be constructed using a process of dimensionality reduction, which obviates the need to observe all possible combinations of values of all features appearing in the environment, overcoming the ''curse of dimensionality.'' The variables describing features shared by multiple instances can be represented in an ''abstract'' format in the brain, a format that can enable generalization in novel situations.

An account of how the brain may represent these variables has remained elusive. Motivated by the fact that the process of abstraction enables generalization, we developed analytic methods for determining when the geometry of neural representations encodes variables in an abstract format. We operationally defined a neural representation of a variable as being in an abstract format (an ''abstract variable'') when a linear neural decoder trained to report the value of the variable can generalize to situations not experienced by the decoder during training. Previously unseen combinations of the values of other variables describe these situations. In experiments, these situations correspond to task conditions not used for training the linear decoder. We call the performance of this decoder ''cross-condition generalization performance'' (CCGP), because it reflects the ability of a decoder to generalize to task conditions not used for training. CCGP is distinct from the type of generalization normally referred to when a variable is decoded by training on some samples from all experimental conditions and testing on held-out samples from the same types of conditions. Techniques similar to CCGP have previously been employed to examine the representation of one variable at a time and/or to identify a common neural substrate

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

that might underlie 2 or more cognitive operations (Horikawa et al., 2013; Zabicki et al., 2017; Parkinson et al., 2014; King and Dehaene, 2014; Saez et al., 2015; Munuera et al., 2018; Isik et al., 2014, 2018).

Representations of variables in an abstract format, as identified by CCGP, are typically low dimensional (their dimensionality equals the number of encoded variables). The machine learning community often refers to them as "disentangled," or factorized (e.g., Higgins et al., 2017). A disentangled representation can encode multiple variables in an abstract format simultaneously. However, since a perfectly factorized representation is low dimensional, it places severe limits on the number of different potential responses that a simple linear readout can generate (Rigotti et al., 2013; Fusi et al., 2016). To quantify this limitation, we used another measure that characterizes the geometry of representations, the shattering dimensionality (SD) (see also Rigotti et al., 2013). The SD is the number of different ways that points corresponding to the firing rates of one or more neurons in different (experimental) conditions can be separated (shattered) by a linear decoder. Typically when SD increases, CCGP decreases. However, this trade-off between CCGP and SD is not necessary, and there are geometries that allow for a surprisingly good compromise in which both SD and CCGP are high. These geometries enable good generalization in novel situations and retain properties that confer the flexibility to respond in many different ways to complex combinations of inputs.

We used CCGP and SD to examine the geometry of neural representations recorded from monkeys as they performed a serial reversal-learning task in which they switch back and forth between 2 un-cued contexts. A distinct stimulus-response outcome (SRO) mapping described each trial, and sets of SRO mappings defined contexts ("task sets"). Some variables were observable (related to sensory input or motor output), while context, a hidden variable, was defined by the temporal statistics of events and could not be directly inferred by the value of any observable variable. Therefore, if a neural ensemble represents the variable context in an abstract format, it would reflect a process of dimensionality reduction (i.e., abstraction) that consistently captures the relational properties of the states of the external world across time (Eichenbaum, 2017; Behrens et al., 2018; Recanatesi et al., 2019; Whittington et al., 2019; Benna and Fusi, 2019).

Neurophysiological recordings were targeted to the hippocampus (HPC) and two parts of the pre-frontal cortex (PFC), the dorsolateral pre-frontal cortex (DLPFC), and anterior cingulate cortex (ACC). The HPC has long been implicated in generating episodic associative memories that could play a central role in creating and maintaining representations of variables in an abstract format (Milner et al., 1998; Eichenbaum, 2004; Wirth et al., 2003; Schapiro et al., 2016; Kumaran et al., 2009). Neurons in ACC and DLPFC have been shown to encode rules and other cognitive information (Wallis et al., 2001; Miller et al., 2003; Buckley et al., 2009; Antzoulatos and Miller, 2011; Wutz et al., 2018; Saez et al., 2015), but testing whether a neural ensemble of single units represents one or more variables in a format that can support high CCGP has generally not been examined (but see Saez et al., 2015). Our data reveal that neural ensembles in all 3 brain areas, and ensembles of units in simulated multi-layer networks, simultaneously represent hidden and explicit variables in an abstract format, as defined by high CCGP, yet also possess high SD. Our results highlight the importance of characterizing the geometry of a neural representation—not just what information is represented—in order to understand a brain region's potential contribution to different types of flexible cognitive behavior.

RESULTS

We first present behavioral data and the theoretical framework and analytic methodology developed to characterize geometry. Then we characterize the geometry of neural representations in the HPC, DLPFC, and ACC and in simulated neural networks.

Monkeys Use Inference to Adjust Behavior

Monkeys performed a serial-reversal learning task in which each of 2 blocks of trials contained 4 types of trials (conditions). Three variables described each condition: a stimulus and its operant and reinforcement contingencies (SRO mapping). Un-cued switches occurred between the blocks of trials in which SRO mappings changed simultaneously for all 4 conditions. Thus, each block was a context defined by its set of 4 SRO mappings ("task sets"), a hidden variable.

Correct performance for 2 stimuli in each context required releasing a button after stimulus disappearance; for the other 2 stimuli, the correct operant response was to continue to hold the button (Figures 1A and 1B; see STAR Methods). For 2 stimuli, correct performance resulted in reward delivery; for the other 2 stimuli, correct performance did not result in reward, but it avoided a timeout and repetition of the same unrewarded trial (Figure 1B). Without warning, randomly after 50–70 trials, context switched, and many switches happened within an experiment.

Not surprisingly since context switches were un-cued, monkeys' performance dropped to significantly below chance immediately after a context switch (image 1 in Figure 1C). In principle, after this incorrect choice, monkeys could re-learn the correct SR associations for each image independently. Instead, behavioral evidence shows that monkeys perform inference in relation to context switches. After experiencing changed contingencies for one or more stimuli, average performance is significantly above chance for the stimulus conditions not yet experienced (image numbers 2–4, Figure 1C). Once monkeys exhibit evidence of inference, performance remains asymptotic for the remainder of trials in that context ($\sim$ 90% correct, Figure 1D). Note that the temporal statistics of events (trials) define the variable context. The only feature that trials of context 1 have in common is that they are frequently followed or preceded by other trials of context 1; the same applies to context 2 trials. Monkeys' behavior suggests that they exploit knowledge of these temporal statistics.

The Geometry of Neural Representations that Encode Abstract Variables

A neural ensemble can represent variables in many different ways. We now consider different representations that have distinct generalization properties. We use these properties to define when a variable is represented in an abstract format.

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

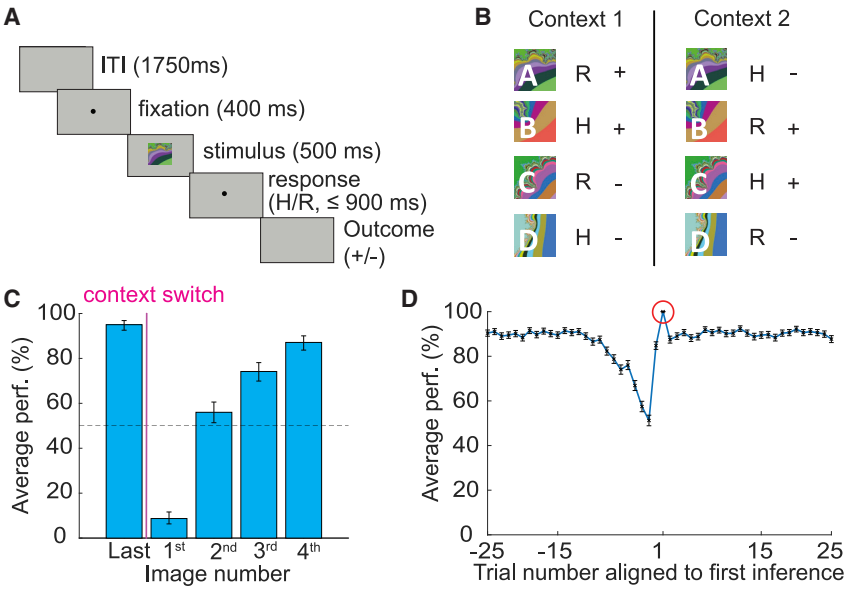


Figure 1. Task and Behavior

(A) Sequence of events within a trial. A monkey holds down a button and then fixates and views one of 4 familiar fractal images. A delay interval ensues, during which the operant response must be indicated (release or continue to hold the button, H and R). After a trace period, a liquid reward is delivered for correct responses for 2 of the 4 stimuli. Correct responses to the other 2 stimuli result in no reward but avoids a timeout and trial repetition.

(B) Task scheme, SRO mappings for conditions in the 2 contexts. (A)–(D), stimuli. +/−, reward/no reward for correct choices. Operant and reinforcement contingencies are orthogonal. After 50–70 trials in one context, context switches; experiments contain many context switches.

(C) Monkeys utilize inference to adjust behavior. Average percentage correct plotted for the first presentation of the last image appearing before a context switch ("Last") and for the first instance of each image after a context switch (1–4). For image numbers 2–4, monkeys performed at above chance despite not having experienced these trials in the current context (inference). Binomial parameter estimate, bars are 95% Clopper-Pearson confidence intervals

(D) Average percentage correct performance plotted versus trial number aligned on the first correct trial where the monkey used inference (red circle, defined as the first correct trial among the first presentations of the 2nd, 3rd, or 4th image type appearing after a context switch). So if image 1 is the first image after a context switch, and the first presentation of image 2 is performed correctly, it is the first correct inference trial. If it is performed incorrectly, the first correct inference trial could occur on the first presentation of image 3 or 4. Error bars, SEM.

Consider the hidden variable context. Some representations encode context but do not reflect a process of abstraction. For example, assume that the average firing rate of each neuron is random for each experimental condition (i.e., for each SRO mapping). Figure 2A depicts in the firing rate space an example of this representation. Each coordinate axis is the firing rate of one neuron. Each point in Figure 2A represents a vector containing the average activity of 3 neurons for each condition within a specific time window. The geometry of the representation is defined by the arrangement of all the points corresponding to the different experimental conditions.

In the random case under consideration, if the number of neurons is sufficiently large, the pattern of activity corresponding to each condition will be unique. If trial-by-trial variability in firing rate (i.e., noise) is not too large, even a simple linear classifier can decode context, as the 2 points of one context can be separated from the 2 points of the other context (Figure 2A). Notice that in this geometry any arbitrarily selected 2 points can be separated from the others. Each way of grouping the points (i.e., each dichotomy) corresponds to a different variable, and when 2 groups of points are linearly separable the corresponding variable is decodable. One way to characterize the geometry of representations is to determine how many dichotomies can be decoded by a linear classifier. We call this quantity shattering dimensionality (SD). SD is defined as the performance of a linear decoder averaged over all possible balanced dichotomies. A high SD means that a linear readout can generate a large number of input-output functions. SD is similar to the measure of dimensionality used in Rigotti et al. (2013). The representation in Figure 2A has maximal SD, as all dichotomies can be decoded.

The random representations just described allow for a form of generalization, as a decoder trained on a subset of trials from all conditions can generalize to held-out trials. This form of generalization is clearly insufficient to characterize a representation of a variable that is in an abstract format because the representations are random, and hence they do not reflect the links between the different instances (SRO mappings) of the contexts. Therefore, despite encoding context, this type of representation cannot be considered to represent context in an abstract format.

A neural representation of context in an abstract format needs to incorporate into the geometry information about the links between different instances of the same context. One way to accomplish this is to cluster together patterns of activity that correspond to conditions in the same context (see Figure 2B). Clustering is a geometric arrangement that permits an important and distinct form of generalization that we use to define when a neural ensemble represents a variable in an abstract format. We propose that this format can support a fundamental aspect of cognitive flexibility, the ability to generalize to novel situations. To identify in our experiments when a variable is represented in a format that could support this type of generalization, we determined whether one could decode a variable in experimental conditions not used for training the decoder. This type of generalization is illustrated in Figure 2B. A linear decoder is trained to decode context on the rewarded conditions of the 2 contexts. Then the decoder is tested on trials not rewarded. The clustered geometric arrangements of the points ensures that the decoder successfully generalizes to the conditions not used for training.

In marked contrast to clustered geometric arrangements, random responses to trial conditions do not allow for this form of generalization. In the case of random responses (e.g.,

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

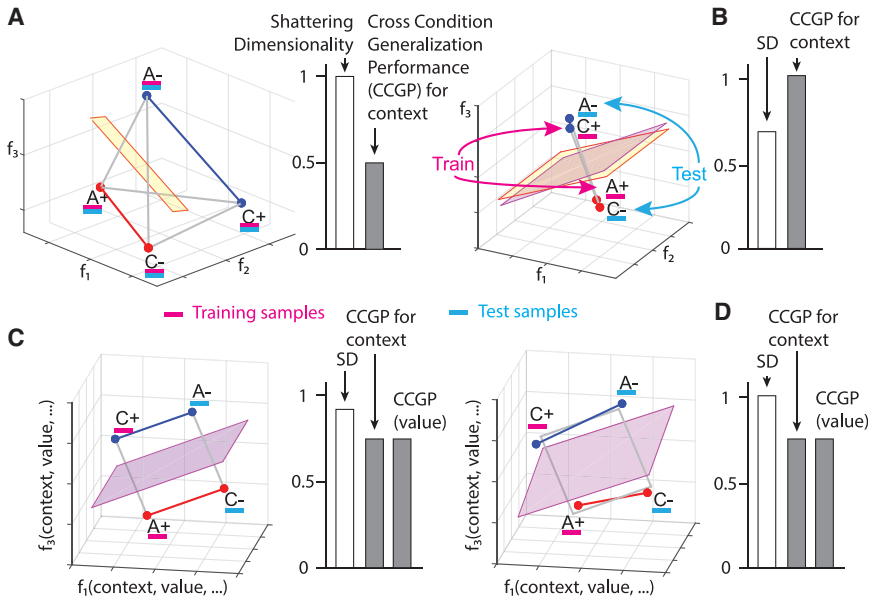


Figure 2. The Geometry of Abstraction

Different representations of context can have distinct geometries, each with different generalization properties. Each panel depicts in the firing rate space points that represent the average firing rate of 3 neurons in only 4 of the 8 conditions from experiments. The 4 conditions are labeled according to stimulus identity (A and C) and reward value (+,–).

(A) A random representation (points are at random locations in the firing rate space), which allows for decoding of context. The yellow plane represents a linear decoder that separates the 2 points of context 1 (red) from the 2 points of context 2 (blue). The decoder is trained on a subset of trials from all conditions (purple) and tested on held-out trials from the same conditions (cyan). All other variables corresponding to different dichotomies of the 4 points can also be decoded using a linear classifier; hence, the shattering dimensionality (SD) is maximal, but CCGP is at chance (right histogram).

(B) Abstraction by clustering: points are clustered according to context. A linear classifier is trained to discriminate context on rewarded conditions (purple). Its generalization performance (CCGP) is tested on unrewarded conditions not used for training (cyan). The separating plane when trained on rewarded conditions (purple) is different from the one obtained when all conditions are used for training (yellow), but, for this clustered geometry, both planes are very similar. With clustered geometry, CCGP is maximal for context, but context is also the only variable encoded. Hence, SD is close to chance (right histogram) (See Methods S2). Notice that the form of generalization CCGP involves is different from traditional decoding generalization to held out trials (see Methods S3).

(C) Multiple abstract variables: factorized/disentangled representations. The 4 points are arranged on a square. Context is encoded along the direction parallel to the two colored segments, and value is in the orthogonal direction. In this arrangement, CCGP for both context and value are high; the SD is high but not maximal because the combinations of points that correspond to an exclusive OR (XOR) are not separable. Individual neurons exhibit linear mixed selectivity (see Methods S6).

(D) Distorted square: a sufficiently large perturbation of the points makes the representation higher dimensional (the 4 points no longer lie on a plane); a linear decoder can now separate all possible dichotomies, leading to maximal SD, but at the same time CCGP remains high for both value and context.

See Methods S5 and Figure S2 that constructs geometries that have high SD and CCGP at the same time. See also Figure S1A for a detailed description of how CCGP is computed.

Figure 2A), if only rewarded conditions are used to train the decoder to classify context, then the separating plane will be very different from the one where the decoder is trained on a subset of trials from all conditions. Now the 2 test points corresponding to unrewarded conditions will have the same probability of being on either side of the separating plane derived from the decoder trained on rewarded conditions only. We designate the performance of decoders in classifying variables when testing and training on different types of conditions as CCGP (see also Saez et al., 2015). We use the average CCGP across all possible ways of choosing training and testing conditions as a metric of the degree to which a variable is represented in an abstract format (see also Figure S1). A variable is defined as being represented in an abstract format when CCGP is significantly different from one in which the points from the same conditions would be at random locations (see STAR Methods).

The Geometry of Multiple Abstract Variables

The clustering geometry allows a single variable to be encoded in an abstract format. In the case in which the differences within each cluster are only due to noise, the single variable encoded in an abstract format is the only decodable variable, and SD would be low. How can neural ensembles represent multiple variables in an abstract format at the same time? One way would be if different neurons exhibit pure selectivity for different variables (e.g., one neuron specialized to encode context, and another specialized to encode value). This factorized or disentangled geometry allows for high CCGP for both context and value. However, in our dataset neurons that respond to a single variable are rarely observed, a finding consistent with many studies showing that neurons more commonly exhibit mixed selectivity for multiple variables (Rigotti et al., 2013; Fusi et al., 2016) (see Methods S7). Nonetheless, the generalization properties of factorized representations are preserved when all the points are rotated in the firing rate space, as in Figure 2C. Here, the 4 data points lie on the corners of a square, with neurons exhibiting linear mixed selectivity (Rigotti et al., 2013). Under the assumption that a decoder is linear, CCGP will not change if a linear operation like rotation is performed on the data points. Using a similar construction, it is therefore possible to represent as many variables in an abstract format as the number of neurons.

For the representation depicted in Figure 2C, SD is high but not maximal because the combinations of points that correspond to an exclusive OR (XOR) are not separable (i.e., a linear classifier cannot separate the visual stimuli A and C). Although in this simple example SD is still relatively high, it can decrease exponentially with the total number of conditions. However, a sufficiently large distortion of the representation in Figure 2C can lead to a

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

representation that allows for maximal SD and, at the same time, high CCGP for both context and value (Figure 2D). Simulations show that there is a surprisingly wide range of distortions in which the representations can have both high SD and high CCGP for multiple variables (see Figure S2).

Measuring Abstraction

The serial reversal-learning task contains 8 types of trials (SRO combinations). There exist 35 different ways of dividing the 8 types of trials into 2 groups of 4 conditions (i.e., 35 dichotomies). Each dichotomy corresponds to a variable that could be in an abstract format. Three of these variables describe the context, reward value, and correct action associated with each stimulus in each of the 2 contexts. We took an unbiased approach to determine which dichotomies are decodable and which are in an abstract format. To assess which variables were in an abstract format and to further characterize the geometry of the recorded neural representations, we used two quantitative measures, CCGP and the parallelism score (PS).

As described in reference to Figures 2B–2D, CCGP can be computed by training a linear decoder to classify any dichotomy on a subset of conditions, and testing classification performance on conditions not used for training. Since there are multiple ways of choosing the subset of conditions used for training, we report the average CCGP across all possible ways of choosing the training and testing conditions (see STAR Methods and Figure S1A). The PS is related to CCGP, but it focuses on specific aspects of the geometry. In particular, the PS quantifies the degree to which coding directions are parallel when training a decoder to classify a variable for different sets of conditions. Consider the case depicted in Figures S1B and S1C. Two different lines (which would be hyperplanes in a higher dimensional plot) are obtained when a decoder is trained to classify context using the two points on the left (rewarded conditions, magenta) or the two points on the right (unrewarded conditions, dark purple). The two lines representing the hyperplanes are almost parallel, indicating that this geometry will allow for high CCGP. The extent to which these lines (hyperplanes) are aligned can be quantified by calculating the coding directions (arrows in Figures S1B and S1C) that are orthogonal to the lines. The PS is the degree to which these coding vectors are parallel. Analogous to CCGP, there are multiple ways of pairing points that correspond to two different values of a dichotomy. We compute the PS for all ways of pairing and report the maximum observed PS. For random representations, which do not represent variables in an abstract format, the representations of individual conditions will be approximately orthogonal if the number of neurons is large. In this case, however, the coding directions will be randomly oriented, and hence orthogonal to each other. Therefore, small PS values would be observed in the case of random representations. PS can be used an an alternative measure of abstraction: a variable would be abstract when its PS is significantly larger than the PS of a random representation. High PS usually predicts high CCGP (see STAR Methods).

HPC, DLPFC, and ACC Represent Variables in an Abstract Format

We recorded the activity of 1,378 individual neurons in two monkeys while they performed the serial reversal learning task. Of these, 629 cells were recorded in HPC (407 and 222 from each of the 2 monkeys, respectively), 335 cells were recorded in ACC (238 and 97 from each of the 2 monkeys), and 414 cells were recorded in DLPFC (226 and 188 from the 2 monkeys). Our initial analysis of neural data focused on the time epoch immediately preceding a response to a stimulus. If monkeys employ a strategy in which context information is mixed with stimulus identity information to form a decision, then context information could be stored right before responses to stimuli begin. Furthermore, information about recently received rewards and performed actions may also be present during this interval, as knowing whether the last trial was performed correctly is useful (see Discussion).

In a 900 ms time epoch ending 100 ms after stimulus onset on the current trial (visual response latencies are greater than 100 ms in the recorded brain areas), individual neurons in all 3 brain areas exhibited mixed selectivity with respect to the task conditions on the prior trial, with diverse patterns of responses observed (see Figure S3A). Information about the current trial is not yet available during this time epoch. We then determined which variables (among the 35 dichotomies) were represented in each brain area and which variables were in an abstract format. The traditional application of a linear neural decoder revealed that most of the 35 variables could be decoded from neural ensembles in all brain areas, including the context, value, and action of the previous trial (Figure 3A). However, very few variables were represented in an abstract format at above chance levels, as quantified by CCGP. Variables with the highest CCGP were those corresponding to context and value in all 3 brain areas, as well as action in DLPFC and ACC. Action was not in an abstract format in HPC despite being decodable using traditional methods. The representation of variables in an abstract format did not preferentially rely on the contribution of neurons with selectivity for only one variable, indicating that neurons with mixed selectivity for multiple variables likely play a key role in generating representations of variables in abstract format (see Methods S7). Consistent with the CCGP analyses, the highest PSs observed in DLPFC and ACC corresponded to the 3 variables for context, value, and action. In HPC, the two highest PSs were for context and value, with action having a PS not significantly different than chance. The geometric architecture revealed by CCGP and the PS can be visualized by projecting the data into a 3D space using multidimensional scaling (MDS) (see Figure S4A). Figures 3C and 3D show how the geometry that represents the variables context, value, and action evolves over time prior to a visual response; the degree to which context is represented in an abstract format increases during this time interval.

Notice that the hidden variable context was represented in an abstract format in all 3 brain areas just before neural responses to stimuli on the current trial occur. An analysis that only assesses whether the geometry in the firing rate space resembles clustering, similar to what has been proposed in Schapiro et al. (2016), would lead to the incorrect conclusion that context is strongly abstract only in HPC (see Figures S4C–S4E). The representation of context in an abstract format in ACC and DLPFC relies on a geometry revealed both by CCGP and the PS but

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

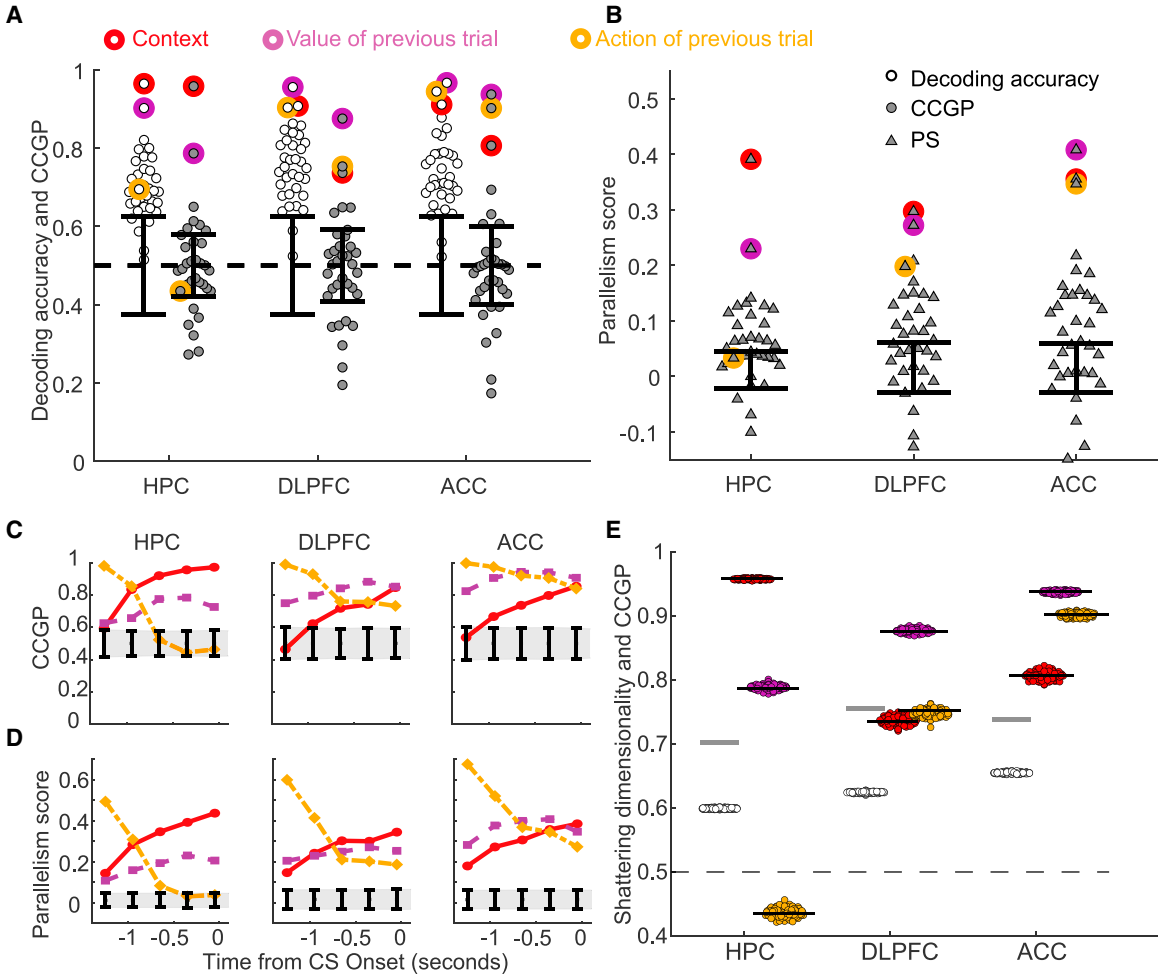


Figure 3. Decoding Accuracy, CCGP, and Parallelism Score (PS) in the Three Recorded Brain Areas

(A and B) CCGP, decoding accuracy, and PS for the variables that correspond to all 35 dichotomies shown separately for each brain area in a 900 ms time epoch beginning 800 ms before image presentation. The points corresponding to the context, value, and action of the previous trial are highlighted with circles of different colors. Table S2 contains the values of CCGP and PS for all dichotomies. Context and value are represented in an abstract format in all 3 brain areas, but action is abstract only in PFC (although it can be decoded in HPC (see also Figure S4 to visualize the arrangement of the points in the firing rate space). Almost all dichotomies can be accurately decoded, and the SD is high: HPC, 0.70; DLPFC, 0.75; ACC, 0.74 (see also Methods S4 which describes other measures of dimensionality). Error bars are ±2 standard deviations around chance level as obtained from a geometric random model (CCGP) or from a shuffle of the data (decoding accuracy and PS). Results were qualitatively similar in the 2 monkeys (see Figure S5).

(C and D) CCGP (C) and the PS (D) plotted as a function of time for the variables context, action, and value in the 3 brain areas (data points plotted in the center of a 900 ms window, which is stepped in 300 ms increments). Error bars, same as in (A and B).

(E) Measured SD in each brain area is significantly greater than the SD of a perfectly factorized representation. To determine whether a perfectly factorized representation, which typically has high CCGP, is consistent with the high SD observed in experimental data, a perfectly factorized null model is constructed by placing the centroids of the noise clouds that represent the 8 different experimental conditions at the vertices of a cuboid. The lengths of the sides of the cuboid are tuned to reproduce (on average) the CCGP values observed in the experiment for the variables context, value, and action. From this artificially generated data corresponding to a perfectly factorized model, SD and CCGP are calculated, with the procedure repeated 100 times for each brain area. SD (empty circles) and CCGP for the variables context, value, and action (colored circles) plotted for each realization of the random model. Gray horizontal lines, SD from the experiments; black horizontal lines, CCGP for context, value, and action, mimicking experimental data shown in (A). The factorized models re-capitulate the recorded CCGP values, but the SD values measured in all 3 brain areas are significantly higher than in any realization of the factorized model. The difference between the experimentally measured SD and the average SD of the factorized model is more than an order of magnitude larger than the standard deviation of the model SD distribution in all cases, indicating that the experimental data are not consistent with such a factorized geometry.

See also Figures S4 and S5, Table S2, and Videos S1, S2, and S3.

missed if only considering clustering. The data also show that CCGP and the PS identify when multiple variables are represented in an abstract format simultaneously.

We next wondered whether the observed geometry is consistent with a perfectly factorized representation. In such a representation multiple variables could exhibit high CCGP, but the

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

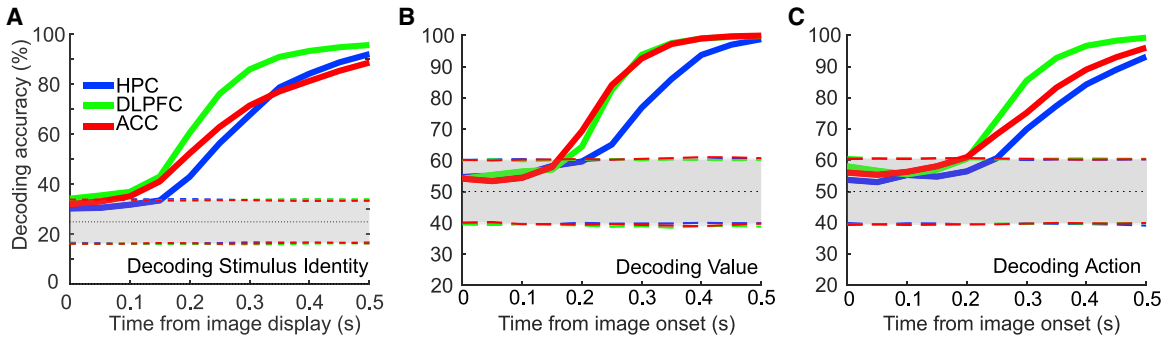


Figure 4. Decoding Accuracy for Stimulus, Value, and Action

Decoding accuracy for stimulus identity, value, and action plotted as a function of time in the 3 brain areas. Decoding of stimulus identity employs a 4-way classifier, so chance is 0.25. A linear decoder was employed because neural responses are highly heterogeneous, exhibiting mixed selectivity (see Figure S3A) and are rarely specialized (see Figure S6). Dotted line, chance. Shaded areas, two-sided 95%-confidence intervals calculated with a permutation test (randomly shuffling trials, 1,000 repetitions). See Figure S3 for decoding of task-relevant variables across a longer timescale). See also Figure S6.

SD would be relatively low. However, as shown in Figure 3A, nearly all dichotomies could be decoded in each brain area, and thus SD was greater than 0.7 in every case. These SD values are significantly greater than the SD expected in the case of a perfectly factorized representation tuned to replicate the observed CCGP values for context, action, and value (Figure 3E). Thus, the geometry of recorded representations has properties inconsistent with a perfectly factorized representation.

The Dynamics of the Geometry of Neural Representations during Task Performance

The different task events in the serial-reversal learning task engage a series of cognitive operations, including perception of the visual stimulus, formation of a decision about whether to release the button, and reward expectation. These task events modulated the geometry of the neural representations. Shortly after stimulus appearance, decoding performance for stimulus identity, expected reinforcement outcome, and the to-be-performed action on the current trial rises rapidly from chance levels to asymptotic levels in all 3 brain areas (Figure 4). The very short temporal gap between the rise in decoding for stimulus identity and the rises in expected outcome and operant action suggests a rapid decision process upon stimulus appearance. Decoding performance for value and action rises the most slowly in HPC, suggesting that the signals reflecting decisions are not first represented there.

We next analyzed the geometry of neural representations during the time interval in which the planned action and expected trial outcome first become decodable, focusing on a 900 ms window beginning 100 ms after stimulus onset. We again took an unbiased approach and considered all 35 dichotomies. Nearly all dichotomies were decodable using a traditional decoding approach (Figure 5A). The SD was accordingly high in all 3 brain areas (>0.88) and significantly larger than the SD computed for a factorized representation (Figure 5E). Despite the high SD, multiple variables were simultaneously represented in an abstract format (Figures 5A and 5B). Strikingly, context was not represented in an abstract format in DLPFC, despite being decodable well above chance levels; in ACC, CCGP indicated that context was only very weakly abstract. These data demonstrate that high decoding performance using traditional cross-validated decoding does not necessarily predict CCGP above chance. For example, decoding performance is ~0.9 for context in DLPFC in the 900 ms window beginning 100 ms after image presentation, and for context in ACC prior in the interval ending 100 ms after image presentation. Yet, in DLPFC, CCGP is at chance (Figure 5A), but, in ACC during the earlier time interval, CCGP is ~0.8, well above chance (Figure 3A). Of course, the converse is not true, and high CCGP is always accompanied by high decoding performance.

CCGP for context is not significantly different from chance in DLPFC and ACC for a sustained period of time after image presentation. During this time, value and action are represented in an abstract format in all 3 brain areas (Figures 5C and 5D). Recall, however, that the geometry of the representation of context in DLPFC and ACC evolves prior to the presentation of the stimulus on the next trial, as context is in an abstract format in DLPFC and ACC during this time interval (Figures 3A and 3B). In HPC, context was maintained more strongly in an abstract format after stimulus appearance, as well as prior to stimulus appearance on the next trial. Overall, the CCGP results were largely correlated with the PS. Together these findings indicate that task events both engage a series of different cognitive operations to support performance and modulate the geometry of neural representations. The analytic approach reveals a fundamental difference in how the hidden variable context is represented in HPC compared to PFC brain areas during and after decision making on this task.

Correlation between Level of Abstraction (CCGP) and Behavior

We next sought evidence that the geometry of neural representations was related to behavioral performance of the task. We focused on the representation of the variable context during the 900 ms time interval beginning 800 ms before image onset. The maintenance of information about context during this time interval is potentially useful, because it may be utilized once a neural response to a stimulus occurs. Decisions may be made by combining information about context and stimulus identity. We note that the analysis of the geometry of representations in

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

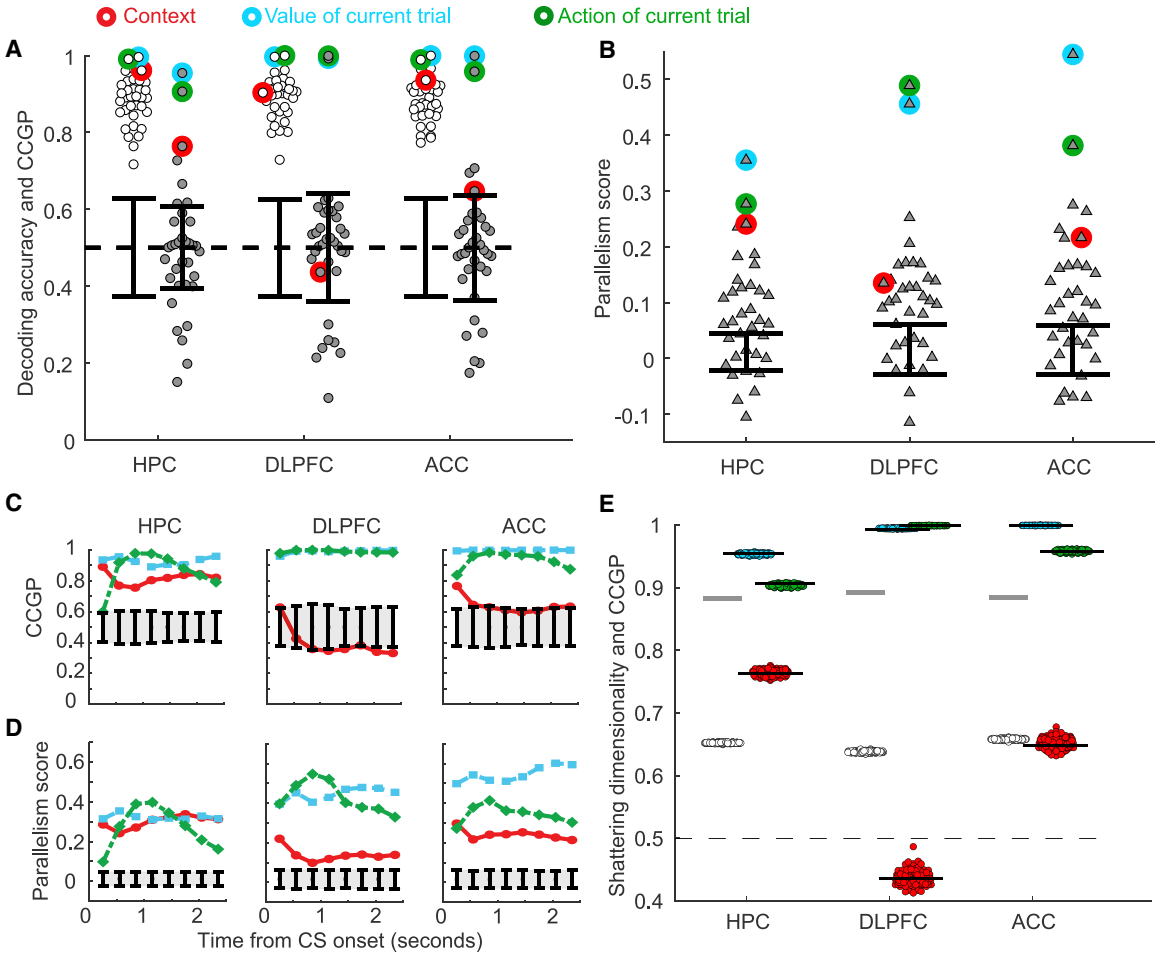


Figure 5. Decoding Accuracy, CCGP, and the PS after Stimulus Onset in the Three Brain Areas

(A and B) CCGP, decoding accuracy, and PS for all 35 dichotomies in the time interval from 100 to 1,000 ms after stimulus onset. See Table S2 for the values of CCGP and PS for all dichotomies. Error bars, ±2 standard deviations around chance as obtained from a geometric random model (CCGP) or from a shuffle of the data (decoding accuracy and PS). The SD is higher in this interval than in the earlier time epoch: HPC 0.88, DLPFC 0.89, and ACC 0.88. Results were qualitatively similar in the two monkeys (see Figures S5).

(C and D) CCGP (C) and the PS (D) plotted as a function of time for the variables context, action, and value in the 3 brain areas (data points plotted in the center of a 900 ms window, 300 ms steps). Error bars, same as in (A and B).

(E) The SD observed in each brain area is significantly greater than the SD of a perfectly factorized representation. The same analysis as in Figure 3E is shown but for this later time interval.

See also Figure S5 and Table S2.

relation to behavioral performance was not feasible in the time interval where the decision itself likely occurs. This is because the variables for the selected action and expected reinforcement become represented extremely rapidly after stimulus identity is decodable (Figure 4), and this time interval is too short for CCGP analysis to be possible.

In all 3 brain areas, there was a statistically significant decrease in CCGP for context on error compared to correct trials (Figure 6A). By contrast, using a traditional linear decoder, the decoding of context in all 3 brain areas was not significantly related to behavioral performance (Figure 6B). Context is a variable that monkeys could not have known about prior to their gaining experience on this particular task with these specific stimuli. Yet, a representation of the variable context within these brain areas conforms to a particular geometry, a geometry that must have been created *de novo* and that specifically relates to task performance.

Abstraction in Multi-layer Neural Networks Trained with Back-Propagation

We wondered whether neural representations observed in a neural network model trained with back-propagation have similar geometric features as those observed experimentally. We designed our simulations such that the 8 classes of inputs contained no structure reflecting a particular dichotomy. The network had to output two arbitrarily selected variables corresponding to two specific dichotomies. We hypothesized that forcing the network to output these two variables would break

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

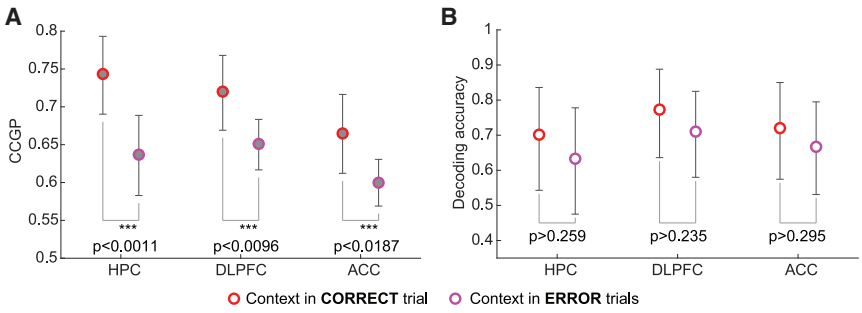


Figure 6. The Relationship between CCGP for Context and Behavioral Performance

(A) CCGP for context, measured in the 900 ms time interval ending 100 ms after stimulus onset, is significantly lower on error trials than on correct trials in all 3 brain areas. Average decreases (±1 standard deviation) in CCGP on error trials are: 0.107 ± 0.038 ($p<0.0011$) in HPC, 0.0691 ± 0.0303 ($p<0.0096$) in DLPFC, and 0.0651 ± 0.0309 ($p<0.0187$) in ACC (average, standard deviation and p values computed over 10,000 repetitions of bootstrap re-sampling trials using a sub-population of 180 neurons per area; error bars, 95th percentiles of the bootstrap distributions). Since errors occurred in a relatively small fraction of all trials, neurons were selected according to a different criterion than other analyses, resulting in fewer neurons being included in this analysis (see STAR Methods). Results in this figure and Figure 3 are thus not directly comparable.

(B) Decoding accuracy for context is not significantly different for correct and error trials. Average drops (±1 standard deviation) of decoding accuracy between correct and error trials: 0.069 ± 0.108 ($p>0.259$) in HPC, 0.0627 ± 0.0903 ($p>0.235$) in DLPFC, and 0.0532 ± 0.0984 ($p>0.295$) in ACC (averages, standard deviations, p values, and error bars obtained analogously as in A on the same neurons).

See STAR Methods and Table S1 for details.

the symmetry between all dichotomies, leading to the creation of representations of the output variables in an abstract format, as defined by CCGP. Other variables (other dichotomies) would not be expected to be in an abstract format. If our hypothesis is confirmed, it would demonstrate a way of generating representations of selected variables in an abstract format, which in turn could be used to benchmark our analytic methods.

We trained a two layer network using back-propagation to read an input representing a handwritten digit between 1 and 8 (MNIST dataset) and to output whether the input digit is odd or even, and, at the same time, whether the input digit is large (>4) or small (≤4) (Figures 7A and 7B). Parity and magnitude are the two variables that we hypothesized could be represented in an abstract format. Training the network to perform this task resulted in changes in the geometry of the representations in each stage of processing, as revealed by 2-dimensional MDS plots of a subset of the images in the input space, and in the first and second layers (Figures 7E–7G). We tested whether the learning process led to high CCGP and PS for parity and magnitude in the last hidden layer of the network. If these variables are in an abstract format, then the abstraction process would be similar to the one studied in the experiment in the sense that it involves aggregating together inputs that are visually dissimilar (e.g., the digits "1" and "3", or "2" and "4"). Analogously, in the experiment very different sequences of events (different conditions defined by SRO mappings) are grouped together into what defines the contexts.

We computed both CCGP and the PS for all possible dichotomies of the 8 digits. Figures 7C and 7D show decoding accuracy, CCGP, and the PS for all dichotomies. The two largest CCGP and PS values are significantly different from those of the random model and correspond to the parity and the magnitude dichotomies. No other dichotomies have a statistically significant CCGP value, but all dichotomies can be decoded. CCGP and the PS therefore identify in the last hidden layer variables that correspond to the dichotomies encoded in the output. Note that the geometry of the representations in the last layer actually allow the network to perform classification of any dichotomy, as decoding accuracy is close to 1 for every dichotomy. Thus, SD is very close to 1 (0.96). Abstraction therefore is not necessary for the network to perform tasks that require outputs corresponding to any of the 35 dichotomies.

A neural network was then trained to perform a simulated version of our experimental task, and a similar geometry was observed as in experiments (see Methods S8). We used a reinforcement learning algorithm (Deep Q-Learning) to train the network. This technique uses a deep neural network representation of the state-action value function of an agent trained with a combination of temporal-difference learning and back-propagation refined and popularized by Mnih et al. (2015). As commonly observed, neural representations displayed significant variability across runs of the learning procedure. However, in a considerable fraction of runs, the neural representations during a modeled time interval preceding stimulus presentation recapitulate the main geometric features that we observed in the experiment. In particular, after learning, the hidden variable context is represented in an abstract format in the last layer, despite not being explicitly represented in the input, nor in the output. The representations also encode value and action in an abstract format, consistent with the observation that hidden and explicit variables are represented in an abstract format in the corresponding time interval in the experiment.

DISCUSSION

In this paper, we developed analytic approaches for characterizing the geometry of neural representations to understand how one or more variables may be represented in an abstract format simultaneously. Electrophysiological recordings of neural ensembles in DLPFC, ACC, and HPC reveal that all 3 areas represent multiple variables in an abstract format, as revealed by CCGP, while still retaining high SD. Artificial multi-layer networks trained with back-propagation exhibited a similar geometry. Thus, geometries exist in which variables are represented in an abstract format to support generalization in novel situations while retaining properties that enable a linear classifier to generate many different responses to complex combinations of inputs.

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

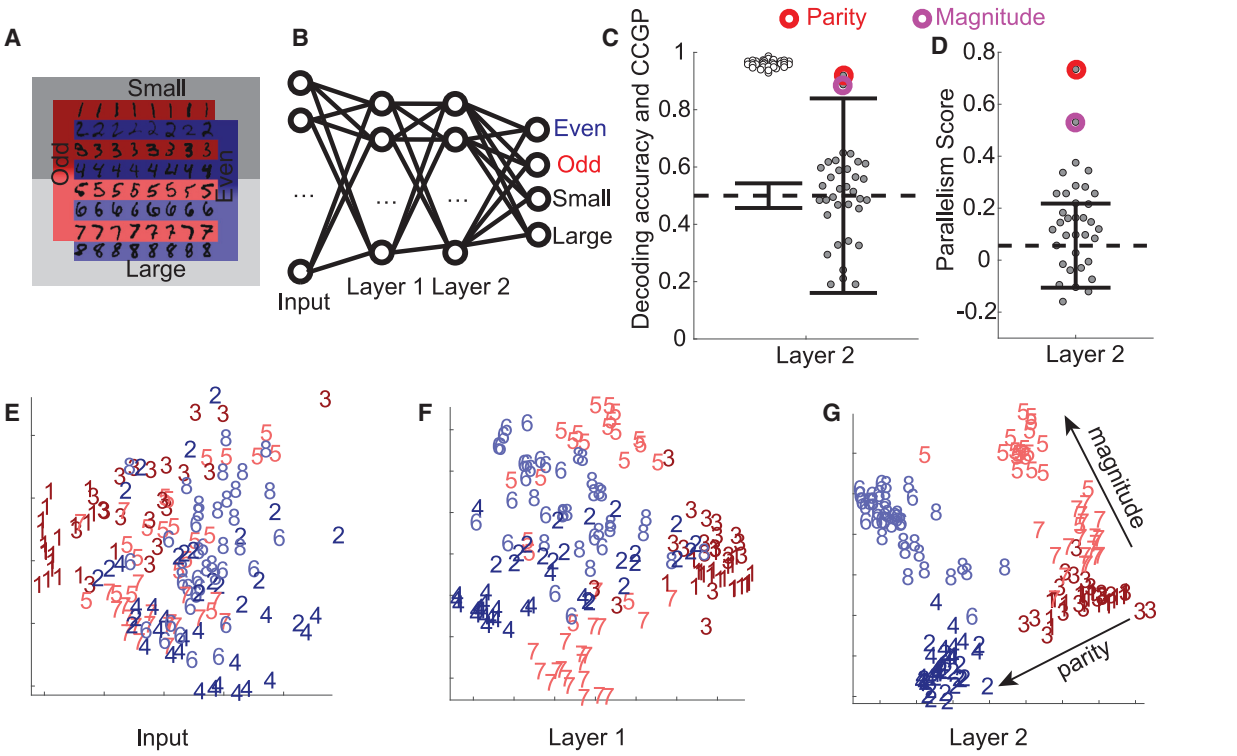


Figure 7. Simulations of a Multi-layer Neural Network Replicate Experimentally Observed Geometry

(A) Schematic of the two discrimination tasks using the MNIST dataset and color code for (E)–(G). The colors indicate parity, and shading indicates the magnitude of the digits (darker for smaller ones).

(B) Diagram of the network architecture. The input layer receives images of MNIST handwritten digits 1–8. The two hidden layers have 100 units each, and in the final layer there are 2 pairs of output units corresponding to 2 binary variables. The network is trained using back-propagation to simultaneously classify inputs according to whether they depict even/odd and large/small digits.

(C) CCGP and decoding accuracy for variables corresponding to all 35 balanced dichotomies when the second hidden layer is read out. Only the 2 dichotomies corresponding to parity and magnitude are significantly different from a geometric random model (chance level: 0.5; the two solid black lines indicate ±2 standard deviations). Decoding performance is high for all dichotomies and hence inadequate to identify the variables stored in an abstract format.

(D) Same as (C) but for the PS, with error bars (±2 standard deviations) obtained from a shuffle of the data. Both CCGP and the PS allow us to identify the output variables used to train the network.

(E–G) Two-dimensional MDS plots of the representations of a subset of images in the input (pixel) space (E), as well as in the first (F) and second hidden layers (G). In the input layer, there is no structure apart from the accidental similarities between the pixel images of certain digits (e.g., ones and sevens). In the first, and even more so in the second, layer, a clear separation between digits of different parities and magnitudes emerges in a geometry with consistent and approximately orthogonal coding directions for the two variables. See Methods S1 for more details.

For neural network simulations of the task performed by the monkeys, see Methods S8 and Figure S7 for a reinforcement learning model, and Figure S8 for a supervised learning model.

The Relationship between Neural Representations and Behavior

CCGP measures how well a decoder generalizes to conditions held out from training. Admittedly and by construction, these conditions need not be new to an experimental subject. In our experiments, all conditions indeed were experienced by monkeys repeatedly. Nevertheless, we assume that generalization across conditions "new to the decoder" is a good proxy for generalization across genuinely novel conditions. This assumption may not be valid in all situations, but it is reasonable to suppose that it holds if new conditions are ecologically and behaviorally not too dissimilar from familiar ones.

In our data, high CCGP for context indicates that the 4 conditions defining each context have been grouped together with a particular geometry that can enable generalization in novel situations. Consider a subject that can already perform the serial-reversal learning task. Suppose that for each already-learned context, a distinct novel contextual cue is presented in association with 3 of the conditions. If the geometry of the representation of context is sufficiently preserved during learning of these new associations, then it is likely that the remaining condition from each context will also be associated with its respective novel contextual cue. In this case, the subject can exhibit behavioral generalization the first time the remaining condition appears with a novel contextual cue. This is made possible by the fact that the links between conditions in the same context have already been learned and are reflected in the geometry of representations. In principle, one can use CCGP for all possible groupings (or variables) to predict whether behavioral generalization will be observed for an exponential number of novel

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

situations. Testing whether these predictions are correct will require new and very challenging studies that generate a sufficiently large number of novel situations during experimental sessions.

In our experiments, the degree to which context is represented in an abstract format decreases significantly in all 3 brain areas when monkeys make mistakes even though objectively novel conditions were not part of the design. Information about context could be useful to monkeys as they utilize inference to adjust behavior after experiencing at least one trial upon a context switch. However, context need not be represented in an abstract format to support inference. Inference could also result from a strategy that creates a large look-up table of all possible sequences of trials; here, no explicit knowledge of context is required to support inference. Nonetheless, the fact that the geometry of the representation of context relates to task performance suggests that at some stage of learning to perform the task, a geometry that confers generalization properties may have been created to support monkeys' strategy. One possibility is that this geometry is created early in the learning process, when stimuli are first being experienced. This possibility will need to be explored by recording from neurons during initial learning when stimuli are novel.

Information about the value and action of the previous trial are also represented in all 3 brain areas just before stimulus onset. When context switches, the reward received on the previous trial is the only feedback from the external world that indicates context has changed, suggesting that representing this information is beneficial. Consistent with this, simulations reveal that value becomes progressively more abstract as the frequency of context switches increases (see Figure S8B and Methods S8). In addition, monkeys occasionally make mistakes that are not due to a context change. To discriminate between these errors and those due to a context change, information about value is not sufficient and information about the previously performed action can help select the correct response on the next trial. Conceivably, the abstract representations of reward and action may also afford the animal more flexibility in learning and performing other tasks.

Abstraction and Linear Mixed Selectivity

Our analytic approach emphasizes the importance of studying the geometry of neural representations at the level of the patterns of activity of a neural ensemble. In principle, one could analyze single neurons to detect low-dimensional structure that relates to abstraction. In perfectly factorized representations, which represent variables in an abstract format as defined by CCGP, individual neurons exhibit either pure selectivity to a factor or linear mixed selectivity to 2 or more factors. Linear mixed selectivity neurons have been observed in previous work (see, e.g., Raposo et al., 2014; Parthasarathy et al., 2017; Chang and Tsao, 2017; Dang et al., 2020). However, examination only of single-neuron coding properties may fail to reveal important properties of a representation considered at the level of an ensemble. This is because, at the level of individual neurons, the linear component of responses dominates in many situations (Rigotti et al., 2010, 2013; Barak et al., 2013; Lindsay et al., 2017; Fusi et al., 2016), and it is easy to miss the fact that non-linear components of multiple neurons make representations high dimensional at the level of an ensemble. More importantly, an analysis of single neurons ignores the correlations between non-linear components across neurons; these correlations can strongly affect the generalization properties of representations.

Dimensionality and Abstraction in Neural Representations

Dimensionality reduction is widely employed in machine learning applications and data analyses because it leads to better generalization. The recorded neural representations here are high dimensional, as assessed by SD, in line with previous studies on monkey PFC (Rigotti et al., 2013). This observation might seem at odds with the idea that high CCGP requires low dimensionality. However, SD is only one method for measuring dimensionality; it focuses on the ability of a linear classifier to decode a large number of dichotomies. SD is correlated but not identical to other measures of dimensionality based on the number of large principal components (see, e.g., Stringer et al., 2018; Machens et al., 2010; Mazzucato et al., 2016). A representation may be well described by a small number of components or dimensions but still have a large SD if the noise is not too large, especially along dimensions relevant for decoding the different dichotomies. Our data confirm that SD can be high when a principal component analysis (PCA)-based measure of dimensionality is low. This discrepancy is particularly evident in the interval that precedes stimulus onset. When the stimulus appears, both PCA dimensionality and SD increase (see Figure S3C and Methods S4).

The observed increase in dimensionality upon stimulus appearance is probably due to the fact that stimulus identity and context need to be mixed non-linearly to make a correct decision on our task. Any non-linear mixing leads to higher dimensional representations (Rigotti et al., 2013; Fusi et al., 2016). The decision on this task is likely made in the time between when stimulus identity becomes decodable and when expected reinforcement value and selected action become decodable. Representations of stimulus identity, planned action, and expected reward emerge rapidly in DLPFC and ACC, faster than in HPC, suggesting that they play a more prominent role in the decision process. The time interval between when stimulus identity and value and action become decodable is extremely short (Figure 4). We lack sufficient data to estimate dimensionality in this interval. As a result, we analyze a larger time window that also includes time bins in which the decision is already made; time bins after the decision have been shown to exhibit lower PCA dimensionality and SD (Rigotti et al., 2013). Nonetheless, the variable context is not represented in an abstract format in DLPFC in this longer time interval, and it is only weakly abstract in ACC. Future studies will require a larger number of recorded neurons to reveal whether the increase in dimensionality and the decrease in CCGP for context, observed in DLPFC and ACC, may be accounted for by non-linear mixing of information about context and stimulus identity.

The Role of Abstraction in Reinforcement Learning

Abstraction provides a solution for the notorious "curse of dimensionality," the exponential growth of the solution space

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

required to encode all states of the environment (Bellman, 1957). Most abstraction techniques in RL can be divided into 2 main categories: "temporal abstraction" and "state abstraction." Temporal abstraction is the workhorse of Hierarchical RL (Dietterich, 2000; Precup, 2000; Barto and Mahadevan, 2003) and is based on the notion of temporally extended actions (or options), which can be thought of as an attempt to reduce the dimensionality of the space of action sequences. Instead of composing policies in terms of long action sequences, an agent can select options that automatically extend for several time steps.

State abstraction methods most closely relate to our work. State abstraction hides or removes information about the environment not critical for maximizing the reward function. This technique typically involves information hiding, clustering of states, and other forms of domain aggregation and reduction (Ponsen et al., 2009). Our use of neural networks as function approximators to represent a decision policy effectively constitutes a state abstraction method (see Methods S8). The inductive bias of neural networks induces generalization across inputs sharing a feature, mitigating the curse of dimensionality. The modeling demonstrates that neural networks create similar geometry to that observed in data, suggesting that our analysis techniques could be useful to elucidate the geometric properties underlying the success of Deep Q-learning neural networks trained to play 49 different Atari video games with super-human performance (Mnih et al., 2015). Future work will consider models that explicitly incorporate structures designed to encode the spatio-temporal statistics of sensory stimuli, motor responses, and reward history. This relational structure has been proposed to be supported by the HPC (Behrens et al., 2018; Whittington et al., 2019; Recanatesi et al., 2019; Benna and Fusi, 2019).

Other Forms of Abstraction in the Computational Literature

The principles delineated for representing variables in abstract format are reminiscent of recent work in computational linguistics. This work suggests that difficult lexical semantic tasks can be solved by word embeddings, which are vector representations whose geometric properties reflect the meaning of linguistic tokens (Mikolov et al., 2013a, 2013b). Recent forms of word embeddings exhibit linear compositionality that makes the solution of analogy relationships possible via linear algebra (Mikolov et al., 2013a, 2013b). For example, shallow neural networks trained in an unsupervised way on a large corpus of documents organize vector representations of common words such that the difference of the vectors representing "king" and "queen" is the same as the difference of vectors for "man" and "woman" (Mikolov et al., 2013b). These word embeddings, which can be translated along parallel directions to consistently change one feature (e.g., gender, as in the previous example), share common coding principles with the geometry of abstraction we describe. This type of vector representation predicts fMRI BOLD signals measured while subjects are presented with semantically meaningful stimuli (Mitchell et al., 2008).

A different approach to extracting compositional features in an unsupervised way relies on variational Bayesian inference to learn to infer interpretable factorized representations (usually called "disentangled" representations) of some inputs (Chen et al., 2016; Higgins et al., 2017; Chen et al., 2018; Kim and Mnih, 2018; Behrens et al., 2018). These methods can disentangle independent factors of variations of a variety of real-world datasets. Our analytical methods can help to gain insight into the functioning of these algorithms.

The capacity to represent variables in an abstract format is also critical for many other cognitive functions. For example, in vision, the creation of neural representations of objects invariant with respect to position, size, and orientation in the visual field is a typical abstraction process studied in machine learning (see e.g., Riesenhuber and Poggio, 1999; LeCun et al., 2015) and in the brain (Freedman et al., 2001; Rust and Dicarlo, 2010). This form of abstraction is sometimes referred to as "untangling" because the retinal representations of objects correspond to manifolds with a relatively low intrinsic dimensionality but are highly curved and tangled together before becoming "untangled" in visual cortex (DiCarlo and Cox, 2007; DiCarlo et al., 2012); untangled representations are not the same as disentangled representations. Untangling typically requires transformations that either increase the dimensionality of representations by projecting into a higher dimensional space or decrease dimensionality by extracting relevant features.

In studies in vision, the final representation is typically required to be linearly separable for only one classification (e.g., car versus non-car [DiCarlo and Cox, 2007; DiCarlo et al., 2012]). The geometry of the representation of "nuisance" variables that describe features of the visual input not relevant for the classification is typically not studied systematically. By contrast, the variables in our experiment are simple binary variables, allowing us to study systematically all possible dichotomies.

Conclusions

Our data demonstrate that geometries of representations exist that support high CCGP for multiple variables yet retain high SD. Thus, the same representation can in principle support 2 forms of flexibility, one characterized by generalization in novel situations, and the other by the ability to generate many responses to complex combinations of inputs.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
  - Lead Contact
  - Materials Availability
  - Data and Code Availability
- EXPERIMENTAL MODEL AND SUBJECT DETAILS
  - Monkeys
- METHOD DETAILS
  - Task and Behavior
  - Electrophysiological Recordings
- QUANTIFICATION AND STATISTICAL ANALYSIS
  - Selection of trials/neurons, and decoding analysis

- The cross-condition generalization performance (CCGP)
- CCGP and decoding performance of context in error trials
- The parallelism score (PS)
- Random models
- Shuffle of the data
- Geometric random model
- Random models and the analysis of different dichotomies
- Expected SD for a perfectly factorized representation
- Simulations of the multi-layer network
- Multi Dimensional Scaling (MDS) plots

SUPPLEMENTAL INFORMATION

Supplemental Information can be found online at https://doi.org/10.1016/j.cell.2020.09.031.

ACKNOWLEDGMENTS

We are grateful to L.F. Abbott and R. Axel for many useful comments on the manuscript. This project is supported by the Simons Foundation and by NIMH (1K08MH115365 and R01MH082017). S.F. and M.K.B. are also supported by the Gatsby Charitable Foundation, the Swartz Foundation, the Kavli foundation, and the NSF's NeuroNex Program award DBI-1707398. J.M. is supported by the Fyssen Foundation. S.B. received support from NIMH (1K08MH115365, T32MH015144, and R25MH086466) and from the American Psychiatric Association, Brain & Behavior Research Foundation Young Investigator, and Leon Levy Foundation fellowships.

AUTHOR CONTRIBUTIONS

Conceptualization and Methodology, S.B., M.K.B., M.R., J.M., S.F., and C.D.S.; Investigation, S.B., M.K.B., and M.R.; Software, S.B., M.K.B., M.R., J.M., and S.F.; Formal Analysis and Visualization, S.B., M.K.B., M.R., S.F., and C.D.S.; Writing, S.B., M.K.B., M.R., J.M., S.F., and C.D.S.; Resources, Administration, and Supervision, S.F. and C.D.S.; Funding, S.B., S.F., and C.D.S.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: January 21, 2020
Revised: June 9, 2020
Accepted: September 9, 2020
Published: October 14, 2020

SUPPORTING CITATIONS

The following references appear in the Supplemental Information: Kingma and Ba, 2014, Paszke et al., 2017

REFERENCES

Antzoulatos, E.G., and Miller, E.K. (2011). Differences between neural activity in prefrontal cortex and striatum during learning of novel abstract categories. Neuron *71*, 243–249.

Barak, O., Rigotti, M., and Fusi, S. (2013). The sparseness of mixed selectivity neurons controls the generalization-discrimination trade-off. J. Neurosci. *33*, 3844–3856.

Barto, A.G., and Mahadevan, S. (2003). Recent advances in hierarchical reinforcement learning. Discrete Event Dyn. Syst. *13*, 341–379.

Behrens, T.E.J., Muller, T.H., Whittington, J.C.R., Mark, S., Baram, A.B., Stachenfeld, K.L., and Kurth-Nelson, Z. (2018). What is a cognitive map? organizing knowledge for flexible behavior. Neuron *100*, 490–509.

Bellman, R.E. (1957). Dynamic Programming (Princeton University Press).

Benna, M.K., and Fusi, S. (2019). Are place cells just memory cells? Memory compression leads to spatial tuning and history dependence. bioRxiv. https://doi.org/10.1101/624239.

Borg, I., and Groenen, P. (2003). Modern multidimensional scaling: Theory and applications. J. Educ. Meas. *40*, 277–280.

Buckley, M.J., Mansouri, F.A., Hoda, H., Mahboubi, M., Browning, P.G.F., Kwok, S.C., Phillips, A., and Tanaka, K. (2009). Dissociable components of rule-guided behavior depend on distinct medial and prefrontal regions. Science *325*, 52–58.

Chang, L., and Tsao, D.Y. (2017). The code for facial identity in the primate brain. Cell *169*, 1013–1028.e14.

Chen, X. (2008). Confidence interval for the mean of a bounded random variable and its applications in point estimation. arXiv, 0802.3458.

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. (2016). Infogan: Interpretable representation learning by information maximizing generative adversarial nets. arXiv, 1606.03657.

Chen, T.Q., Li, X., Grosse, R.B., and Duvenaud, D.K. (2018). Isolating sources of disentanglement in variational autoencoders. arXiv, 1802.04942.

Dang, W., Jaffe, R.J., Qi, X.-L., and Constantinidis, C. (2020). Emergence of non-linear mixed selectivity in prefrontal cortex after training. bioRxiv. https://doi.org/10.1101/2020.08.02.233247. https://www.biorxiv.org/content/early/2020/08/02/2020.08.02.233247.

DiCarlo, J.J., and Cox, D.D. (2007). Untangling invariant object recognition. Trends Cogn. Sci. *11*, 333–341.

DiCarlo, J.J., Zoccolan, D., and Rust, N.C. (2012). How does the brain solve visual object recognition? Neuron *73*, 415–434.

Dietterich, T.G. (2000). Hierarchical reinforcement learning with the maxq value function decomposition. J. Artif. Intell. Res. *13*, 227–303.

Eichenbaum, H. (2004). Hippocampus: cognitive processes and neural representations that underlie declarative memory. Neuron *44*, 109–120.

Eichenbaum, H. (2017). On the integration of space, time, and memory. Neuron *95*, 1007–1018.

Freedman, D.J., Riesenhuber, M., Poggio, T., and Miller, E.K. (2001). Categorical representation of visual stimuli in the primate prefrontal cortex. Science *291*, 312–316.

Fusi, S., Miller, E.K., and Rigotti, M. (2016). Why neurons mix: high dimensionality for higher cognition. Curr. Opin. Neurobiol. *37*, 66–74.

Golland, P., Liang, F., Mukherjee, S., and Panchenko, D. (2005). Permutation tests for classification. In Learning Theory. COLT 2005. Lecture Notes in Computer Science, P. Auer and R. Meir, eds. (Springer), p. 3559. https://doi.org/10.1007/11503415_34.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. (2017). β-VAE: Learning basic visual concepts with a constrained variational framework (ICLR).

Horikawa, T., Tamaki, M., Miyawaki, Y., and Kamitani, Y. (2013). Neural decoding of visual imagery during sleep. Science *340*, 639–642.

Isik, L., Meyers, E.M., Leibo, J.Z., and Poggio, T. (2014). The dynamics of invariant object recognition in the human visual system. J. Neurophysiol. *111*, 91–102.

Isik, L., Tacchetti, A., and Poggio, T. (2018). A fast, invariant representation for human action in the visual system. J. Neurophysiol. *119*, 631–640.

Kim, H., and Mnih, A. (2018). Disentangling by factorising. arXiv, 1802.05983.

King, J.-R., and Dehaene, S. (2014). Characterizing the dynamics of mental representations: the temporal generalization method. Trends Cogn. Sci. *18*, 203–210.

Kingma, D.P., and Ba, J. (2014). Adam: A method for stochastic optimization. arXiv, 1412.6980.

Kumaran, D., Summerfield, J.J., Hassabis, D., and Maguire, E.A. (2009). Tracking the emergence of conceptual knowledge during human decision making. Neuron *63*, 889–901.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. Nature *521*, 436–444.

Lindsay, G.W., Rigotti, M., Warden, M.R., Miller, E.K., and Fusi, S. (2017). Hebbian learning in a random network captures selectivity properties of the prefrontal cortex. J. Neurosci. *37*, 11021–11036.

Machens, C.K., Romo, R., and Brody, C.D. (2010). Functional, but not anatomical, separation of "what" and "when" in prefrontal cortex. J. Neurosci. *30*, 350–360. https://doi.org/10.1523/JNEUROSCI.3276-09.2010.

Mazzucato, L., Fontanini, A., and La Camera, G. (2016). Stimuli reduce the dimensionality of cortical activity. Front. Syst. Neurosci. *10*, 11.

Meyers, E.M., Freedman, D.J., Kreiman, G., Miller, E.K., and Poggio, T. (2008). Dynamic population coding of category information in inferior temporal and prefrontal cortex. J. Neurophysiol. *100*, 1407–1419.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., and Dean, J. (2013a). Distributed representations of words and phrases and their compositionality. arXiv, 1310.4546.

Mikolov, T., Yih, W.-t., and Zweig, G. (2013b). Linguistic regularities in continuous space word representations. Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 746–751.

Miller, E.K., Nieder, A., Freedman, D.J., and Wallis, J.D. (2003). Neural correlates of categories and concepts. Curr. Opin. Neurobiol. *13*, 198–203.

Milner, B., Squire, L.R., and Kandel, E.R. (1998). Cognitive neuroscience and the study of memory. Neuron *20*, 445–468.

Mitchell, T.M., Shinkareva, S.V., Carlson, A., Chang, K.-M., Malave, V.L., Mason, R.A., and Just, M.A. (2008). Predicting human brain activity associated with the meanings of nouns. Science *320*, 1191–1195.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. Nature *518*, 529–533.

Morcos, A.S., Barrett, D.G., Rabinowitz, N.C., and Botvinick, M. (2018). On the importance of single directions for generalization. arXiv, 1803.06959.

Munuera, J., Rigotti, M., and Salzman, C.D. (2018). Shared neural coding for social hierarchy and reward value in primate amygdala. Nat. Neurosci. *21*, 415–423.

Parkinson, C., Liu, S., and Wheatley, T. (2014). A common cortical metric for spatial, temporal, and social distance. J. Neurosci. *34*, 1979–1987.

Parthasarathy, A., Herikstad, R., Bong, J.H., Medina, F.S., Libedinsky, C., and Yen, S.-C. (2017). Mixed selectivity morphs population codes in prefrontal cortex. Nat. Neurosci. *20*, 1770–1779.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). Automatic differentiation in pytorch. NIPS 2017 Autodiff Workshop.

Ponsen, M., Taylor, M.E., and Tuyls, K. (2009). Abstraction and generalization in reinforcement learning: A summary and framework. In International Workshop on Adaptive and Learning Agents (Springer), pp. 1–32.

Precup, D. (2000). Temporal abstraction in reinforcement learning. PhD thesis (University of Massachusetts Amherst).

Raposo, D., Kaufman, M.T., and Churchland, A.K. (2014). A category-free neural population supports evolving demands during decision-making. Nat. Neurosci. *17*, 1784–1792.

Recanatesi, S., Farrell, M., Lajoie, G., Deneve, S., Rigotti, M., and Shea-Brown, E. (2019). Predictive learning extracts latent space representations from sensory observations. bioRxiv. https://doi.org/10.1101/471987.

Riesenhuber, M., and Poggio, T. (1999). Hierarchical models of object recognition in cortex. Nat. Neurosci. *2*, 1019–1025.

Rigotti, M., Ben Dayan Rubin, D., Wang, X.-J., and Fusi, S. (2010). Internal representation of task rules by recurrent dynamics: the importance of the diversity of neural responses. Front. Comput. Neurosci. *4*, 24.

Rigotti, M., Barak, O., Warden, M.R., Wang, X.-J., Daw, N.D., Miller, E.K., and Fusi, S. (2013). The importance of mixed selectivity in complex cognitive tasks. Nature *497*, 585–590.

Rust, N.C., and Dicarlo, J.J. (2010). Selectivity and tolerance ("invariance") both increase as visual information propagates from cortical area V4 to IT. J. Neurosci. *30*, 12978–12995.

Saez, A., Rigotti, M., Ostojic, S., Fusi, S., and Salzman, C.D. (2015). Abstract context representations in primate amygdala and prefrontal cortex. Neuron *87*, 869–881.

Schapiro, A.C., Turk-Browne, N.B., Norman, K.A., and Botvinick, M.M. (2016). Statistical learning of temporal community structure in the hippocampus. Hippocampus *26*, 3–8.

Stefanini, F., Kushnir, L., Jimenez, J.C., Jennings, J.H., Woods, N.I., Stuber, G.D., Kheirbek, M.A., Hen, R., and Fusi, S. (2020). A distributed neural code in the dentate gyrus and in ca1. Neuron *107*, 703–716.

Stringer, C., Pachitariu, M., Steinmetz, N., Carandini, M., and Harris, K.D. (2018). High-dimensional geometry of population responses in visual cortex. bioRxiv. https://doi.org/10.1101/374090.

Wallis, J.D., Anderson, K.C., and Miller, E.K. (2001). Single neurons in prefrontal cortex encode abstract rules. Nature *411*, 953–956.

Whittington, J.C., Muller, T.H., Mark, S., Chen, G., Barry, C., Burgess, N., and Behrens, T.E. (2019). The tolman-eichenbaum machine: Unifying space and relational memory through generalisation in the hippocampal formation. bioRxiv, 770495.

Wirth, S., Yanike, M., Frank, L.M., Smith, A.C., Brown, E.N., and Suzuki, W.A. (2003). Single neurons in the monkey hippocampus and learning of new associations. Science *300*, 1578–1581.

Wutz, A., Loonis, R., Roy, J.E., Donoghue, J.A., and Miller, E.K. (2018). Different levels of category abstraction by different dynamics in different prefrontal areas. Neuron *97*, 716–726.e8.

Zabicki, A., de Haas, B., Zentgraf, K., Stark, R., Munzert, J., and Krüger, B. (2017). Imagined and executed actions in the human motor system: testing neural similarity between execution and imagery of actions with a multivariate approach. Cereb. Cortex *27*, 4523–4536.

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Experimental Models: Organisms/Strains		
Rhesus Monkeys (*macaca mulatta*)	National Primate Centers	https://nprcresearch.org/primate/
Software and Algorithms		
Plexon Offline Sorter	Plexon, Inc.	https://plexon.com
Expo Software	NYU	https://sites.google.com/a/nyu.edu/expo/home
MATLAB	Mathworks	https://www.mathworks.com/products/matlab.html; RRID:SCR_001622
PyTorch	pytorch.org	PyTorch 1.2
Decoding, CCGP, parallelism score, and dimensionality algorithms	In-house MATLAB scripts	N/A
MNIST abstraction experiments	In-house MATLAB scripts	N/A
Supervised learning and Deep Q-learning simulations	In-house PyTorch scripts	N/A
BrainSight 2 System Software	Rogue Research	https://www.rogue-research.com

RESOURCE AVAILABILITY

Lead Contact

Further information and requests for resources should be directed to and will be fulfilled by the Lead Contact, C. Daniel Salzman (cds2005@columbia.edu).

Materials Availability

This study did not generate new unique reagents.

Data and Code Availability

The datasets and analysis code supporting the current study are available from the lead contact on request.

EXPERIMENTAL MODEL AND SUBJECT DETAILS

Monkeys

Two male rhesus monkeys (Macaca mulatta; two males, 10 years old, 8kg: 11 years old, 13 kg respectively) were used in these experiments. Experiments were performed in an AAALAC approved facility at New York State Psychiatric Institute which provided for paired housing of non-human primates as well as regular access to play cages and a robust environmental enrichment program. All experimental procedures were performed in accordance with the National Institutes of Health guide for the care and use of laboratory animals and the Animal Care and Use Committees at New York State Psychiatric Institute and Columbia University.

METHOD DETAILS

Task and Behavior

Monkeys performed a serial-reversal learning task in which they were presented one of four visual stimuli (fractal patterns). Stimuli were consistent across contexts and sessions, presented in random order. Each trial began with the animal holding down a button and fixating for 400 ms (Figure 1A). If those conditions were satisfied, one of the four stimuli was displayed on a screen for 500 ms. In each context, correct performance for two of the stimuli required releasing the button within 900 ms of stimulus disappearance; for the other two, the correct operant action was to continue to hold the button. For 2 of the 4 stimuli, correct performance resulted in reward delivery; for the other 2, correct performance did not result in reward. If the monkey performed the correct action, a trace interval of 500 ms ensued followed by the liquid reward or by a new trial in the case of non-rewarded stimuli. If the monkey made

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

a mistake, a 500 ms time out was followed by the repetition of the same trial type if the stimulus was a non-rewarded one. In the case of incorrect responses to rewarded stimuli, the time-out was not followed by trial repetition and the monkey simply lost his reward. After a random number of trials between 50 and 70, the context switched without warning. Upon a context switch, operant contingencies switched for all images, but for two stimuli the reinforcement contingencies did not change, in order to maintain orthogonality between operant and reinforcement contingencies. A different colored frame (red or blue) for each context appears on the edges of the monitor on 10 percent of the trials, randomly selected, and only on specific stimulus types (stimulus C for context 1 and stimulus D for context 2). This frame never appeared in the first five trials following a context switch. All trials with a contextual frame were excluded from all analyses presented.

Electrophysiological Recordings

Recordings began only after the monkeys were fully proficient in the task and performance was stable. Recordings were conducted with multi-contact vertical arrays electrodes (v-probes, Plexon Inc., Dallas, TX) with 16 contacts spaced at 100 μm intervals in ACC and DLPFC, and 24 contacts in HPC, using the Omniplex system (Plexon Inc.). In each session, we individually advanced the arrays into the three brain areas using a motorized multi-electrode drive (NAN Instruments). Analog signals were amplified, band-pass filtered (250 Hz - 8 kHz), and digitized (40 kHz) using a Plexon Omniplex system (Plexon, Inc.). Single units were isolated offline using Plexon Offline Sorter (Plexon, Inc.). To address the possibility that overlapping neural activity was recorded on adjacent contacts, or that two different clusters visible using principal component analysis (PCA) belonged to the same neuron, we compared the zero-shift cross-correlation in the spike trains with a 0.2 ms bin width of each neuron identified in the same area in the same session. If 10 percent of spikes co-occurred, the clusters were considered duplicated and one was eliminated. If 1-10 percent of spikes co-occurred, the cluster was flagged and isolation was checked for a possible third contaminant cell. Recording sites in DLPFC were located in Brodmann areas 8, 9 and 46. Recording sites in ACC were in the ventral bank of the ACC sulcus (area 24c). HPC recordings were largely in the anterior third, spanning across CA1-CA2-CA3 and DG.

QUANTIFICATION AND STATISTICAL ANALYSIS

Selection of trials/neurons, and decoding analysis

The neural population decoding algorithm was based on a linear classifier (see e.g., (Saez et al., 2015)) trained on pseudo-simultaneous population response vectors composed of the spike counts of the recorded neurons within specified time bins and in specific trials (Meyers et al., 2008). The trials used in decoding analyses, unless noted otherwise, are only those in which the animal responded correctly (both for the current trial and the directly preceding one), in which no context frame was shown (neither during the current nor the preceding trial), and which occurred at least five trials after the most recent context switch. We retain all neurons for which we have recorded at least 15 trials satisfying these requirements for each of the eight experimental conditions (i.e., combinations of context, value and action). Every decoding analysis is averaged across many repetitions to estimate trial-to-trial variability (as explained more in detail below). For every repetition, we randomly split off five trials per condition from among all selected trials to serve as our test set, and used the remaining trials (at least ten per condition) as our training set. For every neuron and every time bin, we normalized the distribution of spike counts across all trials in all conditions with means and standard deviations computed on the trials in the training set. Specifically, given an experimental condition c (i.e., a combination of context, value and action) in a time bin t under consideration, we generated the pseudo-simultaneous population response vectors by sampling, for every neuron i, the z-scored spike count in a randomly selected trial in condition c, which we indicate by $n_i^c(t)$. This resulted in a single-trial population response vector $n^c(t) = (n_1^c(t), n_2^c(t), \ldots, n_N^c(t))$, where N corresponds to the number of recorded neurons in an area under consideration. This single-trial response vector can be thought of as a noisy measurement of an underlying mean firing rate vector $\overline{n}^c(t)$, such that $n^c(t) = \overline{n}^c(t) + \eta^c(t)$, with $\eta^c(t)$ indicating a noise vector modeling the trial-to-trial variability of spike counts. Assuming that the trial-to-trial noise is centered at zero, we estimate the mean firing rate vectors taking the sample average: $\overline{n}^c(t) \approx \langle n^c(t) \rangle$, where the angular brackets indicate averaging across trials. We then either trained maximum margin (SVM) linear classifiers on the estimated mean firing rate vectors for the eight conditions in the training set (this is the approach adopted to train the decoders used to compute CCGP, see below), or we trained such classifiers on the single-trial population response vectors generated from the training set of trials (this is what we used in all the figures that report the "decoding accuracy"). In the latter case, in order to obtain a number of trials that is large compared to the number of neurons, we re-sampled the noise by randomly picking noisy firing rates (i.e., spike counts) from among all the training trials of a given experimental condition for each neuron independently. Specifically, in this case, we re-sampled 10,000 trials per condition from the training set. While this neglected correlations between different neurons within conditions, we had little information about these correlations in the first place, since only a relatively small numbers of neurons were recorded simultaneously. Regardless of whether we trained on estimated mean firing rate vectors or on re-sampled single-trial population response vectors, the decoding performance was measured in a cross-validated manner on 1,000 re-sampled single-trial population response vectors generated from the test set of trials. For every decoding analysis training and testing were then repeated 1,000 times over different random partitions of the trials into training and test trials. The decoding accuracies that we report were computed as the average results across repetitions.

Statistical significance of the decoding accuracy was assessed using a permutation test for classification (Golland et al., 2005). Specifically, we repeated the same procedure just described, but at the beginning of every repetition of a decoding analysis, trials

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

were shuffled, i.e., associated to a random condition. This is a way of estimating the probability that the population decoders that we used would have given the same results that we obtained by chance, i.e., when applied on data that contain no information regarding the experimental conditions.

In Figure S3B we show the cross-validated decoding accuracy as a function of time throughout the trial (for a sliding 500 ms time window) for maximum margin classifiers trained only on the mean neural activities for each condition. Figures 3A and 5A show similar results for linear classifiers trained on the mean firing rates in the neural data within time windows from −800 ms to 100 ms and from 100 ms to 1000 ms relative to stimulus onset, respectively.

For all analyses, data were combined across monkeys, because all key features of the dataset were consistent across the two monkeys.

The cross-condition generalization performance (CCGP)

The hallmark feature of neural representations of variables in abstract format (“abstract variables”) is their ability to support generalization in novel situations. When several abstract (in our case binary) variables are encoded simultaneously, generalization must be possible for all the abstract variables. We quantify a strong form of generalization using a measure we call the cross-condition generalization performance (CCGP.) We use this measure as a quantitative definition of the degree to which a variable is represented in abstract format. CCGP is distinct from traditional cross-validated decoding performance commonly employed to determine if a neural ensemble represents a variable. In traditional cross-validated decoding, the data is split up randomly such that trials from all conditions will be present in both the training and test sets. For CCGP, trials are split instead according to their condition labels, such that the training set consists entirely of trials from one group of conditions, while the test set consists only of trials from a disjoint group of conditions (see the scheme in Figure S1A). In computing CCGP, we train a linear classifier for a certain dichotomy that discriminates the conditions in the training set according to some label (one of the variables), and then ask whether this discrimination generalizes to the test set by measuring the classification performance on the data from entirely different conditions, i.e., conditions not used for training the decoder. Since the conditions used for testing were not used for training, they are analagous to novel situations (conditions the trained decoder has never experienced). We always report the average CCGP across all possible ways of choosing training and testing conditions (see below); thus CCGP provides a continuous measure which quantifies the degree of abstraction.

Given our experimental design with eight different conditions (distinguished by context, value and action of a trial), we can investigate variables corresponding to different balanced (four versus four condition) dichotomies, and choose one, two or three conditions from each side of a dichotomy to form our training set. We use the remaining conditions (three, two or one from either side, respectively) for testing, with larger training sets typically leading to better generalization performance. For different choices of training conditions we will in general obtain different values of the classification performance on the test conditions, and we define CCGP as its average over all possible sets of training conditions (of a given size). In Figures 3A and 5A we show the CCGP (on the held out fourth condition) when training on three conditions from either side of the 35 balanced dichotomies (with dichotomies corresponding to context, value and action highlighted). Note that traditional decoding will always have a performance level as high or higher than CCGP, but high traditional decoding does not ensure that CCGP will be different from chance.

We emphasize that in order to achieve high CCGP, it is not sufficient to merely generalize over the noise associated with trial-to-trial fluctuations of the neural activity around the mean firing rates corresponding to individual conditions. Instead, the classifier has to generalize also across different conditions on the same side of a dichotomy, i.e., across those conditions that belong to the same category according to the variable under consideration.

For the CCGP analysis, the selection of trials used is the same as for the decoding analysis, except that here we retain all neurons that have at least ten trials for each experimental condition that meet our selection criteria (since the split into training and test sets is determined by the labels of the eight conditions themselves, so that for a training condition we don’t need to hold out additional test trials). We pre-process the data by z-scoring each neuron’s spike count distribution separately. Again, we can either train a maximum margin linear classifier only on the cluster centers, or on the full training set with trial-to-trial fluctuations (noise), in which case we resample 10,000 trials per condition, with Figures 3A and 5A showing results using the latter method.

For all analyses, data were combined across monkeys, because all key features of the dataset were consistent across the two monkeys.

CCGP and decoding performance of context in error trials

In order to measure CCGP and decoding performance of context in error trials we had to address the issue that, due to the high behavioral performance of the monkeys, error trials are scarce compared to correct trials. We therefore adapted our approach for computing CCGP and traditional decoding performance in two ways. First, we performed all training procedures only on correct trials, reserving held-out error trials for testing, which could then be compared to testing on held-out correct trials. As a result, our analyses of the geometry of representations in relation to behavioral performance asks specifically about the difference between correct and error trials in terms of the cross-validation performance of CCGP and traditional decoding. Note also that we only considered error trials that occurred more than 5 trials after the context switch in this analysis, similar to that done for analyses of correct trials. Second, we recognized that decoding performance on trials for a specific trial condition can be cross-validated irrespective of any other type of trial condition. This means that a neuron can participate in the population response vector for a given condition if it has been recorded for a sufficient number of error trials for that condition, even if there are not enough recorded trials with errors for other

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

conditions. We therefore built held-out (error trial) population response vectors for each condition independently, including for each such condition only neurons that have enough error trials for that condition. For each condition, on the training set of trials we trained decoders only on the neurons that also participate in the held-out response vectors for error trials (see Table S1). This approach allows us to include many more neurons in the analysis, which was critical for gaining statistical power. This combination of techniques resulted in our being able to include in our analysis 440, 233 and 223 neurons in HPC, DLPFC and ACC, respectively, with at least 2 trials per held-out condition for a CCGP analysis on error trials, and 385, 198 and 184 neurons with at least 3 trials per held-out condition for the traditional decoding analysis on error trials. We also required 18 trials for training, which is performed on correct trials for both CCGP and decoding. All analyses are then performed 10000 times (to estimate confidence intervals with bootstrap resampling), each time sampling 180 neurons from each area and subsampling cross-validation trials so as to equalize the number of held-out correct and error trials. Subsampling 180 neurons from each brain area ensures that comparison across brain areas is not skewed by the numbers of neurons used in the analyses. For each iteration, the same set of 180 sub-sampled neurons is used to decode correct and error trials whether using CCGP or traditional decoding. The results of the CCGP analysis on error versus correct trials, and the corresponding traditional decoding analysis is presented in Figure 6.

The parallelism score (PS)

We developed a measure based on angles of coding directions to characterize the geometry of neural representations of variables in the firing rate space. Consider a pair of conditions, one from each side of a dichotomy, such as the two conditions that correspond to the presentation of stimulus A in the two contexts (here context is the dichotomy under consideration). A linear classifier trained on this pair of conditions defines a separating hyperplane. The weight vector orthogonal to this hyperplane aligns with the vector connecting the two points that correspond to the mean firing rates for the two training conditions if we assume isotropic noise around both of them. This corresponds to the coding direction for the variable under consideration (context in the example). Other coding directions for the same variable can be obtained by choosing a different pair of training conditions (e.g., the conditions that correspond to the presentation of stimulus B in the two context). The separating hyperplane associated with one pair of training conditions is more likely to correctly generalize to another pair of conditions if the associated coding directions are parallel (as illustrated in Figure S1). The parallelism score (PS) that we developed directly quantifies the alignment of these coding directions.

If we had only four conditions as shown in Figure S1, there would be only two coding directions for a given variable (from the two pairs of training conditions), and we would simply calculate the cosine of the angle between them (i.e., the normalized overlap of the two weight vectors). In our experiments, there were 8 conditions whose mean firing rates we denote by $f(c)$, with $c = 1, 2, \ldots, 8$. A balanced dichotomy corresponds to splitting up these eight conditions into two disjoint groups of four, corresponding to the conditions to be classified as positive and negative, respectively, e.g., $G_{pos} = [1, 2, 4, 7]$ versus $G_{neg} = [3, 5, 6, 8]$. To compute four unit coding vectors $\overrightarrow{v_i}$ for $i = 1, 2, 3, 4$ (corresponding to four pairs of potential training conditions) for the variable associated with this dichotomy, we have to match each condition in the positive group with a unique condition in the negative group (without repetitions). We parametrize these pairings by considering all possible permutations of the condition indices in the negative group. For a particular choice of such a permutation $\mathscr{P}$ the resulting set of coding vectors is given by

$$v_i = \frac{f\left(G^i_{pos}\right) - f\left(\mathscr{P}\left(G_{neg}\right)^i\right)}{\left|f\left(G^i_{pos}\right) - f\left(\mathscr{P}\left(G_{neg}\right)^i\right)\right|}.$$

Note that we have defined the v_i as normalized coding vectors, since we want our parallelism score to depend only on their direction (but not on the magnitude of the un-normalized coding vectors). To compute the parallelism score from these unit coding vectors, we consider the cosines of the angles between any two of them $\cos(\theta_{ij}) = \overrightarrow{v_i}.\overrightarrow{v_j}$ and we average these cosines over all six of these angles (corresponding to all possible choices of two different coding vectors).

$$\langle \cos\theta \rangle = \frac{1}{12}\sum_{i=1}^{4}\sum_{j\neq i}^{4}\cos(\theta_{ij}) = \frac{1}{6}\sum_{i=1}^{4}\sum_{j>i}^{4}\cos(\theta_{ij}).$$

In general there are multiple ways of pairing up conditions corresponding to the two values of the variable under consideration. We don't want to assume *a priori* that we know the 'correct' way of pairing up conditions. For example, it is not obvious that the two conditions corresponding to the same stimuli in two contexts are those that maximize the cosine between the coding vectors. It could be that cosine is larger when the conditions corresponding to a certain value are paired. In other words, to perform the analysis in an unbiased way, we should ignore the labels of the variables that define the conditions within each dichotomy (in the case of context we should just consider all conditions corresponding to context 1 and pair them in all possible ways to all conditions in context 2). So we consider all possible ways of matching up the conditions on the two sides of the dichotomy one-to-one, corresponding to all possible permutations $\mathscr{P}$, and then define the PS as the maximum of the average cosine across all possible pairings/permutations. There are two such pairings in the case of four conditions, and 24 for eight conditions. In general there are $(m/2)!$ pairings for m conditions, so if m was large there would be a combinatorial explosion in the obvious generalization of this definition to arbitrary m, which would also require averaging the cosines of $(m/2)(m/2-1)/2$ angles.

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

The parallelism score for a given balanced dichotomy in our case of eight conditions is defined as

$$\text{Parallelism Score} = \max_{\text{permutations } \mathcal{P}} \langle \cos\theta \rangle.$$

Note that this quantity depends only on the normalized coding directions (for the best possible pairing of conditions), which are simply the unit vectors pointing from one cluster center (mean firing rate for a given condition) toward another. Therefore, finding the PS doesn't require training any classifiers, which makes it a very simple, fast computation (unless m is large). However, because it depends only on the locations of the cluster centers, the parallelism score ignores the shape of the noise (within condition trial-to-trial fluctuations).

The parallelism scores of all 35 dichotomies in our data (with the context, value and action dichotomies highlighted) are plotted in Figures 3B and 5B. The selection of trials used in this analysis is the same as for the decoding and cross-condition generalization analyses, retaining all neurons that have at least ten trials for each experimental condition that meet our selection criteria, and z-scoring each neuron's spike count distribution individually.

Note that a high parallelism score for one variable/dichotomy doesn't necessarily imply high cross-condition generalization. Even if the coding vectors for a given variable are approximately parallel, the test conditions might be much closer together than the training conditions. In this case generalization would likely be poor. In addition, for the simple example of only four conditions the orthogonal dichotomy would have a low parallelism score in such a situation (corresponding to a trapezoidal geometry).

We also emphasize that high parallelism scores for multiple variables do not guarantee good generalization (large CCGP) of one dichotomy across another one. When training a linear classifier on noisy data, the shape of the noise clouds could skew the weight vector of a maximum margin classifier away from the vector connecting the cluster centers of the training conditions. Moreover, even if this is not the case (e.g., if the noise is isotropic), generalization might still fail because of a lack of orthogonality of the coding directions for different variables. (For example, the four conditions might be arranged at the corners of a parallelogram instead of a rectangle, or in the shape of a parallelepiped instead of a cuboid for eight conditions).

In summary, while the parallelism score is not equivalent to CCGP, high scores for a number of dichotomies with orthogonal labels characterize a family of (approximately factorizable) geometries that can lead to good generalization properties if the noise is sufficiently well behaved (consider e.g., the case of the principal axes of the noise distributions being aligned with the coding vectors). For the simple case of isotropic noise, if the coding directions for different variables are approximately orthogonal to each other, CCGP will also be high.

For all analyses, data were combined across monkeys, because all key features of the dataset were consistent across the two monkeys.

Random models

In order to assess the statistical significance of the above analyses we need to compare our results (for the decoding performance, abstraction index, cross-condition generalization performance, and parallelism score, which we collectively refer to as scores here) to the distribution of values expected from an appropriately defined random control model. There are various sensible choices for such random models, each corresponding to a somewhat different null hypothesis we might want to reject. We consider two different classes of random models, and for each of our abstraction analyses we choose the more conservative one of the two to compute error bars around the chance levels of the scores (i.e., we only show the one that leads to the larger standard deviation). We note that although we examined all 35 dichotomies in the same way, we had pre-registered interest in the 3 binary variables context, value and action.

Shuffle of the data

One simple random model we consider is a shuffle of the data, in which we assign a new, random condition label to each trial for each neuron independently (in a manner that preserves the total number of trials for each condition). In other words, we randomly permute the condition labels (with values from 1 to 8) across all trials, and repeat this procedure separately for every neuron. When re-sampling artificial, noisy trials, we shuffle first, and then re-sample in a manner that respects the new, random condition labels as described above. This procedure destroys almost all structure in the data, except the marginal distributions of the firing rates of individual neurons. The error bars around chance level for the decoding performance in Figures 3, 5, and S5, and for the parallelism score in Figures 3, 5, and 7 are based on this shuffle control (showing plus/minus two standard deviations). These chance levels and error bars around them are estimated by performing the exact same decoding/PS analyses detailed in the preceding sections on the shuffled data, and repeating the whole shuffle analysis a sufficient number of times to obtain good estimates of the means and standard deviations of the resulting distributions of decoding performances/parallelism scores (e.g., for the PS we perform 1,000 shuffles).

Geometric random model

Another class of control models is more explicitly related to neural representations described by random geometries, and can be used to rule out a different type of null hypothesis. For the analyses that depend only on the cluster centers of the eight conditions (i.e., their mean firing rates, as e.g., for the PS), we can construct a random geometry by moving the cluster of points that correspond to different conditions to new random locations that are sampled from an isotropic Gaussian distribution. We then rescale all the vec-

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

tors to keep the total variance across all conditions (the signal variance, or variance of the centroids of the clusters). Such a random arrangement of the mean firing rates (cluster centers) is a very useful control to compare against, since such geometries do not constitute abstract neural representations, but nevertheless typically allow relevant variables to be decoded (see also Figure 2). For analyses that depend also on the structure of the within condition trial-to-trial fluctuations (in particular, CCGP and decoding with re-sampled trials), our random model in addition requires some assumptions about the noise distributions. We could simply choose identical isotropic noise distributions around each cluster center, but training a linear classifier on trials sampled from such a model would essentially be equivalent to training a maximum margin classifier only on the cluster centers themselves. Instead, we choose to preserve some of the noise structure of the data by moving the (re-sampled) noise clouds to the new random position of the corresponding cluster and performing a discrete rotation around it by permuting the axes separately for each condition. We basically shuffled the neuron labels in a different way for each condition. This shuffling corresponds to a discrete rotation of each noise cloud (i.e., the cloud of points that represents the set of all trials for one specific condition). The rotations are random and independent for each condition. While the structure of the signal is completely destroyed by generating a random set of cluster centers for the eight conditions, the within condition noise structure (but not the correlations across conditions) is retained in these models. If our scores are significantly different from those obtained using this random model, we can reject the null hypothesis that the data were generated by sampling a random isotropic geometry with the same total signal variance (i.e., the variance across the different cloud centers) and similarly shaped noise clouds as in the data. The error bars around chance level for the CCGP in Figures 3, 5, and 7 are derived from this geometric random control model by constructing many such random geometries and estimating the standard deviation of the resulting CCGPs.

Random models and the analysis of different dichotomies

One might be tempted to consider the scores for the 35 different dichotomies as a set of score values that defines a random model. Indeed, this could be described as a type of permutation of the condition labels, but only between groups of trials belonging to the same condition (thus preserving the eight groups of trials corresponding to separate conditions), as opposed to the random permutation across all trials performed in the shuffle detailed above. However, there are clearly correlations between the scores of different dichotomies (e.g., because the labels may be partially overlapping, i.e., not orthogonal). Therefore, we should not think of the set of scores for different dichotomies as resulting from a random model used to assess the probability of obtaining certain scores from less structured data. After all, the different dichotomies are simply different binary functions to be computed on the same neural representations, without changing any of their essential geometric properties. Instead, the set of scores for the 35 dichotomies allows us to make statements about the relative magnitude of the scores compared to those of other variables that may also be decodable from the data and possibly abstract, as shown in the form of bee-swarm plots in Figures 3, 5, and 7.

Expected SD for a perfectly factorized representation

As we showed in Figures 3 and 5, the measured SD in each brain area is significantly greater than the SD of a perfectly factorized representation. A perfectly factorized null model is constructed by placing the centroids of the noise clouds that represent the 8 different experimental conditions at the vertices of a cuboid. The cuboid is randomly rotated and embedded in an N-dimensional firing rate space, where N equals the number of neurons that pass the selection criteria for the decoding analysis of the experimental data. The lengths of the sides of the cuboid are tuned to reproduce (on average) the CCGP values observed in the experiment for the variables context, value, and action. We then sample 10,000 trials for each condition from a Gaussian distribution with unit covariance matrix (centered on the vertex corresponding to that condition) in this firing rate space. From this dataset - artificially generated from a perfectly factorized model - we calculate the SD and CCGP, just as in the analyses of experimental data. This procedure is repeated 100 times for each of the 3 brain areas and 2 time intervals, with results for the early time interval shown in Figure 3E, and those for the late time interval shown in Figure 5E.

Simulations of the multi-layer network

The two hidden layer network depicted in Figure 7 contains 768 neurons in the input layer, 100 in each hidden layer and four neurons in the output layer. We used eight digits (1-8) of the full MNIST dataset to match the number of conditions we considered in the analysis of the experiment. The training set contained 48128 images and the test set contained 8011 digits. The network was trained to output the parity and the magnitude of each digit and to report it using four output units: one for odd, one for even, one for small (i.e., a digit smaller than 5) and one for large (a digit larger than 4). We trained the network using the back-propagation algorithm 'train' of MATLAB (with the neural networks package). We used a tan-sigmoidal transfer function ('tansig' in MATLAB), the mean squared normalized error ('mse') as the cost function, and the maximum number of training epochs was set to 400. After training, we performed the analysis of the neural representations using the same analytical tools that we used for the experimental data, except that we did not z-score the neural activities, since they were simultaneously observed in the simulations.

The description of the methods to model our task using a reinforcement learning algorithm (Deep Q-learning) appears in Methods S8 Deep neural network models of task performance.

Please cite this article in press as: Bernardi et al., The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex, Cell (2020), https://doi.org/10.1016/j.cell.2020.09.031

Multi Dimensional Scaling (MDS) plots

All MDS plots (Figures 7E–7G; Figure S4) are obtained as follows. Within a chosen time bin the activity of each neuron across all conditions is z-scored. It is then averaged across trials within each condition to obtain the firing rate patterns for each condition. These patterns are then used to construct an $n_c \times n_c$ dissimilarity matrix (where n_c is the number of conditions), which simply tabulates the Euclidean distance between firing rate patterns for each pair of conditions. This dissimilarity matrix is then centered, diagonalized, and projected along the first 3 eigenvectors rescaled by the squared root of the corresponding eigenvalues, in accordance with the Classical Multidimensional Scaling algorithm (Borg and Groenen 2003).

Supplemental Figures

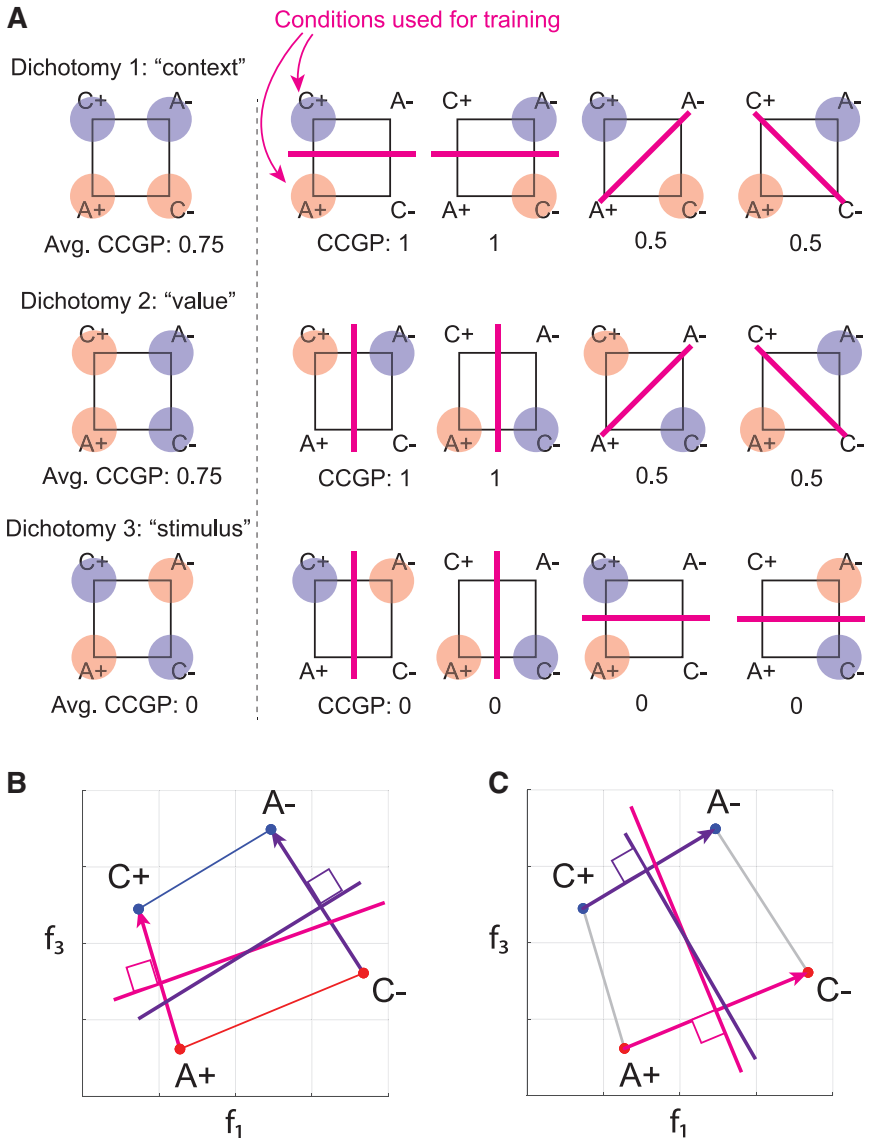


Figure S1. Schemes that Illustrate How CCGP and Parallelism Score Are Computed, Related to Figure 2

Scheme that explains how CCGP is computed for all the dichotomies in an experiment with 4 conditions (2 stimuli in each of the two contexts). The example has the same geometry as the representation in Figures 2C and 2D (value and context are simultaneously abstract). Each dichotomy corresponds to a different way of dividing the 4 conditions in two groups of two (different colored clouds for each group). In this example, all possible dichotomies correspond to variables that have a name (''context,'' ''value'' and ''stimulus''). However, this is not necessarily true in other situations. For each dichotomy, CCGP is computed by training a linear decoder on a subset of conditions. These are the only conditions highlighted (by shaded colored circles) in the figure. The decoder is then tested on the remaining (non-shaded) conditions. For each dichotomy there are 4 possible ways of choosing 2 conditions, one from each side of the dichotomy, as illustrated in the figure in the row next to each dichotomy. The final CCGP is obtained by computing the average test performance over all the different ways of choosing two training conditions. For this geometry, CCGP is 0.75 for the dichotomies context and value, and 0 for the dichotomy stimulus. These values are obtained under the assumption that the noise is isotropic and not too large. For context and value CCGP is equal to one only for two choices of the conditions used for training. This is a peculiarity of this simple case which contains only 4 points in the firing rate space. By contrast, consider the case of 8 points arranged on a cube. Here the CCGP for the analogous dichotomy (i.e., one that separates the 4 points of one face from the 4 points of the opposite face) is equal to 1 for all possible choices of 6 training conditions (3 conditions per face). Dichotomy 3 appears not to be linearly decodable (and thus neither traditional decoding nor CCGP will have performance above chance). However, this scheme is highly simplified, and in the real data we observed variables that are decodable and not in an abstract format (e.g., action in the hippocampus in the interval preceding stimulus appearance, and context in DLPFC during the interval following stimulus presentation). One can visualize a situation in which a variable is decodable but not in an abstract format by introducing a third dimension to the scheme shown for Dichotomy 3. This

(legend continued on next page)

scheme places the 4 points on a squeezed cube, with the two red points on one face and the two blue points on the other face. If the distance between these two faces is small compared to the other distances, then the CCGP will be approximately the same as shown in the Figure for all 3 dichotomies, but now Dichotomy 3 would be decodable at above chance levels. **b,c.** Scheme that explains the Parallelism Score (PS). In the two panels we show the firing rate space of two neurons. These neural representations are similar to those in Figures 2C and 2D, which allow for cross-condition generalization for both context and value. b. Training a linear classifier to decode context on the two rewarded conditions leads to the magenta separating hyperplane, which is defined by a weight vector orthogonal to it. Similarly, training on the unrewarded conditions leads to the dark purple hyperplane and and its associated orthogonal weight vector. If these two weight vectors are close to parallel, the corresponding classifiers are more likely to generalize to the conditions not used for training. The parallelism score (PS) is defined as the cosine of the angle between these coding vectors, maximized over all possible ways of pairing up the conditions (see STAR Methods for details). c. Same as b but for the variable value.

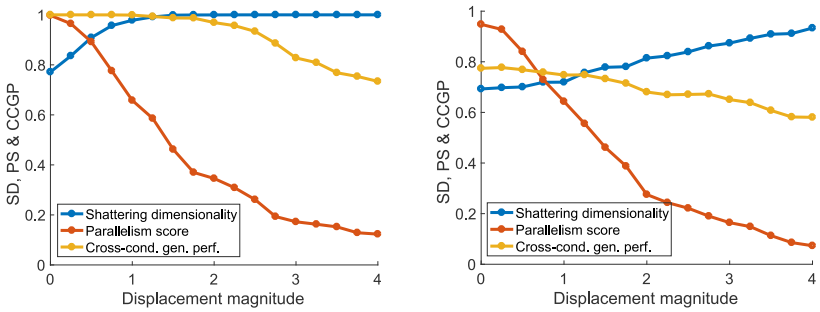


Figure S2. Trade-off between CCGP/PS and Shattering Dimensionality, Related to Figure 2

Analysis of an artificial dataset generated by randomly embedding a (three-dimensional) cube in a 100-dimensional space, displacing its corners in independent random directions by a certain distance (displacement magnitude), and then sampling data points corresponding to the eight experimental conditions from isotropic Gaussian distributions around the cluster centers obtained from this distortion procedure. The shattering dimensionality (SD, blue) is plotted across all 35 balanced dichotomies, as well as the mean PS (red) and CCGP (yellow) of the three potentially abstract variables for low (left) and high noise (right), with noise sampled as i.i.d. unit Gaussian vectors multiplied by overall coefficients 0.2 and 1.0, respectively. For low noise, both SD and CCGP can simultaneously be close to one, indicating maximal dimensionality and the presence of three abstract variables for these representations. In the case of high noise, we observe a smooth tradeoff between CCGP (abstraction) and SD (dimensionality).

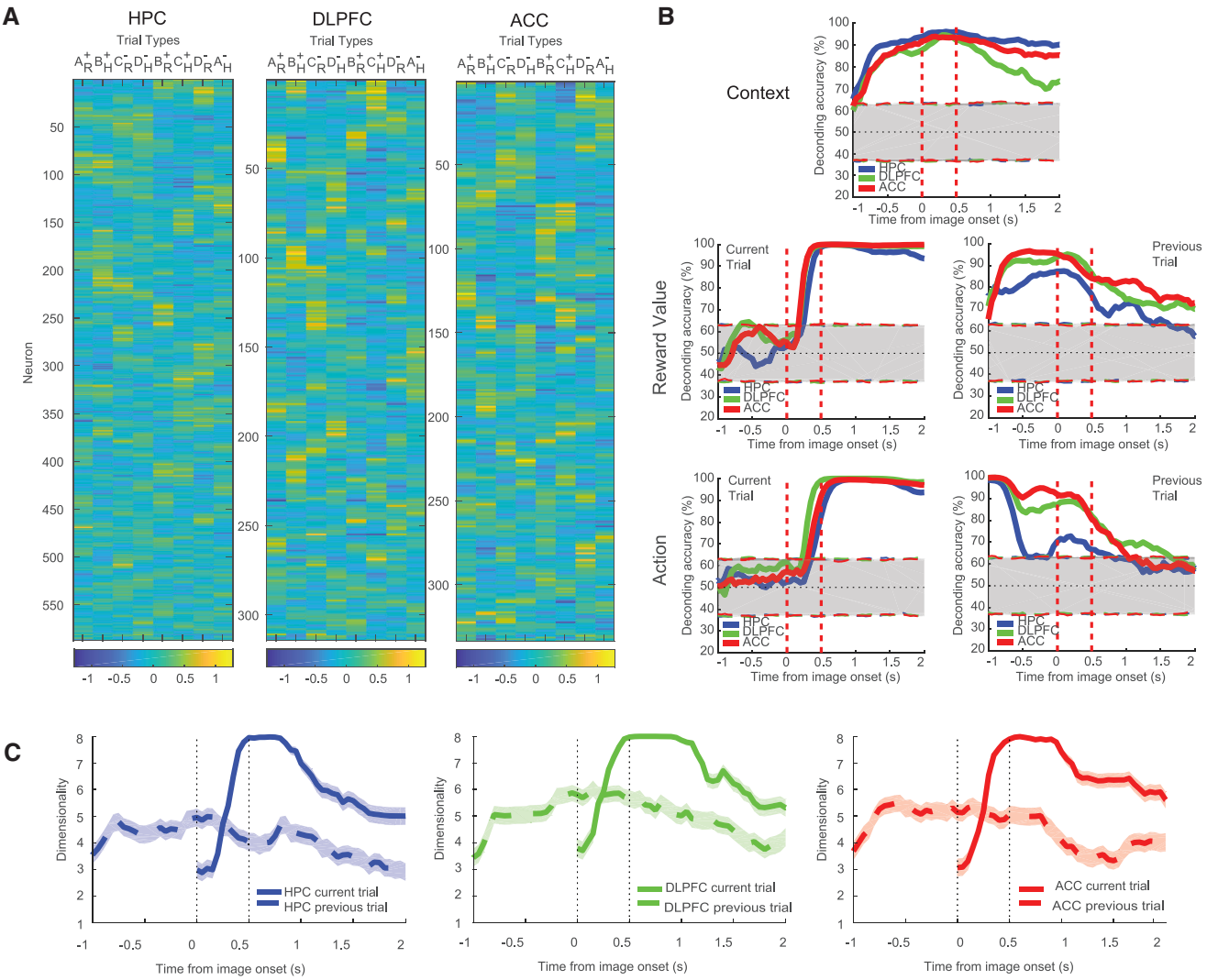


Figure S3. Single-Neuron Activity, Task-Relevant Variable Decoding, and Dimensionality, Related to Figure 4

a. The activity of all neurons recorded for at least ten trials per condition after excluding: 1) those trials performed incorrectly; 2) those trials in which a contextual frame was shown either for the current or the previous trial; and 3) those trials that occurred less than five trials after a context switch). Z-scored firing rates are calculated in the 900 ms time window that starts 800ms before stimulus onset. Each row represents the activity of an individual neuron. Different columns correspond to different trial conditions (i.e., the stimulus-action-outcome sequence preceding the interval). Neurons are ordered such that for adjacent rows, the eight-dimensional vectors of z-scored activities point in similar directions. The responses are very diverse in all three brain areas. b. Population level encoding of task-related variables. Performance of a linear decoder plotted a function of time relative to image onset for classifying a task-relevant variable. 1) Context on the current trial. 2) Reinforcement outcome on current (left) and prior (right) trials. 3) Operant action on current (left) and prior (right) trials. The decoding performance was computed in a 500-ms sliding window stepped every 50 ms across the trial for the three brain areas separately (blue, HPC; red, ACC; green, DLPFC). Shaded areas around chance level (dotted line) indicate two-sided 95%-confidence intervals calculated with a permutation test obtained by randomly shuffling trials (1,000 repetitions). The image is displayed on the screen from time 0 to 0.5 s. Analyses were run only on correct trials at least 5 trials after a context switch. c. Dimensionality of the average firing rate activity patterns as a function of time throughout the trial. The left panel illustrates the result of the analysis developed in (Machens et al., 2010) on HPC, the central panel refers to DLPFC, and the right panel to ACC. Continuous lines refer to the analysis carried out on average firing rate patterns obtained by averaging spike counts according to the task conditions of the trial that was being recorded (current trial). The dashed line shows the same but for conditions defined by the previous trial. The lines indicate the number of principal firing rate components that are larger than all noise components, averaged over 1000 re-samplings of the noise covariance matrix (see (Machens et al., 2010)). The shadings indicate the 95% confidence intervals estimated using the method for quantifying uncertainty around the mean of bounded random variables presented in (Chen 2008).

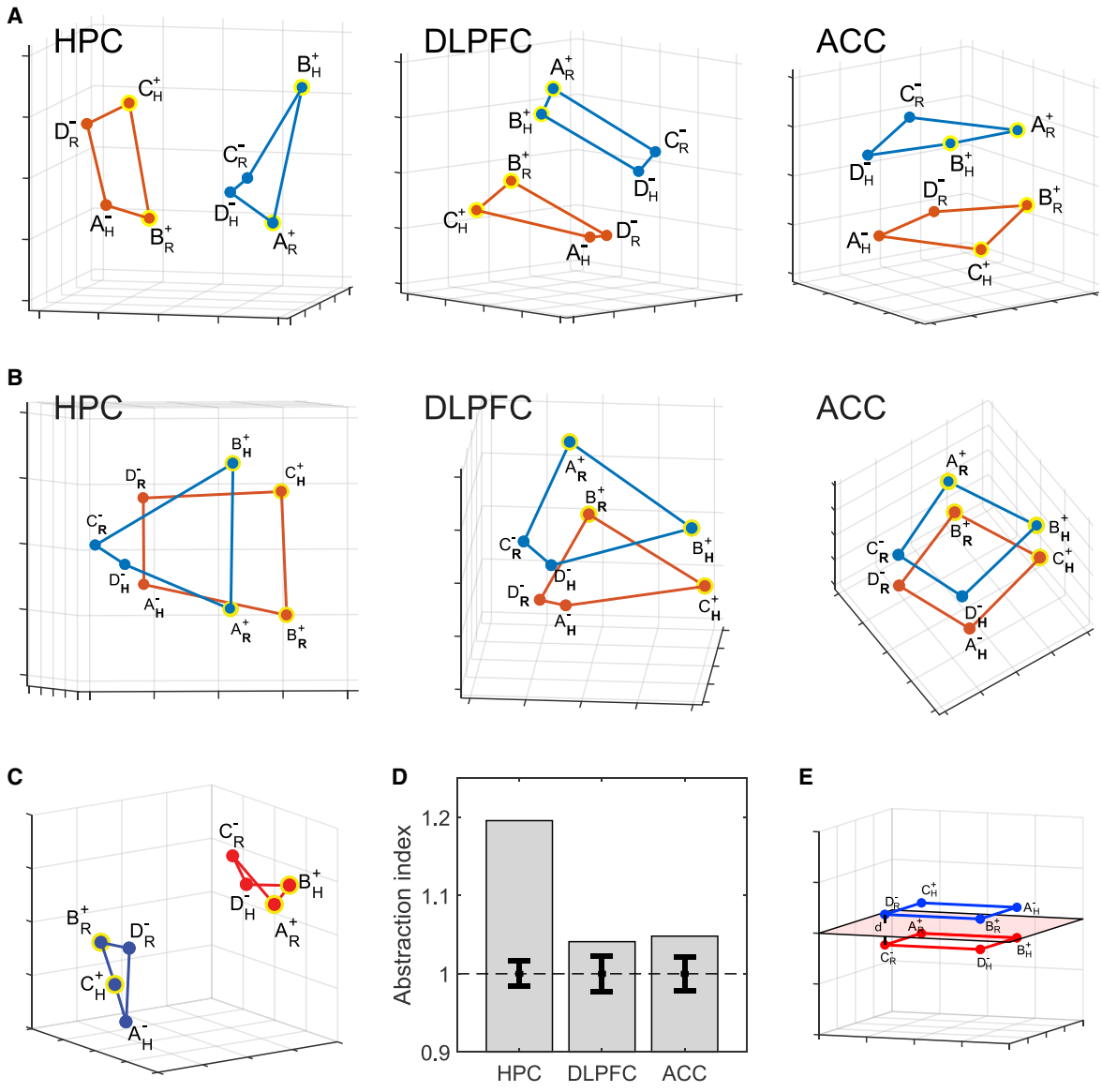
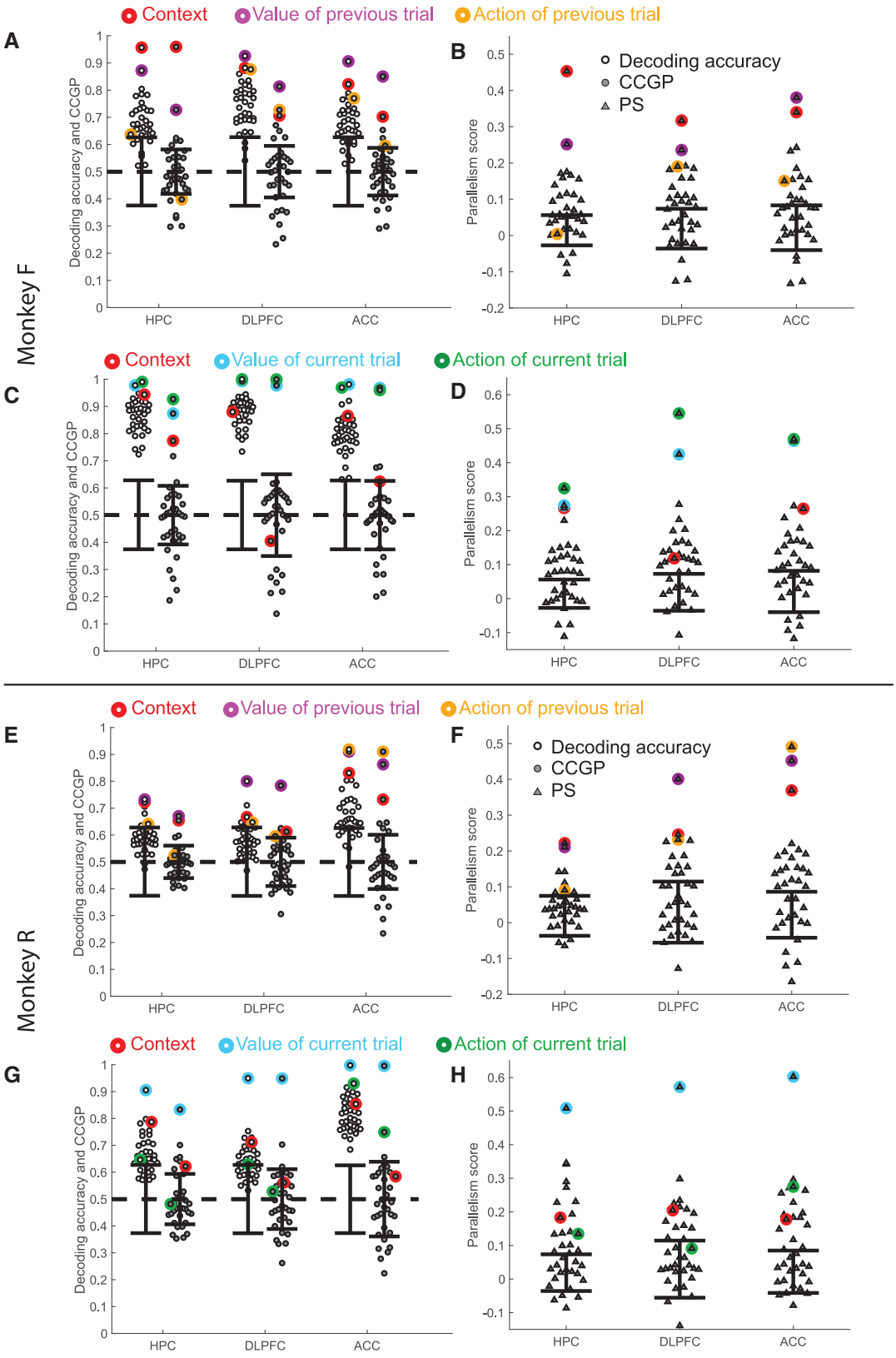


Figure S4. The Geometry of the Recorded Neural Representations and Abstraction by Clustering: Geometry and Measures of the Degree of Abstraction, Related to Figures 2 and 3

a. Multi-dimensional scaling plots (using Euclidean distances on z-scored spike count data in the 900 ms time window that starts 800ms before stimulus onset) showing the dimensionality-reduced firing rates for different experimental conditions in the three brain areas recorded from: HPC, DLPFC and ACC. The labels refer to value $(+/-)$ and operant action (R/H) corresponding to the previous trial. Yellow rings are rewarded conditions. While there is a fairly clean separation between the two context sub-spaces, other variables are encoded as well and the representations are not strongly clustered. Note that the context sub-spaces appear to be approximately two-dimensional (i.e., of lower dimensionality than expected for four points in random positions). See also the videos in the Supplemental Material. b. The MDS plots of panel (a) are rotated to highlight the dependence on the action of the previous trial (H = Hold, R = Release; in boldface). In DLPFC and ACC the points corresponding to the conditions with the same action are on the same side of the plot. In these areas, action is decodable and in an abstract format. In HPC all the 8 points are distinct, and action is decodable. However, it is not in an abstract format. Indeed, the H and R points are along two almost orthogonal diagonals. Notice that value $(+/-)$ symbols) and context (red/blue; less clear from this perspective) are in an abstract format in all three brain areas. c. Schematic of firing rate space in the case of clustering abstraction. Due to the clustering, the average within-context distance is shorter than the mean between-context distance. d. Abstraction index for the context dichotomy (ratio of average between-context distance to average within-context distance using a simple Euclidean metric) for the z-scored neural firing rates recorded from HPC, DLPFC and ACC, averaged over a time window of −800ms to 100ms relative to stimulus onset. The error bars are plus/minus two standard deviations around chance level (unit abstraction index), obtained from a shuffle of the data. Notice that the abstraction index based on clustering for DLPFC and ACC is barely different from chance. e. An example in which the clustering analysis fails to detect that context is in an abstract format. The firing rate space is represented as in panel b. The points are arranged in a rectangular cuboid (a cube squeezed along the

(legend continued on next page)

vertical direction), and the distance between the two squares that represent the two contexts is *d*. This distance is smaller than the sides of the two squares. When *d* is sufficiently small, the abstraction index is one or less than one, despite the fact that context is in an abstract format according to the CCGP (see the separating hyperplane which illustrates one linear classifier that clearly allows for cross-condition generalization). This type of geometry (the squeezed cube) may be observed whenever variables are encoded with different strengths (see also text in this section).

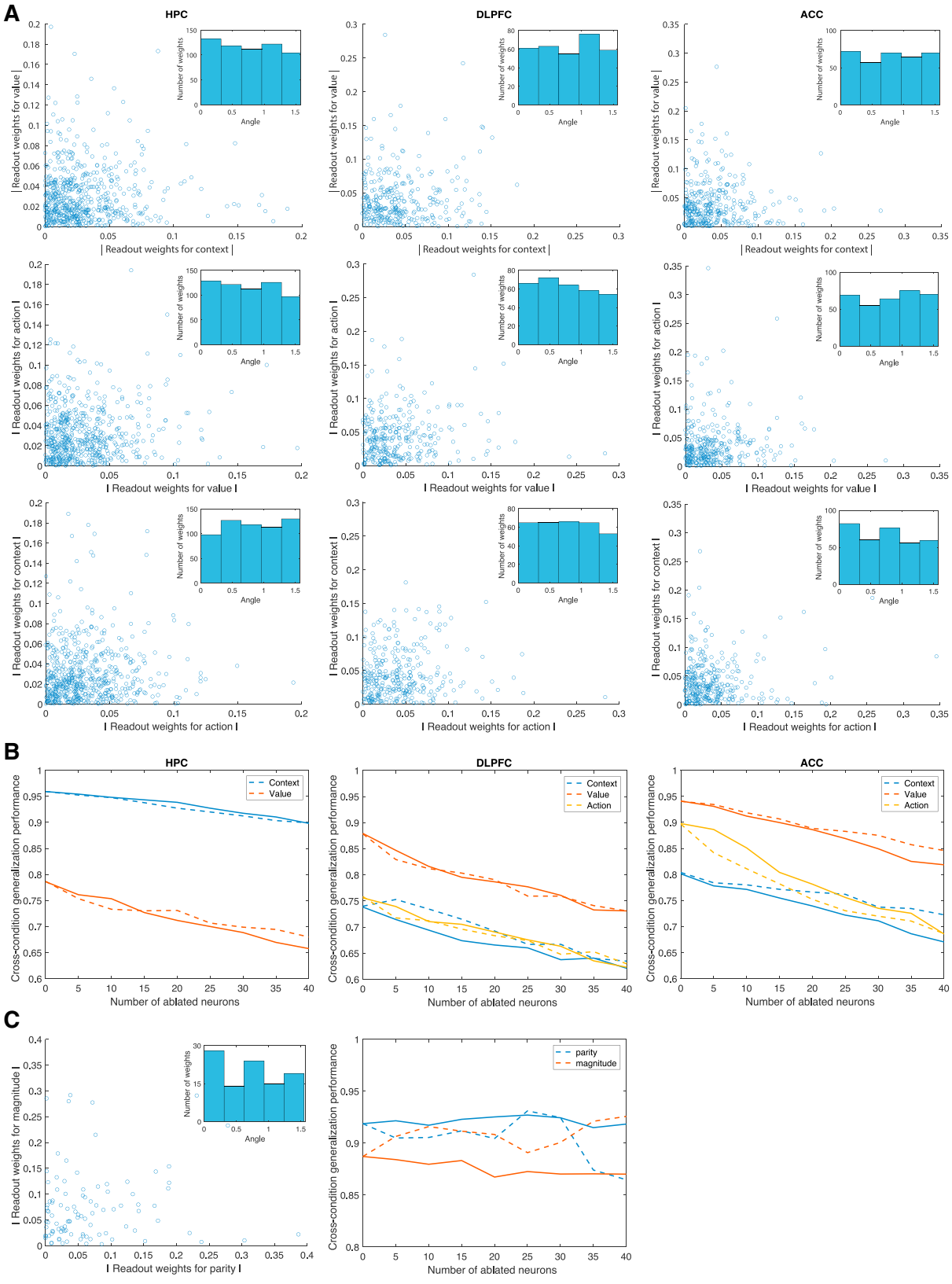


(legend on next page)

Figure S5. Decoding Accuracy, CCGP, and Parallelism Score in the Three Recorded Brain Areas in Monkey F and Monkey R Separately, Related to Figures 3 and 5

a-b,e-f: CCGP, decoding accuracy and PS for the variables that correspond to all 35 possible dichotomies shown separately for each brain area in a 900 ms time epoch beginning 800 ms before image presentation. a-b is for monkey F and e-f for monkey R. The points corresponding to the context, value and action of the previous trial are highlighted using circles of different colors. c-d,g-h. CCGP, decoding accuracy and PS for all 35 dichotomies in the time interval from 100ms to 1000ms after stimulus onset. c-d for monkey F and g-h for monkey R. For all panels, error bars are $\pm$ two standard deviations around chance level as obtained from a geometric random model (CCGP) or from a shuffle of the data (decoding accuracy and PS). Almost all dichotomies can be decoded with accuracy > 0.5 in all 3 brain areas, and the average decoding performance across the 35 dichotomies (shattering dimensionality) is significantly greater than chance ($p < 0.01$, permutation test). Nonetheless, multiple variables are represented in an abstract format simultaneously, with CCGP well above the confidence bounds around chance levels.

Overall, although the decoding performance is weaker in Monkey R than Monkey F, perhaps in part accounted for by the smaller sample size, the basic results concerning how CCGP reveals that multiple variables are represented in an abstract format simultaneously are evident in both experimental subjects.



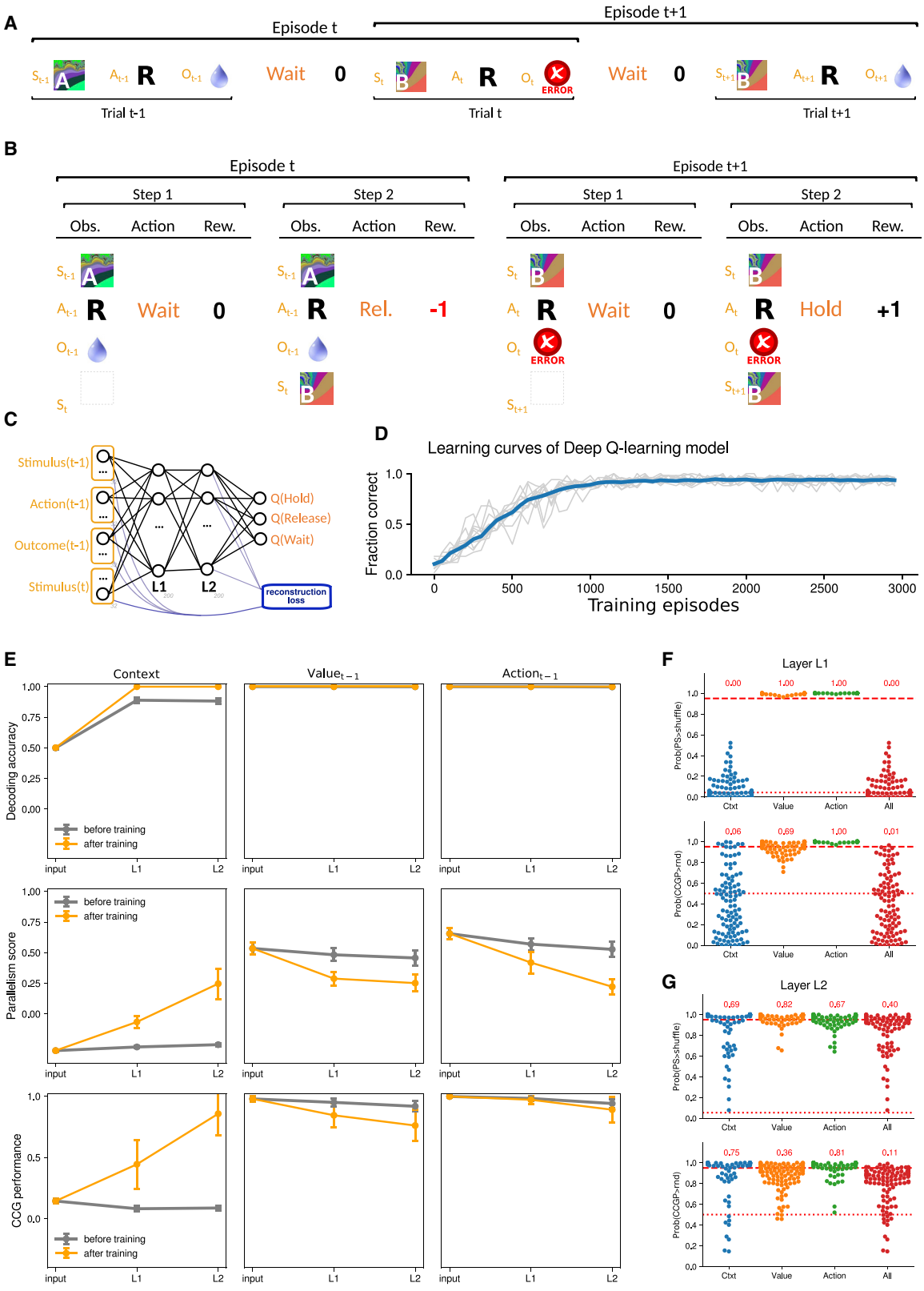
(legend on next page)

Figure S6. Neurons Are Rarely Specialized and the Ability to Cross-generalize Does Not Rely Mainly on the Few Specialized Neurons that Are Observed, Related to Figures 3, 4, 5, and 7

a. Two-dimensional scatterplots of the absolute values of the (normalized) decoding weights for the three task-relevant variables. The three columns (from left to right) correspond to HPC, DLPFC, and ACC. The three rows show the magnitudes of the weights for pairs of variables plotted against each other: context versus value (top), value versus action (middle), and action versus context (bottom). The inset in each scatterplot shows a histogram of the weight counts as a function of the angle from the vertical axis (in radians). These distributions are approximately uniform, and therefore pure selectivity neurons (whose weights would fall close to one of the axes in the scatterplots) are not prevalent. Similar distributions have been observed in the rodent hippocampus (Stefanini et al., 2020).

b. CCGP as a function of the number of ablated neurons for the HPC (left), DLPFC (middle), and ACC (right). The solid lines show the decay of CCGP if we successively remove the neurons with the largest pure selectivity indices for context (blue), value (red) or action (yellow). The dashed lines show the decline of the CCGP for the same three variables if we instead ablate neurons with the largest sum of squares of their three decoding weights (i.e., those with the radial position furthest from the origin in their three-dimensional weight space), independent of their pure selectivity indices. The two sets of curves are rather close to each other, and thus these two sets of ablated neurons are of similar importance for CCGP. (For HPC, the CCGP of the action variable is always below chance level for both curves; not shown). This is similar to what has been observed in simulations of deep networks (Morcos et al., 2018).

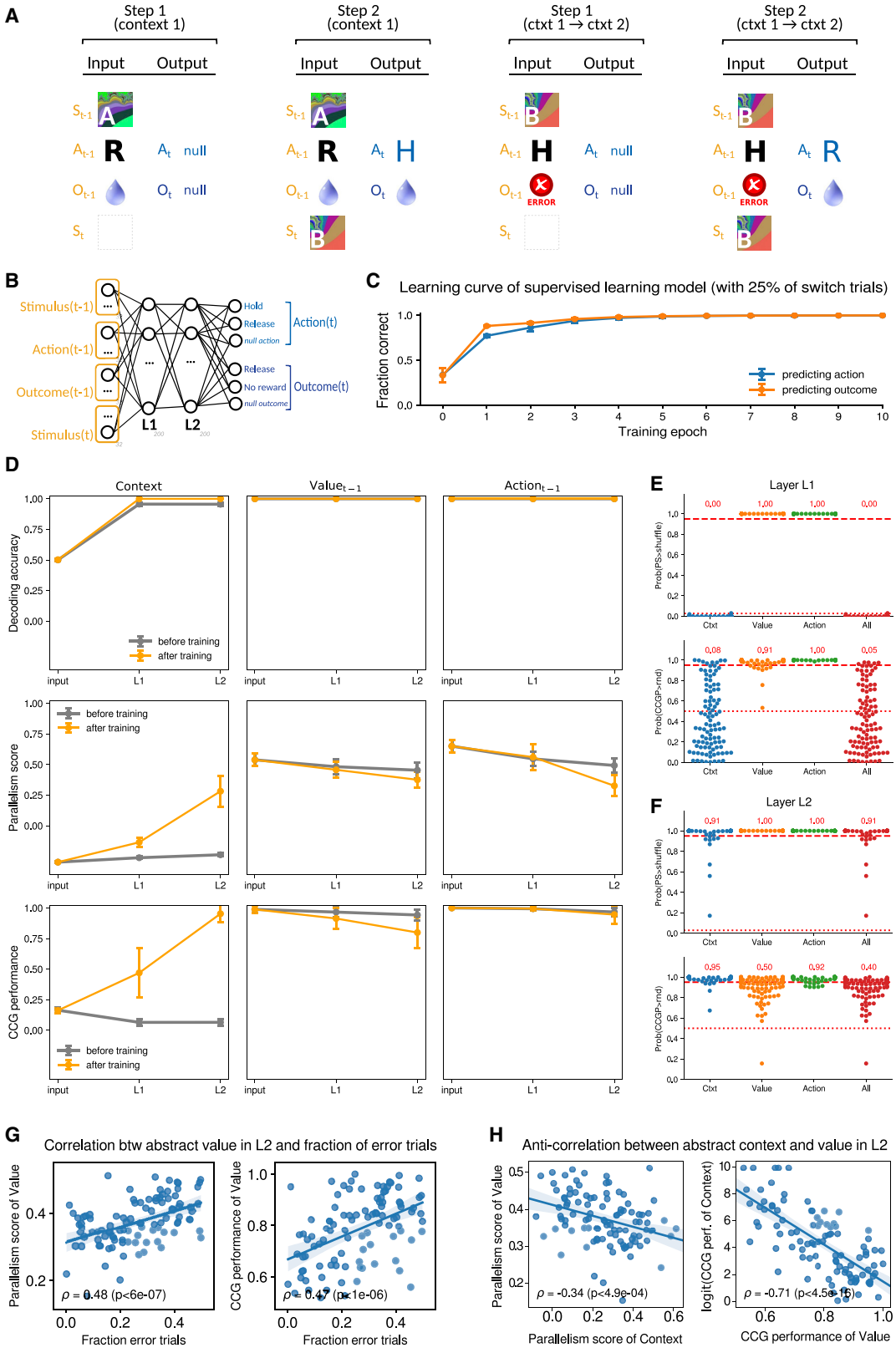
c. Specialization and ablation analyses of second hidden layer in the neural network of Figure 7. Left: Two-dimensional scatterplot of the absolute values of the (normalized) decoding weights for the parity and magnitude dichotomies, as in panel a. Right: CCGP when training on three digits from either side of the magnitude (red) and parity (blue) dichotomies and testing on the fourth one as a function of the number of ablated neurons. We ablate the neurons with the largest pure selectivity indices (solid lines), or the ones with the largest sum of squared decoding weights (dashed lines), as in panel b. Note that even though we don't z-score the neural responses for computations of the CCGP, or for any of the analyses shown in Figure 7, the decoding weights shown here (and used to select the ablated neurons) are taken from classifiers trained on z-scored responses so that the weights reflect the relative importance of different neurons for decoding.



(legend on next page)

Figure S7. Reinforcement Learning Model Trained with Deep Q-Learning on a Simulated Version of the Serial Reversal-Learning Task, Related to Figure 7

a., b. An episode in the simulated environment is composed of two sequential steps, each corresponding to one trial. In the first step, corresponding to the pre-stimulus interval in the trial, the network receives as observations the stimulus, action and outcome of the previous trial, at which point it is required to issue the action "Wait" to initiate a new trial. In the second step of the episode the network receives the current stimulus (in addition to all observations presented in the first step) and is required to issue the correct action in the task ("Hold" or "Release"). If at any step the agent issues an incorrect action, it receives a reward of −1 and the episode terminates. Otherwise, it receives a reward of 0 after correctly issuing "Wait" at the end of the first step, and for trials in which the agent issues the correct action, it receive a reward corresponding to the value of the stimulus in the task at the end of the second step (+1 or 0 depending upon whether it is a rewarded trial type). c. The agent is parametrized as a two-hidden layers neural network trained with DQN. We add a reconstruction loss to the optimization loss (indicated in blue in the lower right corner of the panel) that forces the network to find representations in the L2 hidden layer that can linearly reconstruct the inputs (i.e., the reconstruction loss is the mean squared error between the inputs and a linear transformation of the L2 activations). d. Learning curves for the DQN model on the serial-reversal task. The blue line is the fraction of correct actions averaged over 100 random initializations of the network and blocks of 50 episodes. The gray lines show the performance of 10 randomly chosen individual networks (also in blocks of 50 episodes). e. Analysis of the activity generated by the Reinforcement Learning model: *decoding accuracy* (first row), *parallelism score* (second row) and *cross-condition generalization performance* (third row) during the simulated first step (pre-stimulus epoch) in the task, for the variables *context* (first column), *previous value* (second column) and *previous action* (third column). In each panel we plot one of the quantities of interest as a function of the layer where it is measured in the architecture: inputs, first hidden layer (L1) and second hidden layer (L2) (see Panel c). The plots show the mean quantity of interest across 100 randomly initialized and independently trained networks, before (gray data points) and after training (orange data points). Error bars indicate standard deviations computed over the same distribution of 100 networks. f,g. Beeswarm plots of the probability of PS and CCGP of trained Q-learning models in layers 1 (f) and 2 (g) to be above the null-distributions given by the geometric random model and a shuffle test of the data (empirically estimated with 1000 samples per model), respectively (see Figure 3). Every dot in the graph represents a trained model. The top panels of f,g show the probability that the Parallelism Score of each model for every variable individually (Context, Value and Action) and all variables simultaneously (All) is above the null-distribution given by a shuffle of the data in layer L1 and L2. The bottom panels are laid out similarly, but show the probability that the CCGP of each model is above the null-distribution given by the geometric random model. The dotted lines correspond to the mean of the null-distribution, and the dashed lines correspond to the 95th percentile of the null-distribution. The numbers written in red above each plot report the fraction of models that are above the 95th percentile threshold.



(legend on next page)

Figure S8. Supervised Learning Model, Related to Figure 7

a. The supervised learning model is trained on two types of input-output combinations, corresponding to sequential steps in a trial. The inputs for the first step (the pre-stimulus interval in the trial) are composed of the stimulus, the action and the outcome (reward) in the previous trial. The target output for the first step is a 'null' action and outcome value (corresponding to a third null action and a third null outcome neuron). Inputs corresponding to the second step encode the current stimulus (along with the features that were already presented in the first step), and the second step output encodes the correct action (Hold or Release) and outcome (Reward or No reward) in the trial. Besides being distinguished by whether they correspond to the first step or second step in the trial, generated inputs can be distinguished depending on whether or not they correspond to a switch trial. Inputs generated from non-switch trials (see first and second input-output combinations in the panel) define the correct action and subsequent outcome, under the assumption that the current context is the same as in the previous trial (encoded in the input). Inputs generated from switch trials (the first trial after a context switch) are characterized by outcome features encoding an error in the previous trial (see second and third input-output combinations in the panel), implying that the observed stimulus-action combination is incorrect and counterfactually defining the currently correct context and action. b. The neural network is a multi-layer network with two hidden layers (number of units indicated in the figure) trained with backpropagation in a supervised way to output the correct action and outcome. c. Learning curves for the (simultaneously learned) action and value prediction tasks at the end of the second step (corresponding to the response and value prediction of the animal) as a function of training episodes. Every epoch consists of 128 mini-batches of 100 noisy versions of the inputs just described. Data points are the means and standard deviations across 100 distinct random initializations of the network. d. Activity analysis: *decoding accuracy* (first row), *parallelism score* (second row) and *cross-condition generalization (CCG) performance* (third row) during the simulated first step (pre-stimulus epoch) in the task, for the variables *context* (first column), *previous outcome* (second column) and *previous action* (third column). In each panel we plot one of the quantities of interest as a function of the layer where it is measured in the architecture: inputs, first hidden layer (L1) and second hidden layer (L2). Each point in the plots represents the mean quantity of interest across 100 randomly initialized and independently trained networks, before training (gray data points) and after training (orange data points). Error bars indicate standard deviations computed over the same distribution of 100 networks. The lines connect points computed within the same training condition (before training or after training) in adjacent layers in the architecture. e.,f. Beeswarm plots of the probability of PS and CCGP of trained supervised learning models in layers 1 (e) and 2 (f) to be above the null-distributions given by the geometric random model and a shuffle test of the data (empirically estimated with 1000 samples per model), respectively (see Figure 3). Every dot in the graph represents a trained model. The top panels of e,f show the probability that the Parallelism Score of each model for every variable individually (Context, Value and Action) and all variables simultaneously (All) is above the null-distribution given by a shuffle of the data in layer L1 and L2. The bottom panels are laid out similarly, but show the probability that the CCGP of each model is above the null-distribution given by the geometric random model. The dotted lines corresponds to the mean of the null-distribution, and the dashed line corresponds to the 95th percentile of the null-distribution. The numbers written in red above each plot report the fraction of models that are above the 95th percentile threshold. g. Correlation between abstraction of value and fraction of switch trials in hidden layer L2 of the supervised learning model. Both PS and CCGP for value are positively correlated with the fraction of switch trials with statistically significant Pearson correlation coefficients ρ (t test). Notice that this model makes mistakes only at the switches. h. Anti-correlation between abstraction of value and abstraction of context in hidden layer L2 of the supervised learning model. Both PS and CCGP for value are negatively correlated with respectively the PS and CCGP for context. Since CCGP for context tends to saturate close to one for low CCGP for value, we applied a logit transform to the CCGP for context, before computing the Pearson correlation coefficients.