The geometry of abstraction in hippocampus and pre-frontal cortex

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

[1]*Department of Neuroscience, Columbia University*
[2]*Department of Psychiatry, Columbia University*
[3]*Research Foundation for Mental Hygiene*
[4]*Center for Theoretical Neuroscience, Columbia University*
[5]*Mortimer B. Zuckerman Mind Brain Behavior Institute, Columbia University*
[6]*Kavli Institute for Brain Sciences, Columbia University*
[7]*IBM Research AI*
[8]*New York State Psychiatric Institute*
[9]*Current address: Institut du Cerveau et de la Moelle Epiniére (UMR 7225), Institut Jean Nicod, Centre National de la Recherche Scientifique (CNRS) UMR 8129, Institut Étude de la Cognition, École normale supérieure*

* These authors contributed equally, † co-senior authors

Abstraction can be defined as a cognitive process that finds a common feature - an abstract variable, or concept - shared by a number of examples. Knowledge of an abstract variable enables generalization, which in turn allows one to apply inference to new examples based upon old ones. Neuronal ensembles could represent abstract variables by discarding all information about specific examples, but this allows for representation of only one variable. Here we show how to construct neural representations that encode multiple abstract variables simultaneously, and we characterize their geometry. Representations conforming to this geometry were observed in dorsolateral pre-frontal cortex, anterior cingulate cortex, and the hippocampus in monkeys performing a serial reversal-learning task. These neural representations allow for generalization, a signature of abstraction, and similar representations are observed in a simulated multi-layer neural network trained with back-propagation. These findings provide a novel framework for characterizing how different brain areas represent abstract variables, which is critical for flexible conceptual generalization and deductive reasoning.

High-level cognitive processing relies on the ability of the brain to represent information about abstract variables, such as concepts, contexts, and rules. Knowledge of these types of abstract variables enables one to use inference to generalize and immediately arrive at the value of an abstract variable characterizing a new example[1]. New examples often can be linked to multiple abstract variables; for example, a bottle of rare aged burgundy can be linked to the concept of ”valuable” and to the concept of ”drinkable”. The capacity to generalize across multiple abstract variables enhances cognitive and emotional flexibility, enabling one to adjust behavior in a

more efficient and adaptive manner. However, a conceptual framework and corresponding data for understanding how the brain represents simultaneously multiple variables in an abstract format - i.e., how the brain can link a single example to multiple concepts simultaneously - has been elusive.

One possibility is that in representing an abstract variable in a population of neurons, all information about the specific examples is discarded while retaining only the combination of features essential to the abstract variable. For example, the only information retained in an abstract format could be the feature that all the instances belonging to a conceptual set have in common. However, in this case, generalization applied to a new instance can only occur with respect to this encoded abstract variable. The capacity to link a new example to multiple abstract variables simultaneously promotes flexibility, but it would require neural populations to retain multiple pieces of information in an abstract format. To investigate whether and how variables are represented in an abstract format within a neural population, we targeted neurophysiological recordings to the hippocampus, dorsolateral pre-frontal cortex (DLPFC) and anterior cingulate cortex (ACC) while monkeys performed a serial reversal-learning task. In this task, monkeys utilized multiple task-relevant variables to guide their operant behavior and reinforcement expectation. The task involved switching back and forth between two contexts, where the sets of stimulus-response-outcome mappings (or contingencies) differed in each context. Knowledge of the variable context could be acquired by using the temporal statistics of events (the sequences of trial types within each context). We targeted the hippocampus because it has long been implicated in generating episodic associative memories [2–4] that could play a central role in creating and maintaining representations of variables in an abstract format. Indeed, studies in humans have suggested a role for the hippocampus in the process of abstraction [5]. We also targeted two parts of PFC due to its established role in encoding rules and other cognitive information [6–10]. Although signals representing abstract cognitive variables have been described in PFC [7,9–11], prior studies have not tested explicitly whether multiple variables are represented in an abstract format within a population of neurons.

Neurophysiological recordings showed that multiple task-relevant variables, including context, operant response, and reinforcement outcome, were represented simultaneously in an abstract format in hippocampus, DLPFC, and ACC. This abstract format was revealed by an analysis of the geometry of the representations, which is characterized by the arrangement of the points representing different experimental conditions in the firing rate space for all recorded neurons. In this firing rate space, the parallelism of the coding directions for multiple variables was significantly enhanced compared to a random unstructured geometry in which abstraction does not occur. The observed geometry also enables generalization across conditions within the recorded neural populations in all three brain areas, a signature of abstraction. A multi-layered neural network trained with back-propagation revealed a similar capacity for generalization that was related to the emergence of parallel coding directions in the geometry of the representations. These results provide a conceptual and mechanistic framework for understanding how the brain can relate a single example to multiple abstract variables simultaneously within a population of neurons.

Monkeys demonstrate utilization of inference to adjust their behavior We designed a serial-reversal learning task in which switches in context involve un-cued and simultaneous changes in operant and reinforcement contingencies for each of four images. In other words, two dis-

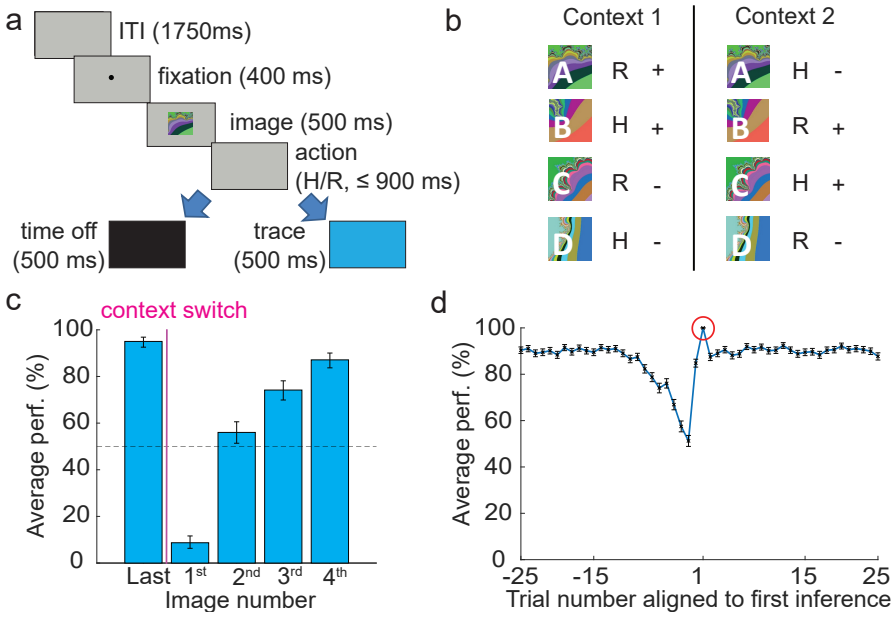


Figure 1: Task design and behavior. a. Sequence of events within a trial. A monkey holds down a press button, then fixates, and then views a fractal image (discriminative stimulus). A delay interval follows image viewing during which the operant response (hold or release the press button, respectively indicated as H or R) must be performed. A liquid reward is then delivered for correct responses to 2 of the 4 images. b. Task scheme. In each of the 2 contexts, correct responses result in reward for 2 of the images, and no reward for the other 2 images (plus or minus). Operant and reinforcement contingencies are unrelated, so neither operant action is linked to reward per se. Monkeys switch back-and-forth between contexts many times in each experiment. A different colored frame (red or blue) for each context appears on the edges of the monitor on 10 percent of the trials and only on specific image types (image C for context 1 and image D for context 2) although never in the first five trials following a contextual switch. c. Monkeys utilize inference to adjust their behavior. Average percent correct is plotted for the first presentation of the last image presented before a context switch (”Last”) and for the first instance of each image after the context switch (1-4). Binomial parameter estimate, bars are 95% Clopper-Pearson confidence intervals d. Average percent correct performance plotted as a function of trial number when aligning the data to the first correct trial where the monkey utilized inference (circled in red). Performance remains at asymptotic levels once evidence of inference is demonstrated.

tinct sets of stimulus-response-outcome mappings exist implicitly, one for each context. Correct performance for two of the stimuli in each context requires releasing a button after stimulus disappearance; for the other two stimuli, the correct operant response (action) is to continue to hold the button (Figure 1a,b). For half of the trials, correct performance results in reward delivery; for the other half of the trials, correct performance avoids having to repeat the trial but does not result in reward receipt (Figure 1b). Neither operant response is associated with reward, as the reward contingencies of the trials are orthogonal to the operant contingencies. Without warning, randomly after 50-70 trials, the operant and reinforcement contingencies switch to the other context; contexts switch many times within an experiment.

On average, the monkeys' performance drops to significantly below chance immediately after a context switch, as the change in contingencies is un-cued (see image number 1 in Fig. 1c). In principle, monkeys could simply re-learn the correct stimulus-action associations for each image independently after every context switch. Behavioral evidence indicates that this is not the case because the monkeys perform inference. After a context switch, as soon as they have experienced the changed contingencies for one or more stimuli, on average they infer that the contingencies have changed for the stimuli not yet experienced in the new context, as reflected by performance significantly above chance for these stimulus conditions where inference could be applied (see image numbers 2-4 in Fig. 1c). As soon as monkeys exhibited evidence of inference by performing correctly on a trial's first appearance after a context switch, the monkeys' performance was sustained at asymptotic levels for the remainder of the trials in a context (Fig. 1d).

The observation that monkeys can perform inference suggests that the different stimulus-action-outcome associations of the same context are somehow linked together. The observed behavior can be explained in at least two ways. First, monkeys could simply remember the stimulus-action-outcome of the previous trial and then use this information to select the action in response to the stimulus of the current trial. This strategy essentially uses memories of the sequences of trials that occur within the experiment, and it could explain all the aspects of the behavior described above, including inference. However, this strategy is not the most efficient in terms of memory resources, as it requires learning and storing 32 different trial sequences (4 stimuli multiplied by the 8 possible stimulus-action-outcome combinations of the previous trial). The second possibility entails that monkeys create a new abstract variable that pools together all the stimulus-response-outcome combinations (instances) that are present within each context, to create representations of the two contexts. This process of abstraction results in dimensionality reduction, as it reduces the number of entities to be remembered from 32 (in the first strategy) to 8 (4 stimuli that could be presented on the current trial multiplied by two contexts).

Decoding context and other task-relevant variables from neural activity Examination of the observed behavior itself is not sufficient to understand how the brain enables monkeys to perform this task, because, as we just discussed, the behavior is consistent with at least two strategies. To understand the neural mechanisms underlying the observed behavior, we first sought to determine which task-relevant variables were represented in the neuronal populations recorded. We measured the activity of 1378 individual neurons in the PFC and hippocampus in two monkeys while they performed our task. Of these, 629 cells were recorded in hippocampus (HPC, 407 and 222 from

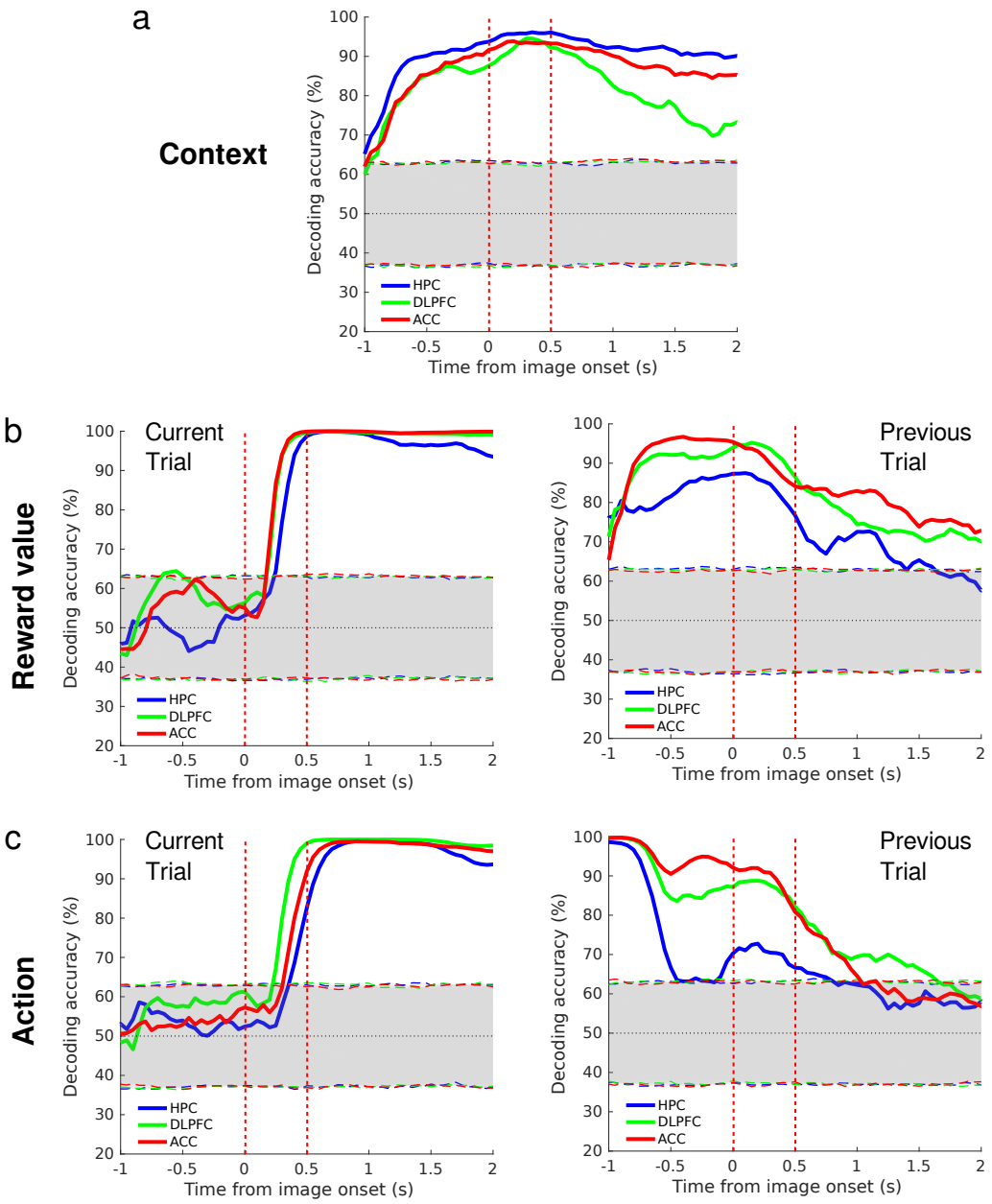


Figure 2: Population level encoding of task related variables. a-e. Performance of a linear decoder plotted a function of time relative to image onset for classifying a task-relevant variable. a. Context on the current trial. b. Reinforcement outcome on current (left) and prior (right) trials. c. Operant 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). Dashed lines around chance level indicate 97.5 percent confidence intervals obtained by shuffling trials 1000 times (bootstrap). The image is displayed on the screen from time 0 to 0.5 sec. Analyses were run only on correct trials at least 5 trials after a context switch.

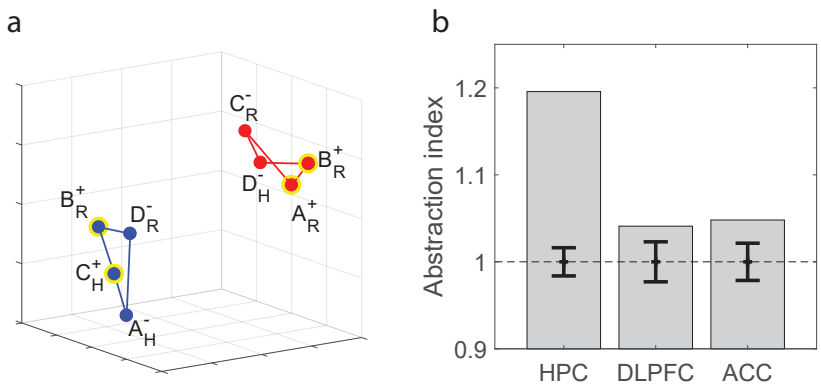


Figure 3: Abstraction by clustering. a) Schematic multi-dimensional scaling plot (MDS dimensionality reduction) of artificially generated data in which the mean firing rates corresponding to the eight experimental conditions are clustered according to context. The two contexts are indicated by the colors red and blue, while the eight conditions are labeled by the stimulus identity (A,B,C or D), the value of the stimulus in the present context (plus or minus), and the required operant action (R or H, for release or hold). Due to the clustering, the average within-context distance is shorter than the mean between-context distance. b) 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.

each of the two monkeys, respectively), 335 cells were recorded in ACC (238 and 97 from each of the two monkeys), and 414 cells were recorded in the DLFPC (226 and 188 from the two monkeys). We used a linear decoder applied to the populations of neurons recorded from each area separately to assess the degree to which task-relevant variables were represented (see Methods). Information about context was decoded with high accuracy in all three brain areas (Fig. 2a). In particular, context could be decoded in the time interval preceding the stimulus presentation, indicating that information about context was available as soon as an image appeared, which is when it is needed to make a decision about the operant action to perform and the reinforcement to expect. In the same time interval, it was also possible to decode the action and the value of the previous trial (Fig. 2b,c right panels), indicating that not only context is encoded, but also the specific instances of response and outcome from the last trial. Information about context is sustained in the representations throughout the current trial as well, but the representations of operant action and expected outcome for the current trial do not emerge until shortly after image onset (Fig. 2b,c left panels).

Clustering abstraction The capacity to decode context from a neuronal population does not imply that the representation is in an abstract format. For example, consider the first strategy we described, in which monkeys rely on what happened in the previous trial to decide their action. In this strategy, the neural activity in the interval preceding the visual stimulus must represent the sequence of events of the previous trial for which there were 8 possible combinations of stimulus,

action and reinforcement outcomes. There are many situations in which a simple linear decoder can decode context from the activity of a neuronal population. For example, when the 8 conditions correspond to 8 random patterns, it is very likely that for a sufficient number of neurons, the 4 conditions corresponding to one context are separable from the 4 conditions of the other context. This is true also in the case in which there is a cloud of points for each condition [12]. Hence, a simple linear decoder can extract the information about context from neural activity. Nevertheless, random representations are obviously not abstract, and they would not permit generalization across conditions.

To understand which features of the neural representations can enable generalization, it is instructive to consider the geometry of the firing rate space (see Fig. 3a for an example of simulated data). In this space, each coordinate axis is the firing rate of one neuron, and hence, the total number of axes is as large as the number of recorded neurons. To visualize this space, we will use a standard dimensionality reduction technique (multi-dimensional scaling, MDS). For each of the 8 conditions, we plot the simulated average firing rate in a 900 ms interval that starts 800ms before the visual stimulus, and determine the coordinates of the corresponding point. The geometry of the representation is defined by the arrangement of these points in the firing rate space. One simple way to achieve abstraction of context is to retain only the information about the context and discard the information about the specific instances that correspond to the particular combinations of stimulus, action and reinforcement outcome of the previous trial. In this case, the 4 points corresponding to context 1 would coincide in the firing rate space, or more realistically, in the presence of noise they would cluster around a single point. The other 4 points, for trials occurring in context 2, would constitute a different cluster (blue indicates context 1 and red context 2). As we will see later, this is not the only possible geometric format that can allow for abstraction, but it provides a representation that is disassociated from specific instances, since patterns of firing rates are similar for all the conditions within a context, despite the fact that these conditions differ for other variables (e.g. operant action or reinforcement outcome). Importantly, clustering leads to a geometric arrangement that permits generalization. Indeed, a readout trained to decode context from a small subset of clustered points will generalize right away to all the other points, if the noise is not too large. This is a fundamental property of abstraction that has already been discussed in [11] and that we will study in detail below.

The degree of clustering in a neural representation can be characterized by comparing the distances of points within a cluster to the distances of points across clusters for the points in the firing rate space that correspond to the 8 conditions. This method has been suggested in [5], where abstraction was studied in an fMRI experiment. The authors reported that in an experiment similar to this one, the intra-context distance was significantly shorter than the inter-context distance in the hippocampus (our ”contexts” are analogous to their ”communities”). We performed the same analysis on our data, focusing on the 900 ms interval that starts 800ms before image onset, and found that the difference between inter-context distances and intra-context distances is larger in the HPC than in DLPFC or ACC (Fig. 3b). In DLPFC and ACC the degree of clustering is only barely different from that predicted by a non-abstract random model, i.e., a situation in which the 8 points corresponding to the 8 conditions are at random locations in the firing rate space (see Methods for an exact definition). However, this analysis can be misleading. As we will show below, it is

possible to construct abstract representations in which the intra-context distances are comparable to the inter-context distances. Furthermore, some geometric arrangements offer the computational advantage that they can encode multiple abstract variables simultaneously. To illustrate this, we will first visualize the recorded representations using MDS and then, taking inspiration from these visualizations, we will construct neural representations that encode multiple abstract variables. Finally, we will show that the geometry of these representations conforms to the geometry of the observed neural representations.

Visualizing the geometry of recorded representations Neural activity was recorded from hundreds of neurons, and we visualized the firing rate space by using MDS to reduce the dimensionality of the data to three (see Methods for more details). Like other dimensionality reduction methods, MDS is a useful visualization tool, but it provides only an approximate depiction of the original high-dimensional data. In Figure 4 we show the MDS plots for all three brain areas, using the same notation as in Figure 3a. In the hippocampus (HPC) the red and the blue points, which represent the two contexts, are well separated, as expected from the clustering analysis. However, it is clear that also in this case the intra-context distances are not negligible and that the points within the clusters are nicely organized (e.g. the rewarded and non-rewarded conditions are well separated – this organization is particularly evident in the movies in the Supplementary Material, in which these plots can be viewed from many different angles). This type of structure is even more prominent in the DLPFC and ACC, where the intra-context distances are comparable to the inter-context distances. Moreover, the movies in the Supplementary Material suggest that the four points of each context are contained in a low-dimensional subspace, almost a plane. The planes corresponding to the two contexts are approximately parallel. These plots suggest that there might be a different geometry that underlies abstraction and is not captured by clustering. Taking inspiration from the plots of Figure 4, we now construct a simple geometry in which an abstract variable can be encoded without clustering and hence without sacrificing the possibility of encoding other variables. This construction can be extended to the case of multiple variables encoded in an abstract format.

Beyond clustering: constructing neural representations that encode multiple abstract variables To construct neural representations that encode multiple abstract variables, it is useful to start from the simple example that we illustrate in Figure 5. These plots depict the geometry of a representation in the original firing rate space (not the MDS projection) where the neuronal population includes only three neurons, with each axis representing the activity of one neuron. In Figure 5a we constructed a geometry in which the firing rate f_3 of the third neuron in the interval preceding image onset depends only on context and not the stimulus identity, operant action or value of the previous trial. This explains why the points of the two contexts lie on two parallel planes that are orthogonal to the 3rd axis. The other two neurons encode the other task-relevant variables as strongly as the third neuron encodes context. As a result, intra-context distances are comparable to inter-context distances. Nevertheless, an abstract representation of context is clearly embedded in this geometry because the third neuron encodes only context and throws out all other information.

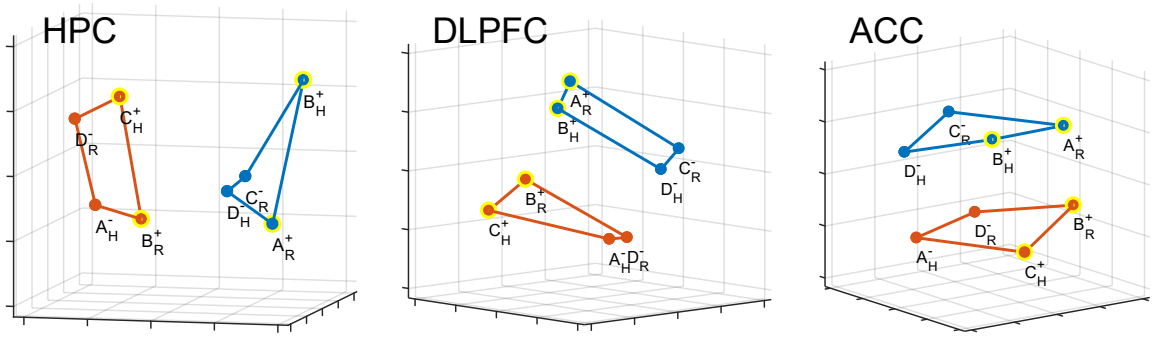


Figure 4: The geometry of neural representations. Multi-dimensional scaling plots (using Euclidean distances on z-scored spike count data in the 900 ms time window that starts 800ms before stimulus onset) showing the dimensionality-reduced firing rates for different experimental conditions in three brain areas we recorded from: HPC, DLPFC and ACC. The labels are as in Fig. 3a, with value (+/-) and operant action (R/H) corresponding to the previous trial. While there is a fairly clean separation between the two context sub-spaces, clearly 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).

The simple example depicted in Fig. 5a does not reflect the geometry observed in our dataset, because the third neuron is assumed to encode context only, and we rarely observe that a neuron is so highly specialized (see Supplementary Information S1). However, we can preserve all the generalization properties of the representation of Figure 5a even when we rotate it (see Figure 5b). This means that context is still an abstract variable, even though all neurons may now respond to multiple task-relevant variables.

To construct a representation that encodes multiple abstract variables, we start from a representation similar to the one of Figure 5a in which neurons 1 and 2 are also specialized. For example, neuron 1 could respond only to the outcome and neuron 2 only to the action of the previous trial. Now consider the case in which this representation is rotated; now all neurons respond to more than one task-relevant variable, and, more specifically, they exhibit linear mixed selectivity [13,14] to context, operant action and reward value. Moreover, both action and value are also abstract, as illustrated in Figure 5c, where we show the exact same geometry of Figure 5b, but highlight how it encodes also the value of the previous trial in an abstract format. Indeed, all the points corresponding to the rewarded conditions are contained in the yellow plane, and the non-rewarded points are in the gray plane. These two planes are parallel to each other, just like the ones for the different contexts in Figure 5b. Using a similar construction, it is possible to represent as many abstract variables as the number of neurons in a population that can be read out. However, additional limitations would arise from the amount of noise that might corrupt these representations.

We will now show that the recorded neural representations are likely implementing an encoding strategy very similar to the one we have just illustrated. In order to demonstrate this, we revert to the original high-dimensional representations and show that the observed geometry in that

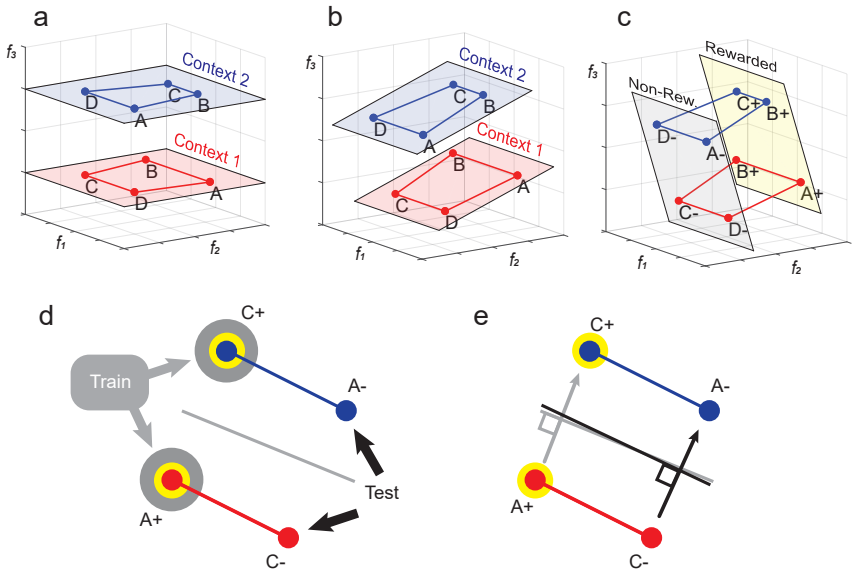


Figure 5: A neural code for multiple abstract variables. a. Schematic of the firing rate space of three neurons, of which one is specialized for encoding context (f_3 axis). The other neurons encode different variables. The points of each context are in one of the two low-dimensional manifolds (planes in this case) that are parallel. Neurons that are highly specialized to encode only context are rarely observed in the data (see Suppl. Info. S1). b. The same neural representation geometry as in a, but rotated in firing rate space, leads to linear mixed selectivity. Even though there are no longer any neurons specialized to encode only context, in terms of decoding as well as generalization using linear classifiers, this case is equivalent to that shown in panel a. c. The same neural geometry as b, but with planes depicted that highlight the encoding of reward value. Data points corresponding to the same value fall within a plane, just as data points corresponding to the same context fall within a plane in the previous panel. d. Schematic explanation of cross-condition generalization (CCG) in the simple case of only four experimental conditions, labeled according to context (red versus blue) and value of the stimulus (a yellow ring indicates a rewarded condition). We can train a linear classifier to discriminate context on only two of the conditions (one from each context, in the case shown the rewarded conditions), and then test its generalization performance on the remaining conditions not used for training (here the unrewarded conditions). The resulting test performance will depend on the choice of training conditions, and we refer to its average over all possible (in this case four) ways of choosing them as the cross-condition generalization performance (CCGP). A CCGP that is above chance level would indicate that the neural representation of a given variable is in an abstract format because it enables generalization. e. Schematic explanation of the parallelism score (PS). Training a linear classifier on the two rewarded conditions leads to the gray separating hyperplane which is defined by a weight vector orthogonal to it. Similarly, training on the unrewarded conditions leads to the black hyperplane and weight vector. If these two weight vectors are close to parallel, the corresponding classifiers are more likely to generalize to the other conditions not used for training. In the case of isotropic noise around the two training conditions, these weight vectors will be proportional to the (context) coding vectors connecting the mean neural activities of the training conditions across the context divide. Therefore, instead of training classifiers we can look directly at the angle between these coding vectors, and define a parallelism score as the cosine of the angle between the coding vectors (maximized over all possible ways of pairing up the conditions; see Methods for details and generalization to eight conditions).

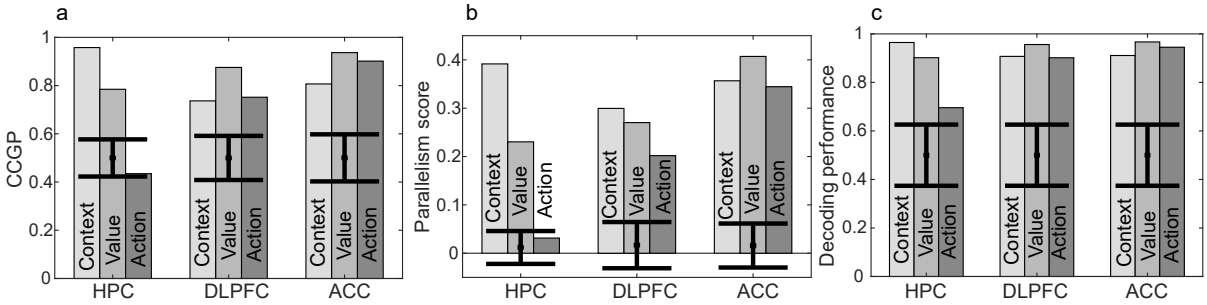


Figure 6: Neural representations of multiple abstract variables simultaneously. Data from all three brain areas reveal that the cross-condition generalization performance (CCGP) for a variable reflects the parallelism score (PS) that describes the geometry of neural representations. a. CCGP for the context, value and action of the previous trial for all three brain areas from which we recorded. Note that CCGP does not only rely on the rarely observed specialized neurons that encode only one variable (see Fig. S2). b. PS for the same variables. Note that according to both of these measures context and value are abstractly represented in all three brain areas, but action is abstract only in prefrontal cortex. c. Decoding performance of the three variables using maximum margin linear classifiers on the average firing rates in the same time interval. Even though the operant action is not abstractly represented in the hippocampus, it can still be decoded with a cross-validated performance significantly above chance level by a simple linear classifier. All data presented were obtained from correct trials in a -800ms to 100ms time window relative to stimulus onset. See Methods for details of the selection criteria for trials used and for neurons retained for these analyses. All error bars are $\pm$ two standard deviations around chance level as obtained from a geometric random model (panel a) or from a shuffle of the data (panels b and c).

space provides a means of representing multiple abstract variables simultaneously.

Cross-condition generalization as a signature of abstraction and the parallelism score Figure 5d provides an example for how the geometry of neural representations can be related to the ability of a linear readout to easily generalize for multiple variables simultaneously (context and value), which is a fundamental property of abstraction. To illustrate how a linear readout can generalize, consider for simplicity only a subset of four of the eight conditions in our experiments, where two trial types come from each of contexts 1 and 2, and only one of the trial types in each context is rewarded. We can train a decoder to classify context only on the conditions in which the monkey received a reward in the previous trial. Thanks to the arrangement of the four points, the resulting hyperplane (gray line in Figure 5d) successfully classifies context when testing on the other two conditions, in which the monkey did not receive reward. This corresponds to generalization, and it is a signature of abstraction. In other words, if a decoder is trained on a subset of conditions, and it immediately generalizes to other conditions, without any need for retraining, we conclude that a variable is represented in an abstract format (one that enables generalization). In order to determine whether the data exhibits the geometry of Figure 5, which supports generalization and therefore abstraction, we can directly test the ability to generalize by following the same procedure illustrated in Figure 5d: we can train a decoder on a subset of conditions and test it on the other conditions. We define the performance of the decoder on these other conditions as the

cross-condition generalization performance (CCGP).

We hypothesized that a specific aspect of the geometry of neural representations may account for generalization performance: the degree to which the coding directions determined when training a decoder are parallel for different sets of training conditions. Consider the case depicted in Figure 5e. Here we draw the two hyperplanes (which are lines in this case) obtained when a decoder is trained on the two points on the left (the rewarded conditions, gray) or on the two points on the right (unrewarded conditions, black). The two lines representing the hyperplanes are almost parallel, indicating that this geometry will allow good generalization regardless of which pair of points we train on.

One way to estimate to what extent these hyperplanes are aligned is to examine the coding directions (the arrows in the figure), which are orthogonal to the them. For good generalization, these coding directions should be as close to parallel as possible. This is the main idea behind the parallelism score (PS), a measure described in detail in the Methods. A large PS indicates a geometry likely to permit generalization and therefore the corresponding variable would be represented in an abstract format. When multiple abstract variables are simultaneously represented, the PS should be large for all variables, constraining the points to approximately define a geometry of the type described in Figure 5a,b. As the PS focuses on parallelism between coding directions, it can detect the existence of abstract variables even when the neurons are not specialized, or, in other words, when the coding directions are not parallel to the coordinate axes.

In Figure 6 we report both the CCGP and PS measured in the three areas during the 900 ms time interval that starts 800 ms before the presentation of the visual stimulus (see Methods for more details). The CCGP analysis reveals that context is abstract in all three areas, and the level of abstraction is more comparable across brain areas than suggested by the analysis shown in Figure 3b, where the abstraction index for DLPFC and ACC was more similar to values computed from a random model. In fact, all three variables are represented in an abstract format in all three areas, except the action of the previous trial in the hippocampus. Interestingly, the action can be decoded in HPC (see Figure 6c), even if it is not abstract. Remarkably, the PS exhibits a pattern very similar to the CCGP, indicating a direct correspondence between the geometry of representations and generalization. In conclusion, this analysis shows that multiple abstract variables are encoded in the populations of neurons recorded from each of the brain areas that we recorded from. Moreover, the geometry of these representations is similar to the one that we described in the previous section.

Abstraction in multi-layer neural networks trained with back-propagation Next we asked whether a simple neural network model trained with back-propagation would exhibit the same geometry as observed in the experiments. Back-propagation algorithms are popular in machine learning and have proven successful in many real world applications. We trained a two layer network (see Figure 7a) 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 (< 5) (Figure 7b). We wanted to test whether the learning process would lead to abstract representations of two concepts: parity and magnitude (i.e., large or small). This abstraction process is similar to the one studied in the

experiment in the sense that it involves combining together inputs that are visually very dissimilar (e.g. the digits '1' and '3', or '2' and '4'). Analogously, in the experiment, very different sequences of events (visual stimulus, operant action and value) are combined together into what we defined as contexts.

After training the network, we presented inputs that were not used for training, and we 'recorded' the activity of the two hidden layers. The multidimensional scaling plots, similar to those of Figure 4 for the real data (but reduced to two dimensions), are shown in Figure 7c for the input layer and for the two hidden layers of the simulated network. Each digit in these plots represents a different input. They are colored according to the parity/magnitude task illustrated in Figure 7b. While it is difficult to detect any structure in the input layer (the slight bias towards red on the left side is mostly due to the similarity between '1's and '7's), in the second hidden layer we observe the type of geometry that would be predicted for a neural representation that encodes two abstract variables, namely parity (even digits on the left, odd digits on the right), and magnitude (large at the top, small at the bottom). The digits tend to cluster at the four vertices of a square, which is the expected arrangement.

Just as in the experiments, we computed both the CCGP and the PS. We analyzed these two quantities for all possible dichotomies of the eight digits, not just for the dichotomies corresponding to magnitude and parity. This corresponds to all possible ways of dividing the digits in two equal size groups. In Figure 7d,e we ranked these dichotomies according to their CCGP and their PS, respectively. The largest CCGP and PS correspond to the parity dichotomy, and the second largest values correspond to the magnitude dichotomy (circles marked by crosses in Figure 7d,e). For these two dichotomies, both the CCGP and the PS are significantly different from those of the random models. There are other PS values that are significant. However, they correspond to dichotomies whose labels are correlated with one or both of the two trained dichotomies. If one restricts the analysis only to the dichotomies orthogonal to both of them (filled circles in Figure 7d,e), none are significantly above chance level. This analysis shows that the geometry of the neural representations in the simulated network is similar to that observed in the experiment. Furthermore, the CCGP and the PS can identify the dichotomies that correspond to abstract variables even when one has no prior knowledge about these variables. Indeed, it is sufficient to compute the CCGP and the PS for all possible dichotomies to discover that parity and magnitude are the abstract variables in these simulations.

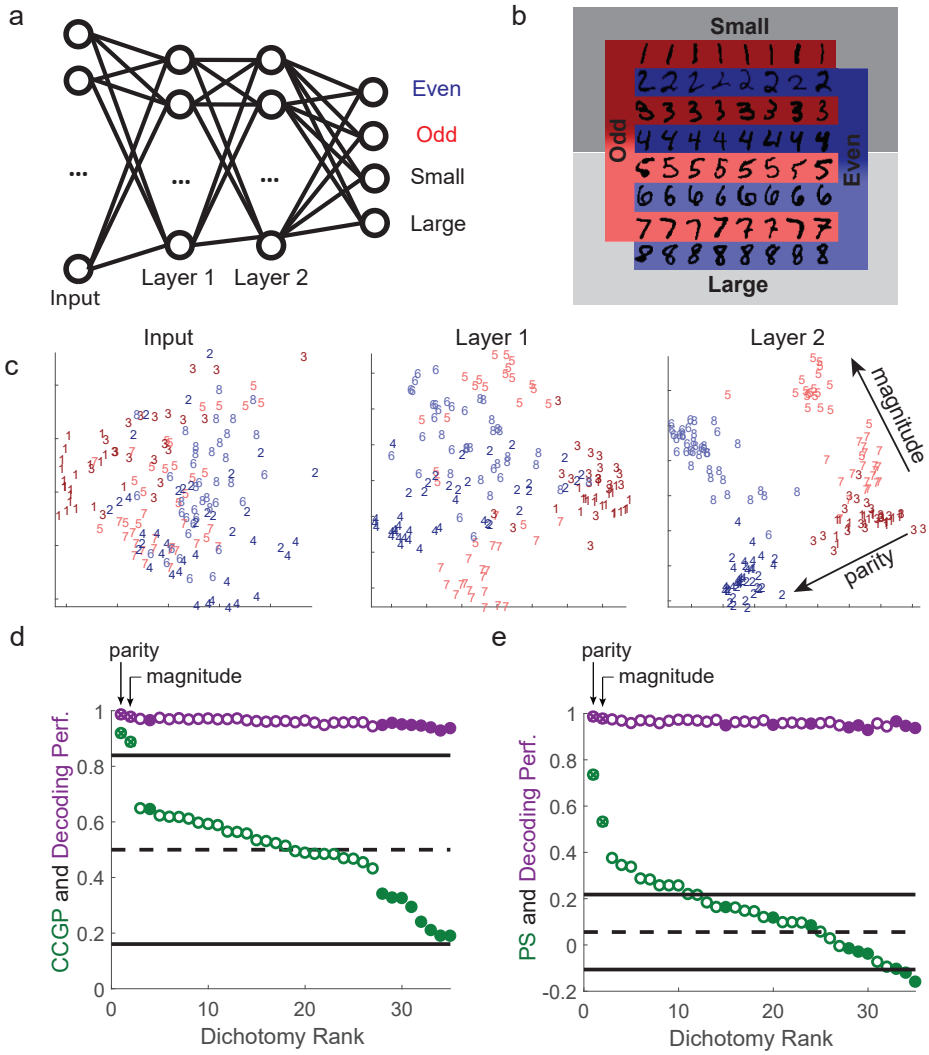
Discussion The cognitive process that finds a common feature - an abstract variable - shared by a number of examples or instances is called abstraction. Abstraction enables one to utilize inference and deduce the value of an abstract variable when encountering a new example. Here we developed a general method for determining when a variable is represented in an abstract format. We constructed neural representations that allowed multiple variables to be represented in an abstract format simultaneously. These representations are characterized by a specific geometry within the firing rate space. This geometry can be recognized by measuring either one of two quantities: the cross-condition generalization performance, which is directly related to the ability of a linear readout to generalize, and the parallelism score, which considers the angles between coding directions for any given variable.

In our experiments, monkeys performed a serial reversal learning task in which they switch back and forth between two contexts. Both cross-condition generalization and the parallelism score revealed that the task-relevant variable "context" is represented in an abstract format in HPC, DLPFC and ACC. Moreover, multiple abstract variables were represented simultaneously, as our measures of abstraction revealed that all the recorded brain areas actually represent at least two abstract variables (context, action and reward value of the previous trial in DLPFC and ACC, and context and value of the previous trial in HPC). We then showed that simple neural network models trained with back-propagation or with reinforcement learning algorithms exhibit the same neural geometry that we observed in the data, suggesting that this geometry may be a general feature underlying how abstract variables are represented in the brain.

During the performance of the serial reversal learning task, the provision of a neural representation of context when a stimulus appears enables monkeys to know which operant action to perform. The value and action of the previous trial are also represented in all three brain areas, but they are actually not needed for the next trial if the animal did not make a mistake. However, at the context switch, the reward received on the previous trial is the only feedback from the external world that indicates that the context has changed. Therefore, reward value is essential when adjustments in behavior are required. Moreover, monkeys occasionally make mistakes that are not due to a context change. To discriminate between these occasional errors and those due to a context change, information about value is not sufficient and information about the previously performed action could be essential for deciding the motor response on the next trial. Thus there is a clear benefit in retaining information about the value and action of the previous trial, and we find that

---

Figure 7 *(following page)*: Simulations of a multi-layer neural network reveal that the geometry of the observed neural representations can be obtained with a simple model. a. Diagram of the network architecture. The input layer receives gray-scale images of MNIST handwritten digits with 784 pixels. The two hidden layers have 100 units each, and in the final layer there are two pairs of output units corresponding to two binary variables represented by a concatenation of two one-hot vectors. b. Schematic of the two discrimination tasks. The network is trained using back-propagation to simultaneously classify inputs (we only use the images of digits 1-8) according to whether they depict even/odd and large/small digits. The colors indicate the parity and the shading the magnitude of the digits (darker for smaller ones). c. Two-dimensional MDS plots of the representations of a subset of images in the input (pixel) space, as well as in the first and second hidden layers. While 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, which suggests a simultaneously abstract representation for both variables. d. Cross-condition generalization performance (CCGP, green) for the variables corresponding to all possible balanced dichotomies when the second hidden layer is read out. The dichotomies are ranked according to the strength of their CCGP. Only the two dichotomies corresponding to parity and magnitude are significantly different from a geometric random model (chance level is 0.5 and the two solid black lines indicate plus/minus two standard deviations). The decoding performance (purple) is high for all dichotomies, and hence inadequate to identify the abstract variables. e. Same as panel d, but for the parallelism score (PS), with error bars obtained from a shuffle of the data. Both the CCGP and PS allow us to identify the correct abstract variables.


a
Even
Odd
Small
Large
Input
Layer 1
Layer 2
b
Small
Odd
Even
Large
c
Input
Layer 1
Layer 2
magnitude
parity
d
parity
magnitude
CCGP and Decoding Perf.
Dichotomy Rank
e
parity
magnitude
PS and Decoding Perf.
Dichotomy Rank

this information can also be represented in abstract format. Conceivably, these abstract representations may also afford the animal more flexibility in learning and performing other tasks. Consistent with this, previous work has shown that recent history is represented whether it is task-relevant or not (see e.g. [15,16]), even when it can degrade the performance of the animal [17]. This degradation may affect the specific task studied in the experiment, but the memory trace causing it might be beneficial in other scenarios that are closer to real-world tasks.

Our analysis showed that DLPFC and ACC represent more variables in an abstract format than hippocampus, as the action of the previous trial is in an abstract format only in DLPFC and ACC. This may reflect the prominent role of pre-frontal areas in supporting working memory (see e.g. [18–20]). Moreover, the fact that the hippocampus represents fewer variables in an abstract format as characterized by the parallelism score and cross-condition generalization explains why if one only considers clustering as a signature of abstraction, context is strongly identified as being in an abstract format only in the hippocampus (see Figure 3). However, our novel methods reveal that pre-frontal cortex also represents context in abstract format. In general, detecting abstract variables becomes more difficult as their number grows, since this increases the dimensionality of the sub-spaces encoding different values of each abstract variable. In this case, one therefore requires more samples in order to generalize, which affects the statistics of the cross-condition generalization performance.

Context, action and value of the previous trial can all be represented in an abstract format in the recorded areas, but context is particularly interesting because it is not explicitly represented in the sensory input, nor in the motor response, and hence it requires a process of abstraction (learning) based on the temporal statistics of sequences of stimulus-response-outcome associations. However, it is important to stress that learning may also be required for creating abstract representations of more concrete variables, such as action, which corresponds to a recent motor response, or value, which encodes a sensory experience, namely recent reward delivery.

Abstraction in Reinforcement Learning Techniques based on abstraction are an important active area of research in Reinforcement Learning (RL), and fertile ground for solution strategies to cope with the notorious "curse of dimensionality", i.e., the exponential growth of the solution space of a problem with the size of the encoding of its states [21]. Most abstraction techniques in RL can be divided in two main categories: temporal abstraction and state abstraction.

Temporal abstraction is the workhorse of Hierarchical Reinforcement Learning [22–24] and is based on the notion of temporally extended actions (or options): the idea of enriching the repertoire of actions available to the agent with "macro-actions" composed of conditional sequences of atomic actions built to achieve useful sub-goals in the environment. Temporal abstraction can be thought of as an attempt to reduce the dimensionality of the space of action sequences: instead of having to compose policies in terms of long sequences of actions, the agent can select options that automatically extend for several time steps.

State abstraction methods rely on the idea of simplifying the representation of the domain exposed to the agent by hiding or removing information about the environment that is non-critical to maximize the reward function. Typical techniques involve information hiding, clustering of states, and other forms of domain aggregation and reduction [25]. Recently, the use of neural networks as function approximators to represent value functions and policies has come to the fore as a versatile

and powerful state abstraction method to mitigate the curse of dimensionality in high-dimensional domains.

A particularly well-known example is the deep Q-network of [26], which employed a deep neural network representation of the Q-function of an agent trained using a combination of temporal-difference learning and back-propagation. The deep Q-network architecture was successfully trained to play 49 different Atari games, merely based on the set of pixels on the screen and the game score. The success of this type of techniques relies on the capability of deep neural networks trained with back-propagation to efficiently reduce the dimensionality of their inputs and implicitly identify the relevant features providing a useful description of the states of the environment.

Dimensionality of abstract neural representations Dimensionality reduction is widely employed in many machine learning applications and data analyses because, as we have seen, it leads to better generalization. In our theoretical framework, we constructed representations of abstract variables that are indeed relatively low-dimensional, as the individual neurons exhibit linear mixed selectivity [13,14]. In fact, these constructed representations have a dimensionality that is equal to the number of abstract variables that are simultaneously encoded. Consistent with this, the neural representations recorded in the time interval preceding the presentation of the stimulus are relatively low-dimensional, as expected (Supplementary S2). A previous analysis of prefrontal cortex recordings in a different experiment [13] showed that in DLPFC neural representations can exhibit maximal dimensionality. A more recent analysis of neural data from rodents also showed that the dimensionality of the neural representations is high [27]. However, dimensionality is not a static property of neural representations; in different epochs of a trial, dimensionality can vary significantly. Dimensionality has been observed to be maximal in a time interval in which all the task-relevant variables had to be mixed non-linearly to support task performance [13]. Here we analyzed a time interval in which the variables that are encoded do not need to be mixed. In this time interval, the most relevant variable is context, and encoding it in an abstract format can enhance flexibility and support inference. However, during the presentation of the stimulus, the dimensionality of the neural representations increases significantly (Supplementary S2), indicating that the context and the current stimulus are mixed non-linearly later in the trial, similar to prior observations [13,14,27,28]. Finally, we should emphasize that the data presented here might reflect intermediate regimes in which the coding directions are not perfectly parallel. Distortions of the idealized geometry can significantly increase dimensionality, providing representations that preserve some ability to generalize, but at the same time providing representations that can support operations requiring higher dimensional representations (see Supplementary Information S4).

Characterizing brain areas by analyzing the geometry of neural representations Historically, brain areas have been characterized by describing what task-relevant variables are encoded, and by relating the encoding of these variables to behavior either by correlating neural activity with behavioral measures or by perturbing neural activity to assess the necessity or sufficiency of the signals provided by the brain area. Here we provide a method that goes beyond variable encoding and instead emphasizes the importance of examining the geometry of neural representations to determine if a representation reflects a process of abstraction. As we discussed, random representations can encode all task-relevant variables, but they do not encode a variable in an abstract format, and they do not facilitate generalization. The ability to decode a variable does not

put strong constraints on the nature of the neural representation. The analysis of the geometry of neural representations promises to reveal important functional differences between brain areas, differences that may not be evident from a decoding analysis alone or from an analysis of single neuron response properties. For example, the analysis that we describe can discriminate between neural representations that require some form of learning to represent a variable in an abstract format from representations that derive from randomly connecting inputs to neurons within a brain structure.

The generation of neural representations of variables in an abstract format is central to many different sensory, cognitive and emotional functions. For example, in vision, the creation of neural representations of objects that are invariant with respect to their position, size and orientation in the visual field is a typical abstraction process that has been studied in machine learning applications (see e.g. [29,30]) and in the brain areas involved in representing visual stimuli (see e.g. [31,32]). This form of abstraction may underlie fundamental aspects of perceptual learning. Here we have focused on a form of abstraction that we believe is essential to higher cognitive functions, such as context-dependent decision-making, using conceptual reasoning to learn from experience, and making inferences. The types of abstraction that underlie these processes almost certainly rely on reinforcement learning and memory, as well as the ability to forge conceptual links across category boundaries. The analysis tools developed here can be applied to electrophysiological, fMRI and calcium imagining data and may prove valuable for understanding how different brain areas contribute to various forms of abstraction that underlie a broad range of mental functions. Future studies must focus on the specific neural mechanisms that lead to the formation of abstract representations, which is fundamentally important for any form of learning, for executive functioning, and for cognitive and emotional flexibility.

Acknowledgements This project is supported by the Simons Foundation, and by NIMH (1K08MH115365, R01MH082017). SF and MKB are also supported by the Gatsby Charitable Foundation, the Swartz Foundation, the Kavli foundation and the NSF's NeuroNex program award DBI-1707398. SB received support from NIMH (T32MH015144 and R25MH086466), and from the American Psychiatric Association and Brain & Behavior Research Foundation young investigator fellowships.

Competing Interests The authors declare that they have no competing financial interests.

Correspondence Correspondence and requests for materials should be addressed to S. Fusi or to D. Salzman (email: sf2237@columbia.edu, cds2005@columbia.edu).

1. Altmann, G. Abstraction and generalization in statistical learning: implications for the relationship between semantic types and episodic tokens. *Philos Trans R Soc Lond B Biol Sci.* **372**, 20160060 (2017).

2. Milner, B., Squire, L. & Kandell, E. Cognitive neuroscience and the study of memory.. *Neuron* **1998**, 445–468 (1998).

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

4. Wirth, S. *et al.* Single neurons in the monkey hippocampus and learning of new associations. *Science* **300**, 1578–1581 (2003).

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

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

7. Miller, E. K., Nieder, A., Freedman, D. J. & Wallis, J. D. Neural correlates of categories and concepts. *Current opinion in neurobiology* **13**, 198–203 (2003).

8. Buckley, M. J. *et al.* Dissociable components of rule-guided behavior depend on distinct medial and prefrontal regions.. *Science* **325**, 52–58 (2009).

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

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

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

12. Chung, S., Lee, D. D. & Sompolinsky, H. Classification and geometry of general perceptual manifolds. *Physical Review X* **8**, 031003 (2018).

13. Rigotti, M. *et al.* The importance of mixed selectivity in complex cognitive tasks. *Nature* **497**, 585 (2013).

14. Fusi, S., Miller, E. K. & Rigotti, M. Why neurons mix: high dimensionality for higher cognition. *Current opinion in neurobiology* **37**, 66–74 (2016).

15. Yakovlev, V., Fusi, S., Berman, E. & Zohary, E. Inter-trial neuronal activity in inferior temporal cortex: a putative vehicle to generate long-term visual associations. *Nature neuroscience* **1**, 310 (1998).

16. Bernacchia, A., Seo, H., Lee, D. & Wang, X.-J. A reservoir of time constants for memory traces in cortical neurons. *Nature neuroscience* **14**, 366 (2011).

17. Akrami, A., Kopec, C. D., Diamond, M. E. & Brody, C. D. Posterior parietal cortex represents sensory history and mediates its effects on behaviour. *Nature* **554**, 368 (2018).

18. Miller, E. K. & Cohen, J. D. An integrative theory of prefrontal cortex function. *Annual review of neuroscience* **24**, 167–202 (2001).

19. Kane, M. J. & Engle, R. W. The role of prefrontal cortex in working-memory capacity, executive attention, and general fluid intelligence: An individual-differences perspective. *Psychonomic bulletin & review* **9**, 637–671 (2002).

20. Curtis, C. E. & D'Esposito, M. Persistent activity in the prefrontal cortex during working memory. *Trends in cognitive sciences* **7**, 415–423 (2003).

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

22. Dietterich, T. G. Hierarchical reinforcement learning with the maxq value function decomposition. *Journal of Artificial Intelligence Research* **13**, 227–303 (2000).

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

24. Barto, A. G. & Mahadevan, S. Recent advances in hierarchical reinforcement learning. *Discrete Event Dynamic Systems* **13**, 341–379 (2003).

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

26. Mnih, V. *et al.* Human-level control through deep reinforcement learning. *Nature* **518**, 529 (2015).

27. Stringer, C., Pachitariu, M., Steinmetz, N., Carandini, M. & Harris, K. D. High-dimensional geometry of population responses in visual cortex. *bioRxiv* 374090 (2018).

28. Tang, E., Mattar, M. G., Giusti, C., Thompson-Schill, S. L. & Bassett, D. S. Effective learning is accompanied by increasingly efficient dimensionality of whole-brain responses. *arXiv preprint arXiv:1709.10045* (2017).

29. Riesenhuber, M. & Poggio, T. Hierarchical models of object recognition in cortex. *Nature neuroscience* **2**, 1019 (1999).

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

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

32. Rust, N. & Dicarlo, J. Selectivity and tolerance ("invariance") both increase as visual information propagates from cortical area v4 to it. *J Neurosci* **30**, 12978–12995 (2010).

33. Stefanini, F. *et al.* A distributed neural code in ensembles of dentate gyrus granule cells. *bioRxiv* 292953 (2018).

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

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

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

Methods

M1 Task and Behavior

Two rhesus monkeys (Macaca mulatta; two males respectively, 8 and 13 kg) were used in these experiments. All experimental procedures were 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. Monkeys performed a serial-reversal learning task in which they were presented one of four visual stimuli (fractal patterns). Each trial began with the animal holding down a button and fixating for 400 ms (Fig. 1a). If those conditions were satisfied, one of the four images was displayed on a screen for 500 +/- 75 ms. In each context, correct performance for two of the visual stimuli required releasing the button within 900 ms of stimulus disappearance; for the other two visual stimuli, the correct operant action was to continue to hold the button down. For half of the trials, correct performance resulted in reward delivery; for the other half of the trials, correct performance avoided having to repeat the trial but did not result in reward. If the monkey performed the correct action, a trace interval of 500 ms ensued followed by a liquid reward or by a new trial in case of a correct, non-rewarded trial. If the monkey made a mistake, a 500 ms time out was followed by the repetition of the same trial type. Monkeys had to perform the correct action in order for a new trial type to occur on the next trial. After a random number of trials between 50 and 70, the context switched without warning and with it the operant and reinforcement contingencies changed. Operant contingencies switched for all images, but for two visual stimuli the reinforcement contingencies did not change, in order to ensure orthogonality between operant and reinforcement contingencies. A contextual cue consisting of a colored frame at the periphery of the screen (Context 1, red; Context 2, blue) appeared from visual stimulus onset until the end of the trace epoch on 10 percent of trials randomly selected. Trials with a contextual frame never occurred within the first 5 trials after a block switch, and all trials with a contextual frame were excluded from all analyses presented.

M2 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 MAP system (Plexon, Inc.). Single units were isolated offline using Plexon Offline Sorter. To address the possibility that overlapping neural activity was recorded on adjacent contacts, or that two different clusters visible on 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.

M3 Selection of trials/neurons, and the decoding analysis:

The decoding algorithm was based on a population decoder trained on pseudo-simultaneous population response vectors [13].

The trials used in the decoding analysis 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 (combinations of context, value and action). From among these trials we randomly split off five trials per condition to serve as our test set, and use the remaining trials (at least ten per condition) in our training set.

Given these pre-processed data sets, we either train maximum margin (SVM) linear classifiers on the mean neural activities for the eight conditions in the training set, or we train such classifiers on the noisy training data including trial-to-trial variability. In the latter case, in order to obtain a number of trials that is large compared to the number of neurons, we re-sample the noise by randomly picking noisy firing rates from among all the training trials of a given experimental condition for each neuron independently. In this manner, we re-sample 10,000 trials per condition from the training set. While this destroys correlations between different neurons of the fluctuations around the cluster centers for each condition, we have little information about these correlations in the first place, since only relatively small numbers of neurons are recorded simultaneously. Regardless of whether we train on cluster centers only or on re-sampled data, the decoding performance is measured on the noisy data from the test set using re-sampling to increase the number of test trials.

In Fig. 2 we show the cross-validated decoding performance 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, while Fig. 6c shows similar results for linear classifiers trained on the mean firing rates in the neural data within a time window from -800 ms to 100 ms relative to stimulus onset.

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

M4 The abstraction index:

A simple way to achieve an abstract neural representation would be to cluster together the activity patterns corresponding to the conditions on either side of a certain dichotomy (defining a binary variable). For example, if the spike count patterns in a certain brain area were equal for all four stimuli in context one, and similarly coincided (at a different value) for context two, this area would exhibit an abstract representation of context, at the expense of not encoding any information about stimulus identity, operant action or reward value. We can assess the degree of clustering by comparing the average distance between the mean neural activity patterns corresponding to the conditions on the same side of a dichotomy (within or intra-group) to the average distance between those on opposite sides of the dichotomy (between or inter-group). For balanced (four versus four) dichotomies of the eight experimental conditions (context, value and action of the previous trial), there are 16 inter-group distances, and 12 intra-group distances (six on each side of the dichotomy) that contribute. We define the ratio of the average between group distance to the average within group distance as the abstraction index, which measures the degree of clustering of a set of neural representations associated with a certain dichotomy. In the absence of any particular geometric structure (such as clustering), we would expect these two average distances to be equal, resulting in an abstraction index of one, while clustering would lead to values larger than one.

Fig. 3 shows the abstraction index for the context variable computed from the measured neural activity patterns. As above for decoding, we retain only correct trials without a contextual frame that didn't occur within 5 trials of a context switch for this analysis. We z-score the overall activity distribution of each neuron before computing the mean activity pattern of each condition, and use a simple Euclidean distance metric (employing a Mahalanobis distance metric instead yields similar results).

M5 The cross-condition generalization performance (CCGP):

The hallmark feature of abstract neural representations is their ability to support generalization. When several abstract (in our case binary) variables are encoded simultaneously, generalization must be possible for all the abstract variables. We quantify a powerful form of generalization using a measure we call the cross-condition generalization performance (in fact we can view this measure as a quantitative definition of the degree of abstraction of a set of neural representations). It is analogous to the cross-validated decoding performance commonly employed, except that instead of splitting up the data randomly, such that trials from all conditions will be present in both the training and test sets, we instead perform the split according to the 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. We train (on the former) a linear classifier for a certain dichotomy that discriminates the conditions in the training set according to some label (one of the abstract variables), and then ask whether this discrimination generalizes to the test set by measuring the classification performance on the data from entirely different conditions, which were never seen during training.

This means that in order to achieve a large cross-conditions generalization performance, 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 an (abstract) dichotomy, i.e., across those conditions that belong to the same category according to the abstract variable under consideration.

Given our experimental design with eight different conditions (distinguished by context, value and action of the previous trial), we can investigate 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 the cross-condition generalization performance (CCGP) as its average over all possible sets of training conditions (of a given size). In Fig. 6a we show the CCGP (on the held out fourth condition) when training on three conditions from either side of the context, value or action dichotomies.

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 re-sample 10,000 trials per condition, with Fig. 6a showing results using the latter method.

M6 The parallelism score (PS):

The abstraction index defined above is a simple geometric measure that can quantify the degree of abstraction for a single variable. We can generalize this quantity to the case of multiple abstract variables, but we found it more fruitful to instead focus on another geometric measure based on angles rather than distances. When training a linear classifier on a pair of conditions (one from each side of a dichotomy) that differ only in one label (but agree on all others), the weight vector defining the resulting separating hyperplane will be aligned with the vector connecting the cluster centers corresponding to the neural representations of the two training conditions if we assume isotropic noise around both of them. This corresponds to the coding direction for the potentially abstract variable under consideration, given this choice of training set. Other coding directions for the same variable can be obtained by choosing a different pair of training conditions (defined by different, but equal values for the other variables corresponding to orthogonal dichotomies). The separating hyperplane associated with one such 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 Fig. 5d,e). Therefore, we introduce a measure to quantify the alignment of the

different coding directions, which we call the parallelism score (PS).

If we had only four conditions (and hence at most two abstract variables) as shown in Fig. 5d, there would be only two coding directions for a given variable (from the two pairs of training conditions), and we would simply consider the cosine of the angle between them (i.e., the normalized overlap of the two weight vectors). In our experiments, there were eight conditions (leading to at most three perfectly abstract variables), and thus pairing them across the separating hyperplane will lead to four normalized coding vectors $\vec{v_i}$ for $i = 1, 2, 3, 4$ (corresponding to four pairs of training conditions). In this case, we consider the cosines of the angles between two of them $\cos(\theta_{ij}) = \vec{v_i}.\vec{v_j}$, and we average over all six of these angles (corresponding to all possible choices of two different coding vectors). Note that these coding directions are simply the unit vectors pointing from one cluster center to another, and we don't train any classifiers for this analysis.

In general there are multiple ways of pairing up conditions across the separating hyperplane of a dichotomy under consideration. Because we don't want to assume a priori that we know the correct way of pairing up conditions (which would depend on the labels of the other abstract variables), we instead consider all possible ways of matching up the conditions on the two sides of the dichotomy one-to-one, and then define the PS as the maximum across all possible pairings of the average cosine. There are two such pairings in the case of four condition, and 24 pairings for eight condition (in general there are $(m/2)!$ for m conditions, so there would be a combinatorial explosion if m was large). Therefore, the parallelism score (for eight conditions) is defined as

$$\text{Parallelism Score} = \max_{\text{pairings of conditions}} \sum_{i=1}^{4} \sum_{j>i}^{4} \cos(\theta_{ij})/6 \ . \tag{M1}$$

The parallelism scores of the context, value and action dichotomies of our data are plotted in Fig. 6b. 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 perfect generalization across other variables. 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 and the orthogonal dichotomy would have a low parallelism score. (Moving the cluster centers in neural representation space affects the parallelism scores of at least some dichotomies, and despite being based on angles the set of all such scores depends implicitly on pairwise distances, except on the overall scale of the whole geometry). Even high parallelism scores for multiple variables don't guarantee good generalization 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. In addition, even if this is not the case and the noise is isotropic, generalization might still fail because of a lack of orthogonality of the coding directions for different variables (the eight conditions might be arranged at the corners of a parallelepiped instead of a cuboid). In summary, while the parallelism score is not equivalent to the cross-condition generalization performance, 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), and specifically for the simple case of isotropic noise, if the coding directions for different variables are approximately orthogonal to each other.

M7 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. The simplest case we consider is a shuffle of the data, in which 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). When re-sampling artificial, noisy trials, we shuffle first, and then re-sample in a manner that respects the new, random condition labels. This procedure destroys almost all structure in the data, except the marginal distributions of firing rates of individual neurons. The error bars around chance level for the decoding performance in Figs. 2 and 6, and for the parallelism score in Figs. 6 and 7, are based on this shuffle control (plus/minus two standard deviations).

A different kind of structure is retained in a class of geometric random models, which we construct in order to rule out another type of null hypothesis. For the analyses that depend only on the cluster centers of the eight conditions, we can construct a random geometry by sampling new cluster centers from an isotropic Gaussian distribution (and rescaling it to keep the total signal variance the same as in the data). 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. For analyses that depend also on the structure of the noise (in particular, decoding and CCGP 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 on the cluster centers only. 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 (for each condition independently). If our scores are significantly different from those obtained using this random model, we can reject the null hypothesis that the data was generated by a random isotropic geometry with the same total signal variance and similarly shaped noise clouds as in the data. The error bars around chance level for the CCGP in Figs. 6 and 7 are derived from this geometric random control model.

We can also consider the distribution of scores across the 35 different balanced dichotomies

we can form using the eight conditions in our data set. Since there are clearly correlations between the scores of different dichotomies (e.g. because the labels may be partially overlapping, i.e., not orthogonal), we do not think of this distribution as a random model to assess the probability of obtaining certain scores from unstructured data. However, it does allow 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.

M8 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 data set 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.

Supplementary Information

S1 The recorded neurons are not highly specialized

An ensemble of neurons could in principle encode multiple abstract variables simply by assigning each neuron to be tuned to precisely one of these variables. In this case, the ensemble can be divided into a number of subpopulations each of which exhibits pure selectivity for one of the abstract variables. Geometrically, this situation is similar to the one depicted in Fig. 5a. The situation in which neurons exhibit mixed selectivity can be obtained by rotating this geometry, as shown in Fig. 5b. This rotated representation would show the same generalization properties. In the data, we do not observe many pure selectivity neurons. If neurons were highly specialized for particular variables, training a linear classifier to decode that variable should lead to very large readout weights from the associated specialized subpopulation, but very small readout weights from other neurons (which might specialize on encoding other variables). Therefore, in a scatter plot of the (absolute values) of the readout weights for different classifiers we would expect specialized neurons to fall close to the axes (with large weights for the preferred variable, but small ones for any

others). If this situation occurred for a large number of neurons, we might expect a negative correlation between the absolute values of the decoding weights for different variables. However, in fact the weights do not cluster close to the axes, as shown in Fig. S1. The correlation coefficients of their absolute values are positive. We conclude that highly specialized neurons are not particularly common in the neural ensembles we recorded, i.e. pure selectivity appears to be no more likely than coding properties corresponding to a random linear combination of the task-relevant variables.

For each neuron we can consider the three-dimensional space of the readout weights for the three variables (components of three different unit-norm weight vectors in the space of neural activities). Clearly neurons with large readout weights for the linear classifier trained to read out a particular variable are important for decoding that variable. Some neurons have larger total readout weights than others (e.g. we can consider the sum of squares of the three weight components, corresponding to the squared radius in the three-dimensional space) and are therefore more useful for decoding overall than others which have only small readout weights.

We can also rank neurons according to the angle of their readout weight vector from the axis associated with one of the variables in the three-dimensional space. This quantifies the degree of specialization of the neuron for the chosen variable. We call the absolute value of the cosine of this angle the pure selectivity index. We can now ask whether neurons with a large pure selectivity index are particularly important for generalization, as quantified by the cross-condition generalization performance (CCGP). This can be tested by successively removing neurons with the largest pure selectivity indices from the data and performing the CCGP analysis on the remaining population of (increasingly) mixed selectivity neurons. The results of this ablation analysis are shown in Fig. S2, in which we plot the decay of the CCGP with the number of neurons removed. It demonstrates that while pure selectivity neurons are important for generalization (as expected in a pseudo-simultaneous population of neurons with re-sampled trial-to-trial variability, in which the principal axes of the noise clouds are aligned with the neural axes), they are not more important than neurons with overall large decoding weights which typically have mixed selectivity.

S2 Dimensionality of the neural representations

We utilize a technique developed in [35] to estimate a lower bound of the dimensionality of the neural response vectors in a specific time bin during a task. Specifically, similarly to what we do for other analyses, for all recorded neurons we build average firing rate patterns by averaging spike counts sorted according to task conditions indexing the trial where the activity is recorded (current trial) or the previous trial. The spike counts are z-scored and averaged in 500 ms time bins displaced by 50 ms throughout the trial. We then apply the method presented in [35] on the obtained average firing rate activity patterns independently within each 500 ms time bin. This procedure allows us to bound the number of linear components of the average firing rate patterns that are due to finite sampling of the noise, therefore providing an estimate of their dimensionality. Figure S3 shows the result of this analysis for all neurons recorded in HPC, DLPFC and ACC for which we had at least

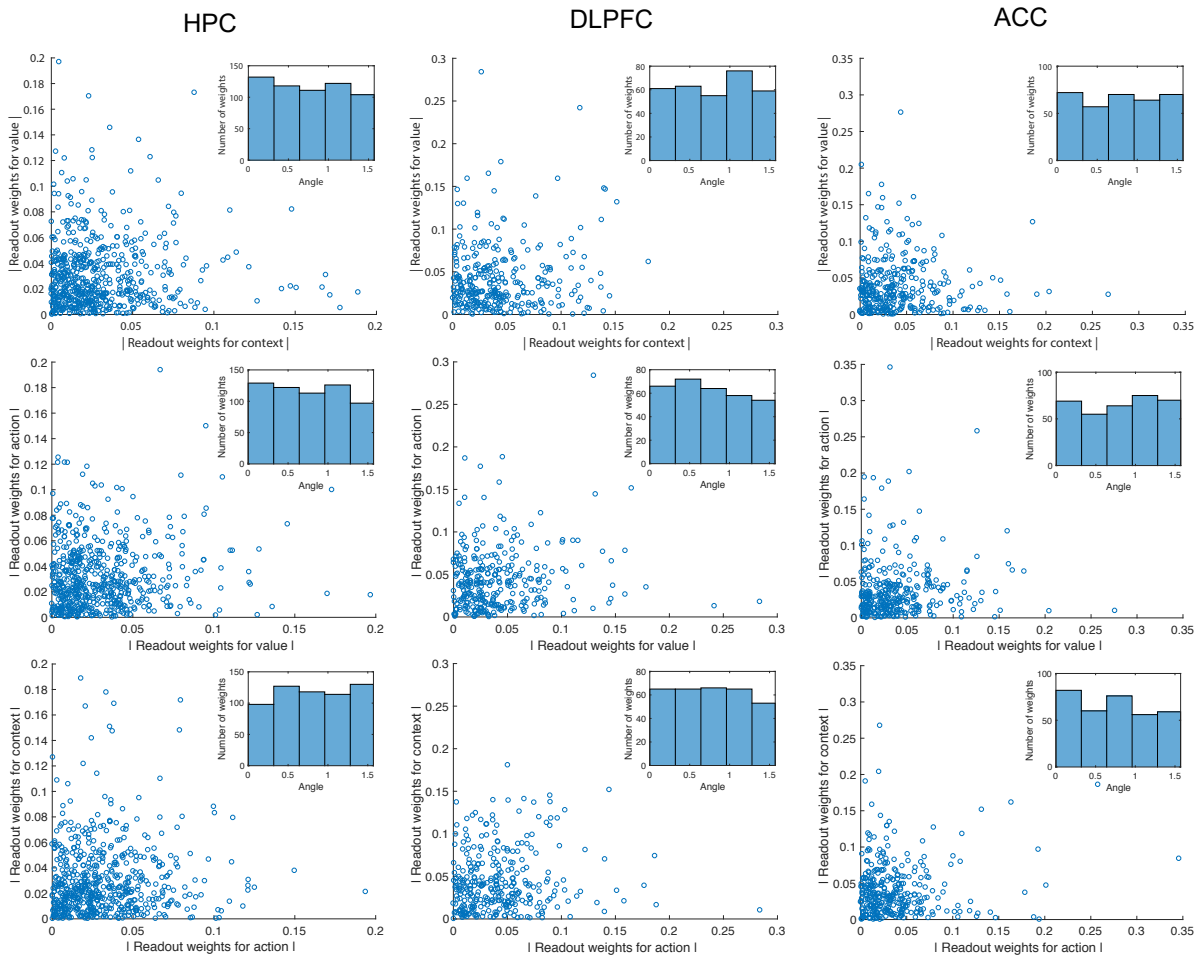


Figure S1: Two-dimensional scatter plots of the absolute values of the (normalized) decoding weights for the three task-relevant variables. The three columns (from left to right) correspond to HPC, DLPFC, and ACC. The three rows show the magnitudes of the weights for pairs of variables plotted against each other: context vs. value (top), value vs. action (middle), and action vs. context (bottom). The inset in each scatter plot 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 scatter plots) are not prevalent. Similar distributions have been observed in the rodent hippocampus [33].

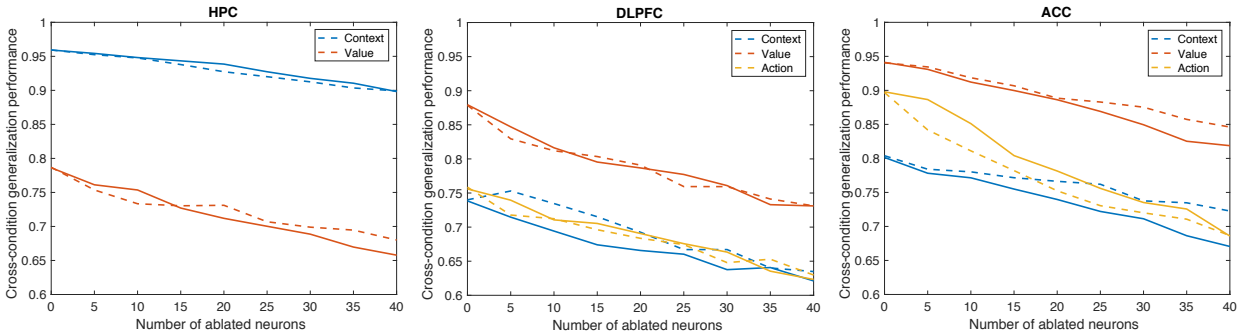


Figure S2: Cross-condition generalization performance as a function of the number of ablated neurons for the HPC (left), DLPFC (middle), and ACC (right). The solid lines show the decay of the CCGP if we successively remove the neurons with the largest pure selectivity indices for context (blue), value (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), independently 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 generalization. (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 [34].

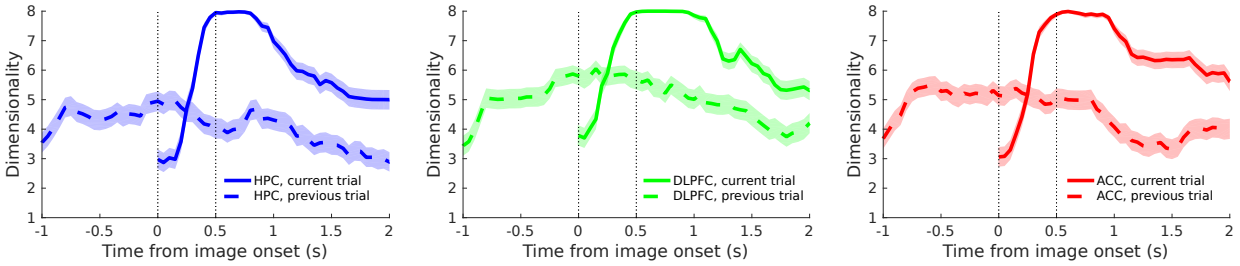


Figure S3: Dimensionality of the average firing rate activity patters as a function of time throughout the trial. The left panel illustrates the result of the analysis developed in [35] on HPC, the central panel refers to DLPFC, and the right panel to ACC. Continuous lines refer to the analysis carried 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), while for dashed line we do the same but for conditions defined in the previous trial. The lines indicate the number of principal firing rate components that are larger than all noise components, averaged over 1000 resamplings of the noise covariance matrix (see [35]). The shadings indicate the 95% confidence intervals estimated using the method for quantifying uncertainty around the mean of bounded random variables presented in [36].

15 trials per condition. As we can see, for average firing rate patterns obtained by sorting spike counts according to the 8 conditions of the current trial (continuous lines), dimensionality peaks at its maximum possible value shortly after the presentation of the image for all three areas. The dimensionality for firing rate patterns obtained by sorting the activity according to the condition of the previous trial remains around 5 throughout the trial, which is close to the value to which dimensionality in the current trial decays towards the end of the trial.

S3 High dimensionality versus generalization: Flexible computation and abstraction are not mutually exclusive

The class of neural geometries we propose would require neurons to (at least approximately) exhibit linear mixed selectivity [13,14], which entails that neural representations have low dimensionality. In fact, the dimensionality of such a geometry would be equal to the number of the (in our case binary) abstract variables involved (not counting the possible offset from the origin of the neural activity space). This dimensionality is small compared to the maximal dimensionality, which grows exponentially with the number of variables, and would be equal to the total number of conditions minus one. In the idealized case in which all abstract variables have a parallelism score of one, the representations of the different conditions coincide with the vertices of a parallelotope, and it is therefore easy to recognize that the dimensionality is small compared to the maximal dimensionality.

It has been argued [13,14] that high-dimensional neural representations are often important for flexible computation, because a downstream area may in principle have to read out an arbitrary dichotomy of the different conditions (i.e., an arbitrary binary function) to solve different tasks. This can be achieved using simple linear classifiers (used to model the type of computation that a readout neuron may be able to implement directly) only if the dimensionality of the neural representation is maximal. This desire for flexible computations afforded by high dimensionality seems to be in direct opposition to the low dimensionality implied by the abstract neural geometries that allow for cross-condition generalization.

However, in fact there is a large class of geometries that combine close to maximal dimensionality with excellent generalization properties for a number of abstract variables (which form a preferred subset of all dichotomies). Maximal dimensionality implies decodability by linear classifiers of almost all dichotomies. We will refer to this classifier-based measure of dimensionality as the 'shattering dimensionality'. This quantity is similar to the one introduced in [13], where it was measured to be maximal in neural representations in monkey pre-frontal cortex. The geometries with high dimensionality and excellent generalization don't have unit parallelism scores for the abstract variables (they are not exactly factorizable). A simple way to construct examples of such geometries is to start from a simple factorizable case, namely a cuboid, and then distort it to reduce the parallelism score. We will illustrate this in the case of eight conditions. In this case, there are at most three completely abstract variables, as in our experimental data. We can generate an artificial data set with the desired properties by arranging the eight conditions initially at the corners

of a cube (with coordinates plus/minus one), embedding this cube in a high-dimensional space by padding their coordinate vectors with zeros (here we use $N = 100$ dimensions, so we append 97 zeros), and acting on them with a random (100-dimensional) rotation to introduce linear mixed selectivity. We then distort the cube by moving the cluster center of each condition in a random direction - chosen independently and isotropically - by a fixed distance, which parameterizes the magnitude of the distortion. This operation reduces the parallelism score for the three initially perfectly abstract variables, which correspond to the three principal axes of the original cube, to values less than one. We sample an artificial data set of 1,000 data points per condition by assuming an isotropic Gaussian noise distribution around each cluster center.

On this data set we can run the same analyses as on our experimental data. In particular, we compute the parallelism score and the cross-condition generalization performance averaged over the three preferred (potentially abstract) dichotomies, with the results of these analyses shown in Fig. S4. In addition to these quantities, we compute across all 35 balanced dichotomies the average decoding performance (ADP), which is a quantity related to the shattering dimensionality. For small noise, the mean parallelism score of the abstract variables starts very close to one and from there decreases monotonically as a function of the magnitude of the distortion. The same is true for their mean CCGP, but its decline is much more gradual and almost imperceptible for small distortions. This means that for intermediate values of the displacement magnitude (of order one), we still see excellent generalization properties, and the three preferred variables are still in an abstract format.

In contrast, the ADP increases as a function of the magnitude of the distortion, due to the increased dimensionality of the representation. Balanced dichotomies include the most difficult (least linearly separable) binary functions of a given set of conditions, and if all of them can be decoded by linear classifiers the dimensionality of the neural representation will be maximal. For small values of the displacement magnitude (for which the neural geometry is three-dimensional) some dichotomies are clearly not linearly separable, and therefore the average decoding performance starts out at a value less than one (around 75%), but it steadily increases with the degree of distortion of the cube. Crucially, it reaches its plateau close to one before the CCGP drops substantially below one, i.e., there is a parameter regime in which this type of neural geometry exhibits almost maximal dimensionality enabling flexible computation, but at the same time also abstraction (in the form of excellent generalization properties) for the three preferred variables. Therefore, we can conclude that these two favorable properties are not mutually exclusive.

In the case of larger noise, the PS and CCGP start out at values substantially smaller than one already for zero distortion. The qualitative trends of all the quantities discussed remain the same, but the tradeoff between the average decoding performance and cross-condition generalization (i.e., of flexible computation and abstraction) is much more gradual under these circumstances.

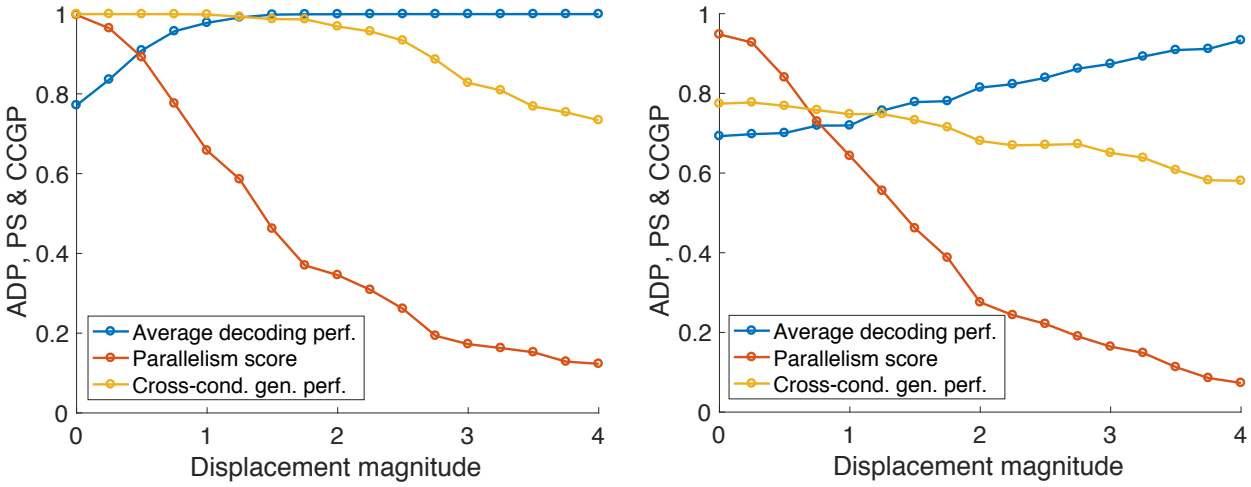


Figure S4: Analysis of an artificial data set generated by randomly embedding a (three-dimensional) cube in a 100-dimensional space, displacing its corners in independent random directions by a certain distance (displacement magnitude), and then sampling data points corresponding to the eight experimental conditions from isotropic Gaussian distributions around the cluster centers obtained from this distortion procedure. We plot the average decoding performance (ADP) across all balanced dichotomies, as well as the mean PS and CCGP 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 the average decoding performance and the cross-condition generalization performance can be simultaneously 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 the CCGP (abstraction) and the ADP (dimensionality).