Modular representations emerge in neural networks trained to perform context-dependent tasks

W. Jeffrey Johnston[1,2,*] and Stefano Fusi[1,2,3,*]

[1]Center for Theoretical Neuroscience
[2]Mortimer B. Zuckerman Mind, Brain, and Behavior Institute
[3]Kavli Institute for Brain Science
Columbia University, New York, NY, USA

[*]Corresponding authors: wjeffreyjohnston@gmail.com and sf2237@columbia.edu

October 11, 2024

Abstract

The brain has large-scale modular structure in the form of brain regions, which are thought to arise from constraints on connectivity and the physical geometry of the cortical sheet. In contrast, experimental and theoretical work has argued both for and against the existence of specialized sub-populations of neurons (modules) within single brain regions. By studying artificial neural networks, we show that this local modularity emerges to support context-dependent behavior, but only when the input is low-dimensional. No anatomical constraints are required. We also show when modular specialization emerges at the population level (different modules correspond to orthogonal subspaces). Modularity yields abstract representations, allows for rapid learning and generalization on novel tasks, and facilitates the rapid learning of related contexts. Non-modular representations facilitate the rapid learning of unrelated contexts. Our findings reconcile conflicting experimental results and make predictions for future experiments.

1 Introduction

The organizational principles of neural activity within single brain regions are still largely unknown. Recent work has shown that representations in a particular brain region tend to have both low- and high-dimensional components[1–5]. Some work argues for the existence of specialized subpopulations of neurons that represent specific variables[6, 7] (but see [8]) or that are active in specific contexts[9, 10] – however, the experimental conditions in which these specialized subpopulations emerge are not well understood. In contrast, anatomical modularity in the brain (i.e., brain regions) is well established[11–13]; yet, it is primarily understood as arising from constraints on the connectivity and physical arrangement of neural systems[14–16]. Elucidation of the computational constraints that give rise to modularity, even in the absence of anatomical constraints, is still needed.

Here, we show that context-dependent behavior naturally leads to the emergence of local modularity (i.e., specialized subpopulations of neurons within a brain region) in some conditions. These neural modules reflect the structure of the contextual tasks. Contextual behavior has been studied in

neuroscience for decades[4, 5, 17–20]. In these studies, one or more features of the experimental setup or stimuli (e.g., the color of a fixation point, the reward contingency, or the shape of a stimulus) define the context, and the animal must apply a context-dependent rule to the other features of the stimulus to determine the correct response (e.g., to select or avoid a particular stimulus). For example, when shopping for fruit in a grocery store, a shopper must keep in mind whether they want to eat their purchase now or later this week (i.e., their context) when selecting fruit with the appropriate ripeness. Yet, how the brain supports contextual behavior is not well understood. Previous work outlined a specialized role for frontal regions in selecting only information that is relevant in a particular context[18, 20, 21]. However, other work has shown context-dependent changes in representations in numerous sensory[22–24] and motor-planning[25] brain regions.

One hypothesis – which is broadly consistent with a variety of experimental and theoretical findings[2–4, 26, 27] – is that context-dependent behavior is enabled by unstructured neural representations, where neural activity is high-dimensional with many nonlinear interactions between context- and decision-related variables. Such high-dimensional representations enable a linear decoder to learn any binary classification of the stimuli, including those related to the different task contexts. However, an alternative hypothesis, which also has experimental[6, 7, 9, 10, 20, 28–31] and theoretical[32–37] support is that neural representations in brain regions essential to context-dependent behavior will be locally structured according to that behavior. Under this hypothesis, context-dependent behavior would rely on *functional modularity* in neural populations, where neural activity in a particular context is confined to either a specific subpopulation of neurons (i.e., *explicit modularity*)[6, 7, 9, 10, 31, 37] or subspace of neural population activity (i.e., *implicit modularity*)[20, 28–30, 38]. Explicit modularity implies implicit modularity.

To understand the computational constraints that give rise to these different solutions, we use the framework of representational geometry[4, 39–41]. This approach views neural representations as existing in high-dimensional population space, where each individual neuron is an axis in the space. Further, it links the arrangement of stimulus representations in this high-dimensional space to specific behavioral affordances through an assumed decoding approach – in this case, linear decoding[4, 26, 27, 41]. In particular, a representational geometry that *disentangles* different variables allows rapid learning of tasks that are linearly related to those variables[42] as well as generalization of a given decision rule across contexts[4, 5, 42]. However, in contextual behavior, the animal does not want to generalize their decision rule across contexts – instead, they must learn context-dependent decision rules. Disentangled representations do not allow a linear decoder to learn such context-dependent decision rules[26, 27]. To allow the learning of this kind of task, the representational geometry must *nonlinearly "mix"* the relevant variables and expand the embedding dimensionality of the corresponding neural representation (high-dimensional in this manuscript always means high embedding dimensionality, rather than high intrinsic dimensionality). Contextual behaviors are nonlinear with respect to the decision and context variables; thus, they require one of these high-dimensional representational geometries. In experimental work, different studies of contextual behavior have given rise to the distinct high-dimensional representational geometries described above[4, 9, 10, 20].

Here, we study the emergence of these different forms of representational geometry in artificial neural networks trained to perform contextual tasks. We show that the learned representational geometry depends strongly on both the geometry of the input representations and on the number of tasks that the network must perform – or, labels that the network must identify – in each context. In particular, we show that the strong explicit modularity observed in previous work[28, 34, 35] emerges primarily for low-dimensional input representations, while high-dimensional input representations

yield learned representations that are either unstructured or implicitly modular depending on the number of trained tasks. We explain these results with reference to several theories of learning in artificial neural networks, and we present a simple intuitive argument to understand how context dependence can lead to modularity. Then, we characterize the computational benefits of the learned representations and show that learning and generalization on new tasks within each context increases with the number of tasks that the network was already trained to perform but does not strongly depend on the input geometry, consistent with previous work[42]. However, zero-shot generalization to stimuli that the network has never seen before increases for low-dimensional input geometry but does not strongly depend on the number of tasks. Finally, we characterize how the learning of novel contexts depends on these two factors. For novel contexts that are related to previously learned contexts, we find that modular representations speed learning; while unstructured representations speed the learning of unrelated, novel contexts. Together, our results provide a framework for understanding diverse experimental findings. Further, they provide ways to go beyond predictions for observed representational geometries and into predictions for the ability of animals to generalize and learn novel contexts. We close by discussing these predictions in more detail.

2 Results

We want to understand how representations of stimuli described by multiple features are shaped by specific task demands. While we study this in artificial neural networks, we will make predictions for real neural data – and provide a theoretical understanding of the large-scale, behavioral and computational constraints that shape neural population representations in the real brain.

To do this, we study stimuli described by a set of features (e.g., fruit can be described by color, size, etc.; fig. 1a), which, for simplicity, we assume to be binary. Previous work has investigated how representations of continuous stimuli in artificial neural networks are shaped by the learning of multiple distinct classification tasks (fig. 1b, top)[42]. This work found that the representation reflected the structure of the underlying latent variables (an "abstract" representation; fig. 1b, bottom), which enables generalization across different stimulus features and rapid learning[4, 42]. Here, we generalize this work to ask how representations in artificial neural networks change when the network is trained to perform multiple contextual tasks (fig. 1c, top). In this case, one of the inputs to the network acts as a context variable. This could represent a change in the internal state of the animal (e.g., searching for a snack versus planning for the future, fig. 1c). It could also be an explicit context cue, like a change in the color or shape of the fixation point[20]. Importantly, the same stimuli (and the same decision-related feature values) will appear in different contexts. In each of these contexts, the network must learn to perform a set of linear classification tasks on other stimulus features (e.g., color). In many situations (e.g., fig. 1c), the abstract representations from before do not allow learning of these tasks with a linear decoder (fig. 1c, bottom left), even when the task is linearly separable in each context. Instead, the network must learn a non-abstract, higher-dimensional representation (fig. 1c, bottom right).

However, the task structure alone does not dictate the representational geometry of a network performing contextual tasks. There are three different representational schemes that allow a linear decoder to perform these tasks. First, the representation could split into distinct subpopulations, where units in each subpopulation are only active in a single context (fig. 1d, left). This "explicitly modular" pattern of responses has been observed experimentally[9, 10]. Second, the representation could be active along specific dimensions of population space in each context, but without the single

unit-level structure of explicit modularity (fig. 1d, middle). This "implicitly modular" pattern of responses has also been observed experimentally[20], and can be viewed as a rotated version of the explicitly modular representations from before. Third, the representation could be totally unstructured and high-dimensional, without context-specific subspaces or subpopulations (fig. 1d, right). To our knowledge, this "unstructured" pattern of selectivity has not been directly tested for in context-dependent tasks, though neural representations are often reported to have the maximum dimensionality afforded by the experiment in which they were recorded[26, 27].

We will study when and how these three distinct representational geometries emerge in neural networks trained to perform context-dependent behavior. We will vary the geometry of the input representations provided to the neural network as well as the number of contextual tasks that the network must learn to perform (fig. 1f).

These different representational schemes are distinguished, in part, by their dimensionality (fig. 1e). While fully disentangled representations have dimensionality that scales linearly with the number of latent variables, they cannot be used by a linear decoder to perform the task (fig. 1e, "disentangled"). In contrast, a maximally unstructured representation has dimensionality that scales exponentially with the number of latent variables (fig. 1e, "max unstructured"), and can be used by a linear decoder to perform any task. So, such a representation is maximally flexible[26], but also impractical for large numbers of latent variables as it can quickly require more dimensions than there are neurons in the brain. The explicitly and implicitly modular solutions discussed above have the same dimensionality, and we show that this dimensionality scales linearly with the number of latent variables (fig. 1e, "modular"). Thus, modular representations provide a solution for contextual behavior while avoiding the primary drawback of the maximally unstructured representation. However, the modular solution is also specialized and does not provide the same flexibility as the maximally unstructured representation. Modular representations provide a solution to contextual behavior through second-order interactions, where the context variable interacts with all of the decision variables. An analogous unstructured solution would be a representation with all possible second-order interactions between variables. Such a representation would not make it possible for a linear classifier to learn any binary classifications of the stimuli, but does make it possible for a linear classifier to learn all classifications that have similar structure to contextual tasks. This minimally unstructured representation also has dimensionality that scales linearly with the number of latent variables, but with a greater slope than the modular representation (fig. 1e, "min unstructured" and see *Dimensionality of unstructured solutions* in *Methods* for details).

2.1 The representational geometry of the input

Our input always consists of several decision-related variables (typically, $D = 3$, several context variables (typically, $C = 2$), and several irrelevant variables (typically, $I = 4$; fig. 2a, left, and see *Input model* in *Methods*). We hypothesize that the geometry of the input representations provided to the contextual multi-tasking model is one of the primary factors that shapes the learned representations. To explore this idea, we develop an input model that interpolates between a low-dimensional, disentangled input representation (fig. 2a, top) and a high-dimensional, conjunctively mixed, unstructured input representation (fig. 2a, bottom). For the low-dimensional, fully disentangled input, each input variable is provided along a separate dimension of the input space. In the brain, the linear tuning for facial features in the inferotemporal cortex is consistent with a disentangled representation[43, 44]. In the unstructured input, all of the input variables are nonlinearly mixed together, such that each dimension in the input corresponds to a particular combination of

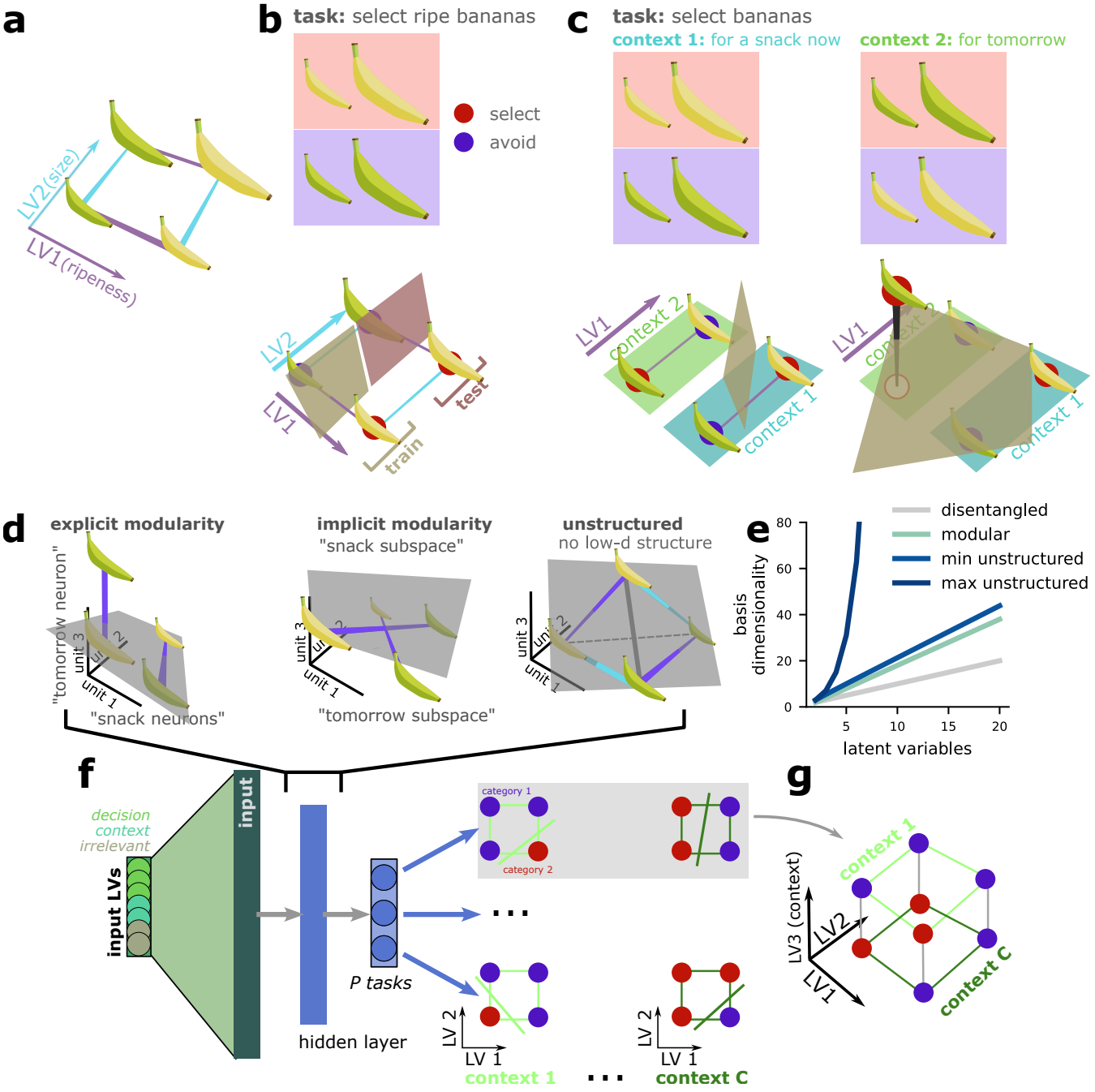

Figure 1: Modular representations may emerge as a consequence of contextual behavior. **a** Stimuli are described by a set of latent variables. **b** A linear task (that depends on either a single latent variable or a linear combination of latent variables; top) can be learned by a linear decoder on a representation that directly reflects the latent variable structure (bottom). This representation is also abstract, and allows for generalization of the decision boundary when the decoder is trained on only a subset of stimuli (train and test). **c** A contextual task, where the value of a contextual feature determines the linear task performed in each context (top) cannot be solved by a linear decoder given a representation reflecting the latent variables (bottom left). Instead, the variables related to the decision must be nonlinearly mixed with the context variable (bottom right). **d** Explicitly modular (left), implicitly modular (center), and unstructured (right) representations allow a linear decoder to learn linear contextual tasks. **e** The modular representations are the lowest dimensional solution to contextual tasks. **f** A feedforward neural network learns to perform different numbers of contextual tasks given different input geometries.

values across all the features (for instance, where instead of having units that respond to color and shape independently, as in the disentangled representation, there are units that respond only to green squares while other units respond only to red circles, and so on). In the brain, many areas of prefrontal cortex have been shown to have highly nonlinear representations[26, 27]. In between these two extremes are representations with both disentangled and nonlinear representations superimposed on each other. Previous theoretical and empirical work has shown that such intermediate representations can preserve the computational benefits of both disentangled and high-dimensional representations[2, 4, 45], as well as shown that many brain regions in sensory and frontal cortex exist in this middle ground [1–4, 45].

As the strength of the nonlinear mixing between the different variables is increased, the dimensionality of the resulting representations also increases (fig. 2b). We develop two methods for visualizing these representations that will reveal modular structure if it exists. We introduce these visualizations now, since they are used to analyze the representations learned by the contextual multi-tasking model later. First, we sample stimuli from each of the two contexts (fig. 2c, d, y-axis) and sort the units in the input representation by the difference in average activity between the two contexts (fig. 2c, d, x-axis). A modular representation would have a clear block structure under this analysis (fig. 2c). In the input, this sorting procedure does reveal a weak block structure in the activity of each network, which could represent pre-existing modular structure within our input model (fig. 2d). Next, we compute the average activity in each context for every unit in our input model. Then, we cluster the units in this mean activity space, and color them according to their cluster membership (fig. 2e, f). We choose the number of clusters according to the Bayes information criterion (see *Clustering analysis* in *Methods* for details). In an explicitly modular network, we expect to see a cluster of units that is active in each context, but inactive in the other context (fig. 2e). In the input model, we find the points are best explained by a single cluster for each input geometry (fig. 2f), indicating that there is no modular structure in any of our input models, despite the apparent structure present in the earlier visualization (fig. 2c). The disentangled input models show a negative correlation in average activity across the two contexts, this is because the contexts themselves are negatively correlated: for each stimulus, one context variable will be one and the other will be zero.

2.2 The contextual multi-tasking model

The contextual multi-tasking model (fig. 3a) receives the input representations produced by the input model (fig. 3a) and is trained to perform P randomly chosen binary classification tasks fig. 3c). Within each context, all tasks are linearly separable with respect to the original latent variables (and they are also linearly separable with respect to our input representations). However, in most cases, the full task (combined across both contexts) will not be linearly separable (as discussed above, fig. 1c, bottom left). The readout layer of the network is effectively a set of linear classifiers. So, we know that, if the network successfully solves all P tasks, then those P tasks must be made linearly separable in the hidden layer. The explicitly modular, implicitly modular, and unstructured solutions introduced above all satisfy this constraint.

We begin by visualizing the activity in four trained networks that vary in both the input geometry and in the number of tasks that the contextual multi-tasking model is trained to perform. First, we train a network to perform a single contextual task $P = 1$ across $C = 2$ contexts, where the same $D = 3$ latent variables are relevant in both contexts. We train this network with fully disentangled input representations (fig. 3d, top left). This network learns explicitly modular representations, in

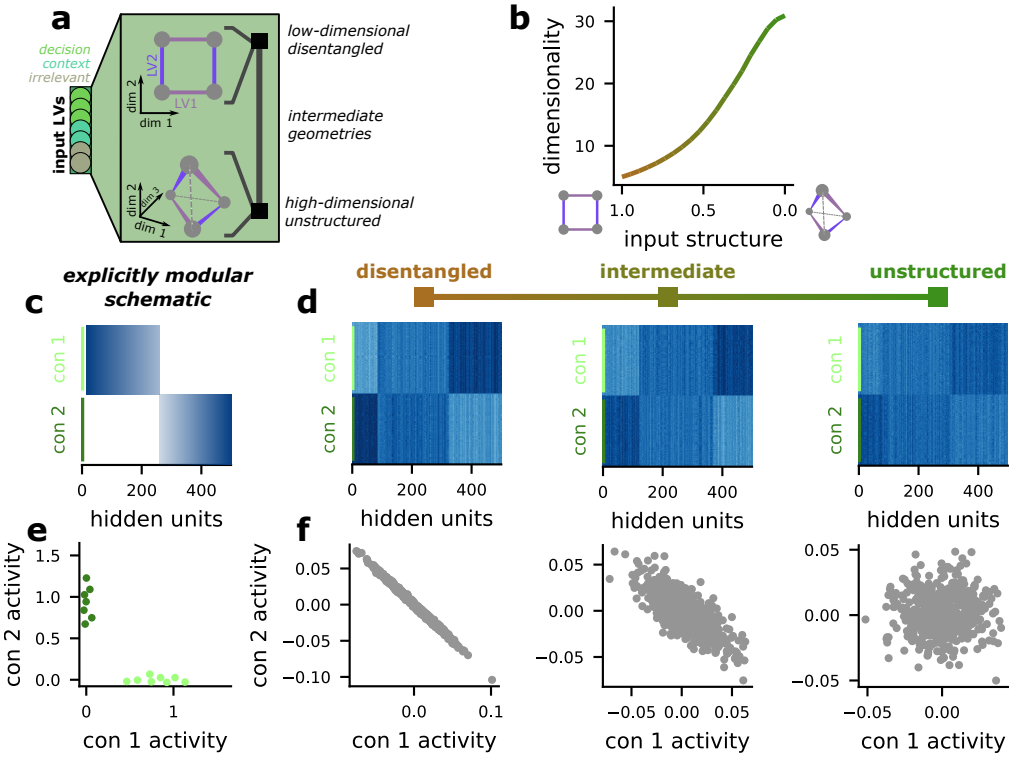

Figure 2: The input model is used to generate the representational geometry of the input. **a** There are decision-relevant, contextual, and irrelevant latent variables (left). We explore the full spectrum between fully disentangled and unstructured representational geometries (right). **b** The disentangled geometry is low-dimensional (left), while the unstructured geometry is high-dimensional (right). **c** A schematic of the expected activity pattern across contexts in an explicitly modular representation. **d** A visualization of the activity pattern in the input model unit activations for disentangled (left), moderately structured (middle), and unstructured models (right). The order of the units is sorted according to their average activity in each of the $C = 2$ contexts. 1000 samples are taken from each context. The visualization is the same as the schematic in **c**. **e** A schematic of the expected average activity across units in the two contexts for an explicitly modular representation. **f** The average activity of each unit in the two contexts for the same three models as above, following the visualization in **e**. The points are colored according to their cluster assignment, this assignment is used to sort the above plots.

which some units in the hidden layer are active in one of the two contexts but not the other. The units in the hidden layer cluster into two broad groups: one subset of units are active in the first context but relatively silent in the second (fig. 3d, dark green); a distinct subset of units is active in the second context but relatively silent in the first (fig. 3d, light green and gray). We quantify modular structure by computing the fraction of units in the hidden layer that are active in one context but nearly silent in the other (fig. 3e, and see *Contextual fraction* in *Methods* for details). This pattern of results holds when we increase the number of contextual tasks that the network is trained to perform ($P = 10$, fig. 3d, top right). The activity in these networks resembles neural activity recorded from posterior parietal cortex in a mouse performing a contextual discrimination task[9].

Next, we train a network to perform one ($P = 1$, fig. 3d, bottom left) and ten ($P = 10$, fig. 3d, bottom right) contextual tasks with fully unstructured input representations. This eliminates the explicitly modular structure from before: Neither network develops significant clustering in average activity space. Next, we apply our metric for explicit modularity to networks trained on a full range of input geometries and numbers of tasks (fig. 3f). This analysis reveals that explicit modularity emerges as a function of the input geometry and does not depend on the number of tasks the

network is trained to perform. A critical value of nonlinear mixing in the input geometry marks a transition between modular and non-modular solutions to the contextual tasks. Below, we show how this critical value depends on the features of the input, and link the emergence of modularity to a competition between learning from the disentangled and unstructured components of the full input representation (fig. 5).

However, there could be additional structure in the representations – that is, this fraction of contextual units only quantifies explicit modularity, but does not quantify implicit modular structure. To quantify this, we develop a metric that we refer to as subspace specialization (fig. 3g). This metric characterizes to what degree the activity from distinct contexts is confined to specialized subspaces as well as to what degree this structure is unique to the context variables rather than being generalized across all variables. For instance, in a fully unstructured representation, the activity in each context would be in distinct subspaces (fig. 3g, left, the dot product between the green lines is approximately zero). However, any other latent variable would also split the activity into distinct subspaces (for instance, by computing the subspace in which activity occurs when that variable takes on one value and comparing it to the subspace when that variable takes on the other value; fig. 3g, left, the dot product between the cyan lines is approximately zero). The subspace specialization metric is 0 when the subspace overlap across contexts is the same as the average subspace overlap for all other variables (fig. 3g, left bottom). Similarly, the subspace specialization metric will be zero for disentangled representations (fig. 3g, middle), where the context and latent variable-conditioned vectors will all be parallel. However, subspace specialization will be positive for representations that split into orthogonal subspaces when conditioned on a contextual variable but not on a different latent variable (fig. 3g, right, and see *Subspace specialization* in *Methods* for a full description of the metric).

Using subspace specialization, we quantify the emergence of implicitly modular representations as a function of input geometry and the number of tasks the network is trained to perform (fig. 3h, right). As expected, whenever a representation is explicitly modular, it also has positive subspace specialization. However, implicit modularity also emerges for high-dimensional input geometries once the network is trained to perform a sufficient number of tasks. Thus, the only regime with truly unstructured representations is for contextual multi-tasking models trained to perform few tasks on high-dimensional inputs. To illustrate these transitions, we focus on a reduced parameter range where networks have relatively unstructured inputs and are trained on relatively few tasks (fig. 3f,h, black box). Then, we binarize both measures used above, and show the transitions between the three representational regimes (fig. 3i).

This framework can help to explain the heterogeneous representations observed for contextual behavior in the brain. It reveals that the representational geometry learned by the network depends strongly on the geometry of the input representations – or, alternatively, on the representational geometry that exists prior to the start of training on the new task – as well as on the number of distinct tasks that the network performs in each context.

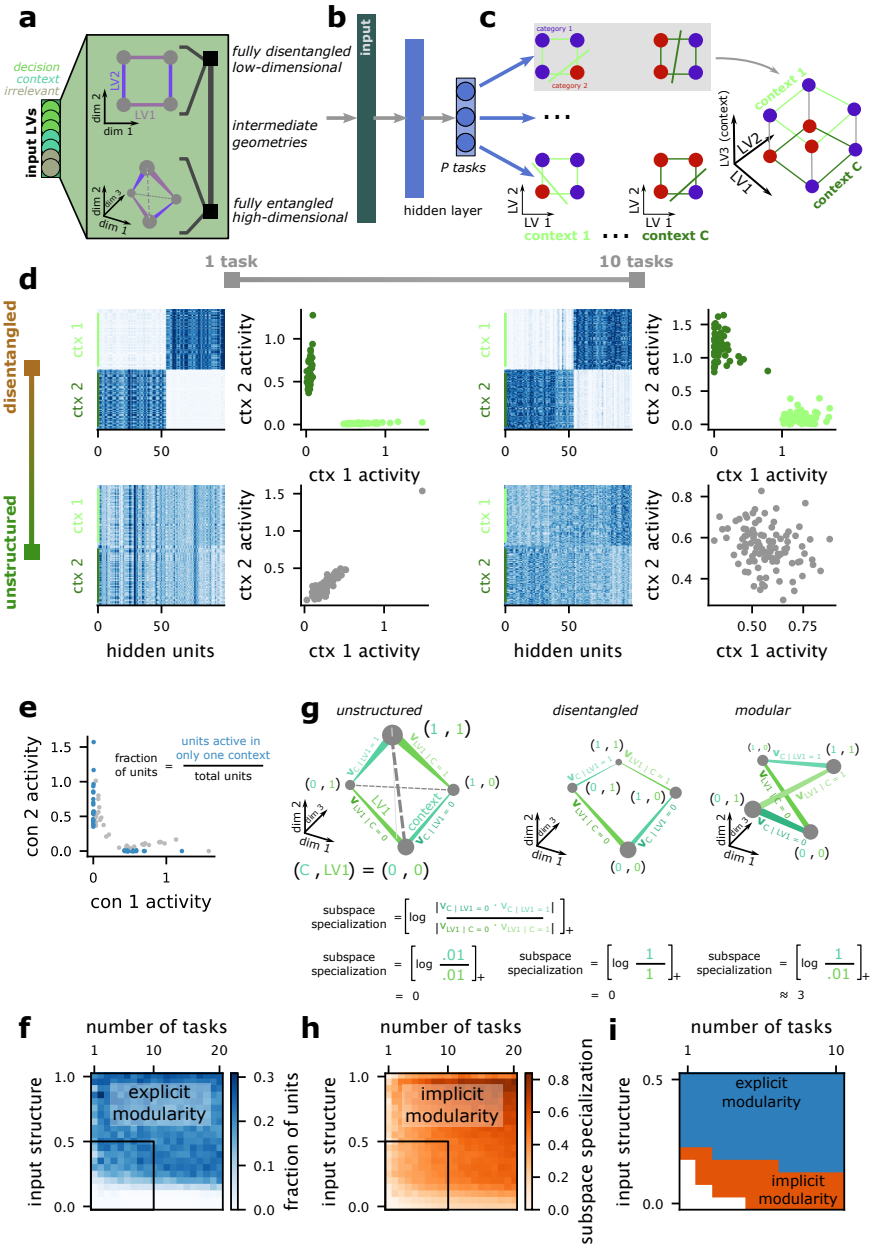

Figure 3: Modular representations emerge when the input is disentangled or the network is multi-tasking within each context. **a** The input model, already described in Fig. 2. **b** The contextual multi-tasking model is a single hidden-layer feedforward network trained to perform P tasks across $C = 2$ contexts. **c** Each task is a linear classification task within each context (left). The linear boundary is randomly selected for each context, giving rise to a nonlinear classification task across multiple contexts (right). **d** Hidden layer representational structure across several example models. The hidden unit activations in both contexts, sorted by cluster membership (left). The average activity of each unit in the two contexts, colored according to cluster membership (right). The models in the left column are trained to perform $P = 1$ task in each context, while the models in the right column perform $P = 10$ tasks on the same 3 latent variables in each context. The models in the top row are given disentangled inputs; the models in the bottom row are given high-dimensional, unstructured inputs. **e** The fraction of units active in only a single context given different choices for the number of tasks P and the input geometry. **f** The fraction of units specialized for a single context (defined as in **e**) for the full parameter range. Specialized units are indicative of explicit modularity (top). **g** The subspace specialization metric applied to three different cases (see *Subspace specialization* in *Methods* for more details). The measure is zero for both unstructured (left) and disentangled (center) representations of the variables. It is positive for a modular representation (right). **h** The same as **f** except showing the subspace specialization metric, a measure of implicit modularity. **i** The reduced parameter range indicated by the black box in **f** and **h**. Blue shows where the fraction of specialized units is $> .05$, while orange shows where subspace specialization is $> .05$ out of the remaining parameter choices. White has unstructured representations.

2.3 Intuition about the computational advantage of modularity

Consider the examples in fig. 4a,b. We assume that there are two contexts with two decision-relevant variables (so, 8 total stimuli), and we start by considering one task that separates the 8 stimuli into two groups of 4 stimuli each (e.g., fig. 4a). To further simplify the argument, each task is constrained to be aligned with one of the relevant latent variables. While we assume that tasks are linearly separable within each context, the full task across both contexts will not be linearly separable in most cases – in fact, the full task will only be linearly separable if the task selected in both contexts is exactly the same (see *Alternate decompositions* in *Methods* for details). This is the problem that the task output unit has to solve (separate the 8 points into two groups according to their color in fig. 4a,b).

The modular representation provides a simple solution to this problem: some of the units in the intermediate layer could be active only in context 1 (unit 1 and 2 in fig. 4a), and the others in context 2 (unit 3 and 4). This will always yield a representation that makes the full task linearly separable, since we assumed that the task is linearly separable in each context. Then, in context 1, the output unit tunes the weights between it and the active neurons. The weights from the other units are irrelevant because these other units are inactive. The same procedure can be repeated in context 2. In other words, by dividing the 8 points into two groups of four points, the network is able to 'handle' each group with separate sets of units and yield a representation that makes the full task linearly separable for the output unit.

This modular decomposition based on context is clearly a solution (fig. 4a, b, "context decomposition"), but it is typically not the only solution. Indeed, there are other ways of decomposing the problem and dividing the 8 points into two groups (fig. 4a,b, "non-context decomposition" and fig. 4c). However, we show that, when the number of tasks increases, the number of non-context decompositions that make all tasks linearly separable shrinks rapidly (fig. 4d, left, and see *Alternate decompositions* in *Methods*). In contrast, the context decomposition is guaranteed to make all tasks linearly separable for any number of tasks. The modular solution then has a computational advantage over these alternative decompositions. For simplicity we discussed a solution that is explicitly modular, but the same arguments would apply to implicit modularity. Indeed, an implicitly modular solution is just an explicit one that is rotated, and the same logic of non-interference between learning in different contexts applies in this rotated space as well.

If this understanding is correct, then we expect that networks performing only a single task may have lower subspace specialization – and less modular structure – than networks performing multiple tasks. To test this hypothesis, we calculate the average number of non-context decompositions that yield a linearly separable representation as a function of both the number of tasks and the number of relevant latent variables. To approach this, we view each task as eliminating certain decision-related variables but not others as candidates for decomposition, depending on how the two linear sub-tasks relate to each other (fig. 4c, and see *Alternate decompositions* in *Methods* for details). We find that more decision-related variables yield fewer possible non-context decompositions (fig. 4d, left). Thus, we also expect more decision-related variables to be associated with greater subspace specialization for the same number of tasks, and our simulations confirm both of these predictions (fig. 4d, right).

However, we do not expect unstructured input geometries to be constrained in the same way. In this case, the full contextual task is already linearly separable in the input – and therefore decomposition into sub-tasks is not necessary. However, the contextual task is still lower-dimensional than

the input geometry, so we expect our network to find a solution with dimensionality in between the dimensionality of the input and of the required output (fig. 4e)[46]. We expect this low-dimensional solution to inherit its structure from the structure of the required output. Thus, we expect the subspace specialization in the representation to follow the subspace specialization in the required output. To test this, we calculate the subspace specialization for the output for different numbers of decision-related variables (fig. 4f). Then, we show that this pattern of emergence matches the pattern that we see in the representation layer of our trained networks (fig. 4g). Interestingly, this pattern is the opposite of what we found for the disentangled inputs. For disentangled inputs, subspace specialization emerges more quickly for more decision-related variables; in contrast, for unstructured inputs, subspace specialization emerges more quickly for fewer decision-related variables. This further illustrates how changes to the input geometry alter learning dynamics in the network. However, the overall cause of the emergence of subspace specialization remains the same: The presence of low-dimensional structure within the tasks that the network exploits in different ways for different input geometries.

2.4 Explicit modularity emerges when the disentangled component of the input provides faster learning.

The input geometries used to train our models (fig. 5a) can be decomposed into two components: a structured, disentangled component (fig. 5b, brown) and an unstructured component (fig. 5b, green). The input structure is varied by changing the relative strength of these two components while keeping the overall strength of the full representation constant (see *Input model* in *Methods* for details). So, an input geometry with high structure has a large disentangled component and a relatively small unstructured component (fig. 5b, top); in contrast, an input geometry with low structure has a small disentangled component and a large unstructured component (fig. 5b, bottom).

We hypothesize that, in most cases, learning is dominated by one of these of two components. This follows from previous work that shows winner-take-all dynamics for learning in artificial neural networks[35], where possible solutions with a small initial advantage in learning speed come to dominate the learned solution. This prior work shows that for disentangled inputs and a network trained to perform a single contextual task, the explicitly modular solution provides the fastest learning – and dominates the resulting hidden layer representation[35]. Here, we hypothesize that our networks have explicit modularity precisely when learning from this disentangled component of the input is faster than learning from the unstructured component of the input. To test this, we train models on the decomposed inputs, where one model is trained only on the disentangled component of an input geometry with a given level of structure (fig. 5c, yellow line) and a second model is trained only on the unstructured component of the geometry (fig. 5c, green line). We then take the difference between the average loss of these models across training (fig. 5c, "Δ learning" and right) and repeat this procedure for many different input structures: for high input structure, the disentangled component will be large in magnitude and the unstructured component small; and vice versa for low input structure. As the input structure changes, the green and brown curves change because the magnitude of the representation they are learning from changes in size. In particular, for high input structure (fig. 5c, top), the brown curve reflects learning from a large disentangled component and the green curve reflects learning from a small unstructured component.

Our hypothesis predicts that, once the network begins to learn more quickly (lower average loss) from the unstructured component relative to the disentangled component (fig. 5d, top, grey bar), the modular structure in the learned representation will start to disappear since the unstructured

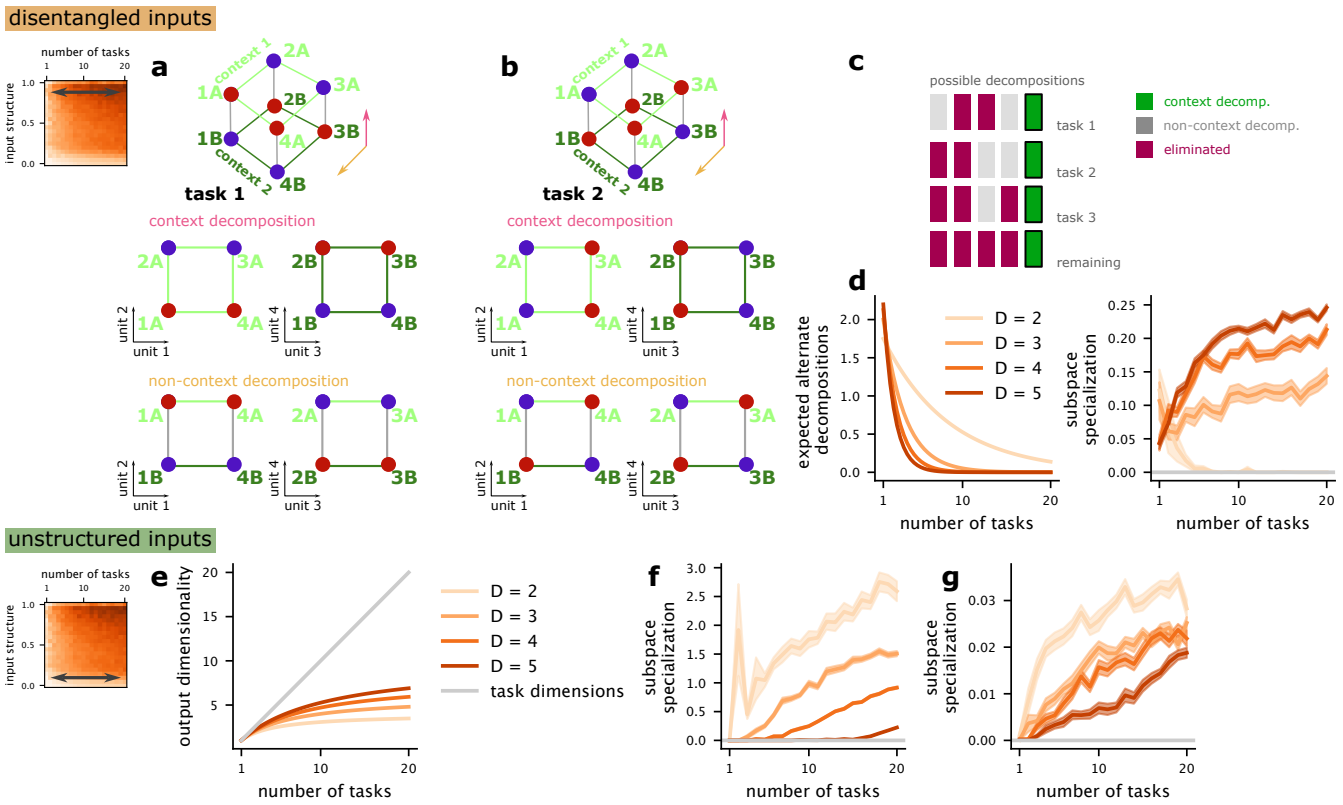

Figure 4: Understanding the emergence of implicit modularity. **a** (top) An example contextual task in a $D = 2$-dimensional latent variable space with a single context variable. (middle) The context decomposition for this task, which makes both component tasks linearly separable. (bottom) An example non-context decomposition which makes the component tasks linearly separable. **b** The same as **a** except the non-context decomposition does not yield linearly separable tasks. Thus a network learning to perform both this task and the task in **a** would be forced to use the context-decomposition. **c** Schematic showing how each task eliminates different non-context decompositions, while the context decomposition is guaranteed to not be eliminated. **d** (left) The expected number of surviving non-context decompositions as a function of the number of tasks. (right) The emergence of implicit modularity (i.e., subspace specialization) as a function of the number of tasks for different numbers of decision-related variables. The $D = 2$ case goes to zero because implicit modularity is not distinct from the unstructured representation (i.e., both are tetrahedrons). **e** The dimensionality of the required output as a function of the number of tasks for different numbers of latent variables. **f** The subspace specialization of the output itself as a function of the number of tasks. **g** The subspace specialization in trained contextual multi-tasking models as a function of number of tasks for different numbers of latent variables. The pattern of emergence is the same as in **f**.

component leads to unstructured representations. Our analysis shows that the points at which Δ learning crosses zero (vertical lines) correspond to the start of a decrease in explicit modularity and a transition to an unstructured or implicitly modular solution (fig. 5d, bottom, grey bar).

We further hypothesize that the transition point at which the unstructured component begins to provide faster learning will depend on the number of irrelevant latent variables. In particular, as the number of irrelevant latent variables increases, we expect that learning from the unstructured component will slow – since the learning problem itself becomes higher-dimensional and the error gradient used in learning is distributed across more dimensions. In contrast, the disentangled component of the representation will be unaffected or weakly affected, since the relevant part of the representation has the same dimensionality regardless of the number of irrelevant variables. Simulations with different numbers of irrelevant latent variables follow the expected pattern (fig. 5e, different green lines). The amount of modularity begins decreasing for higher input structure in

situations with fewer irrelevant latent variables, indicating that this increased learning speed in the unstructured component is affecting the learned representation.

Finally, we visualize the structure of learned representations for three key points along the spectrum of input geometries (fig. 5d, circles along the curve) and two different numbers of irrelevant variables (fig. 5e, light and dark green lines). First, we show that modular structure emerges with high input structure, when the disentangled component dominates learning (fig. 5f, left). Second, we show that this modular structure becomes less apparent as the input structure decreases and the unstructured component begins to dominate learning (fig. 5f, center). Third, in a highly unstructured case, the modular structure is completely eliminated (fig. 5f, right). These results provide an explanation for when explicit modularity emerges as input structure becomes stronger, and indicate that other elements of the input – i.e., the number of irrelevant variables – can affect this transition.

2.5 The model develops an abstract and high-dimensional representational geometry

Next, we study the geometry of representations developed by the contextual multi-tasking model across the entire hidden layer for stimuli that are sampled from a single context. We begin by visualizing the activity for the same range of parameters discussed above (fig. 6). We show that increasing the number of tasks that the network is trained to perform also increases the disentangled structure of the representations, consistent with previous work[42]. In particular, when the network is trained to perform a single task, the dimension relevant to that task dominates the within-context representation (fig. 6a, left). These task-specific representations emerge when animals are overtrained on specific categorization tasks as well[47–49]. In contrast, when the network is trained to perform multiple tasks within each context, the representation preserves information about all of the relevant latent variables within each context (fig. 6a, right), consistent with previous work in artificial neural networks[42]. These variables are encoded in low dimensional disentangled representation, which reconstructs the part of the latent space containing the relevant variables.

To quantify this change in structure, we adapt the cross-condition generalization performance used in previous work[4, 42] to quantify disentangled or abstract structure in neural representations. Within each context, we train a linear decoder to perform a random, linear classification that depends on the relevant latent variables. However, we train this decoder only using a subset of all stimulus conditions (fig. 6b, "train"), and then test whether or not it generalizes to the held-out set of stimulus conditions (fig. 6b, "test"; see *Cross-condition generalization performance* in *Methods* for more details). We then quantify the average generalization performance in the same parameter range as before (fig. 6c) as well as compute the average D' of the margin of the classifier (fig. 6d). This reveals that generalization performance primarily depends on the number of tasks the network performs in each context and not on the geometry of the input – which is consistent with previous work in artificial neural networks for non-contextual tasks[42]. The average margin D', however, strongly reflects the input geometry (fig. 6d), with higher D' for more structured input representations – which indicates a greater robustness to noise. The high generalization performance even for relatively unstructured learned representations (fig. 6c, d, bottom left of parameter space) indicates that task training still influences representations enough to impose abstract structure prior to the emergence of context specialization.

Further, we quantify the strength of nonlinear perturbations to these low-dimensional abstract representations that emerge as more tasks are learned. To quantify this structure, we use a previously

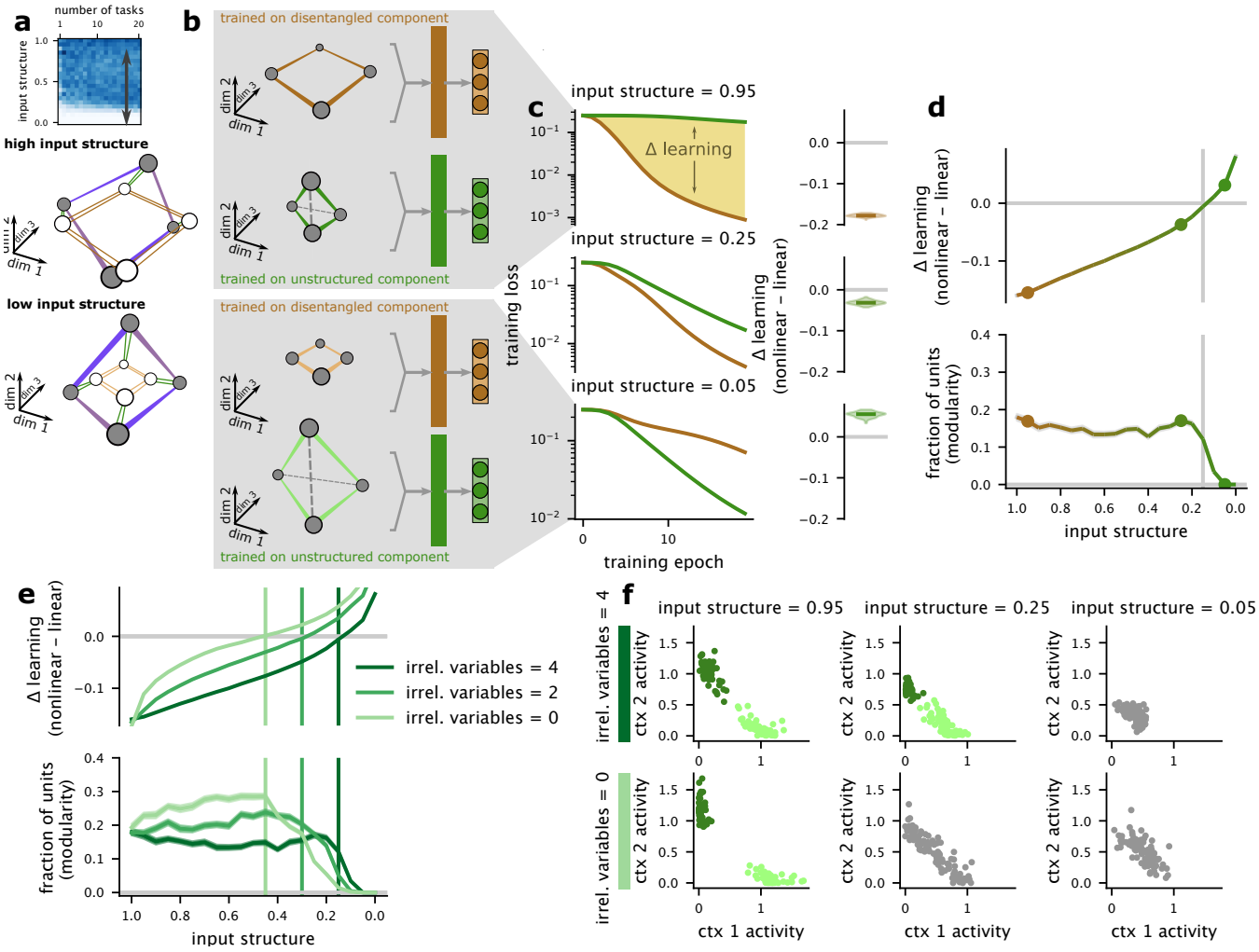

Figure 5: Understanding the emergence of explicit modularity. **a** (top) We focus on the how the representation changes with changes in input structure and contrast between input geometries with high (middle) and low (bottom) structure. **b** All input geometries can be decomposed into distinct disentangled and unstructured components. We decompose input geometries into these two components and train contextual multi-tasking models on only the isolated component (right). **c** We track the loss of models trained only on the disentangled and unstructured components across training (left), and summarize the average difference (right) as a function of the input structure (top to bottom). **d** As the input structure decreases, the unstructured component begins to yield faster learning than the disentangled component (top). The modular structure learned by the full network starts to disappear (bottom) as the unstructured component begins to provide faster learning (vertical line). **e** The same as **d** for networks with different numbers of irrelevant variables. Fewer irrelevant variables leads to an earlier transition away from modularity. **f** Example average activity in each context for networks with input structure indicated by the outlined points in **d** and either 4 irrelevant variables (top) or 0 irrelevant variables (bottom). While distinct clusters exist for high input structure (left), the transition to faster learning from the unstructured component of the representation is associated with a loss of distinct clusters (center) and eventually with a fully unstructured representation (right).

developed measure of representational geometry referred to as the shattering dimensionality (fig. 6e, and see *Shattering dimensionality* in *Methods* for details)[4, 26, 27]. A high shattering dimensionality (that is, close to 1) indicates that the representation has the maximum dimensionality for the number of conditions. A high shattering dimensionality requires nonlinear interactions between the representations of different decision-related variables[26]. We find a high shattering dimensionality across the entire range of parameters explored here (fig. 6f) as well as relatively large average margin D' (fig. 6g). This is surprising because the network does not require any nonlinear interactions

between variables to perform tasks within a particular context. Instead, the network must either inherit these nonlinear interactions from the input geometry, or – in the case where the input geometry has no nonlinear interactions – they must emerge when the input representation is through the initially random weights leading to the nonlinear units in the hidden layer.

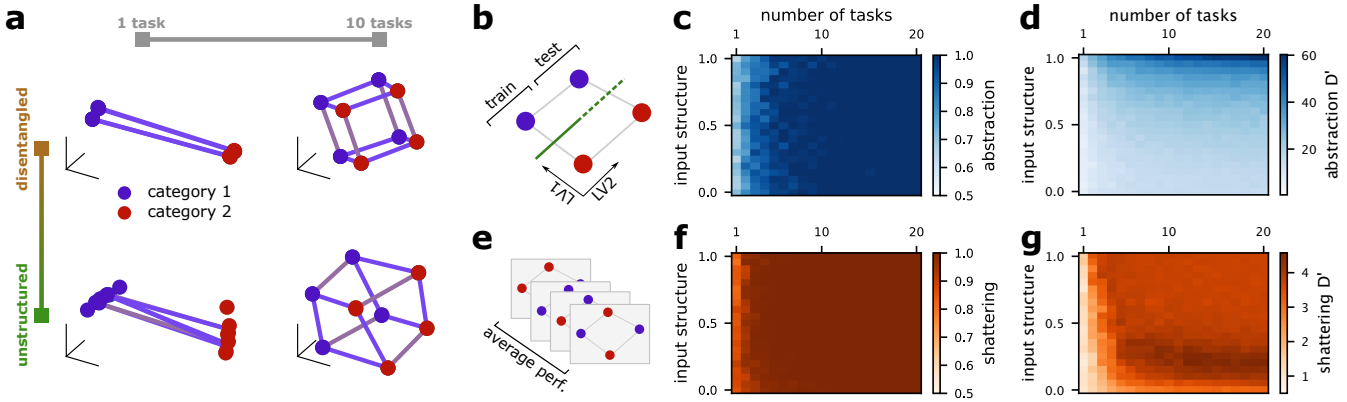

Figure 6: The trained networks have representational geometries within each context that depend on their training. **a** A grid of example models. The plot shows the geometry of representations within each context. The models trained to perform only one task develop a highly specialized geometry for that task. The models trained to perform more tasks represent the full set of latent variables relevant in that context. Each plot shows the representation within a single context, for which three latent variables are relevant. Eight points are shown in each plot. **b** Quantification of the level of abstraction in the representation within each context. Abstraction is measured with the cross-condition generalization performance, where a linear classifier is trained to perform a linear classification when one irrelevant feature is fixed at one value ("trained") and then tested when the irrelevant feature has the opposite value ("tested"). **c** The level of abstraction of models trained on different numbers of tasks and for different input geometries. For a sufficiently large number of tasks, the representations are always abstract, no matter the level of entanglement of the inputs. **d** The same as **c** but showing the average D' of the margin of the classifier for the generalization analysis. The average D' reflects both the input geometry and number of tasks. **e** The shattering dimensionality of representations in each context, measured by quantifying the performance of a linear decoder trained on random, often nonlinear categorizations of all the stimuli in the context. **f** The average performance of a classifier trained on the random classification tasks shown in **e** (i.e., the shattering dimensionality) for models with different input geometries and that were trained to perform different numbers of tasks. The shattering dimensionality is always above chance. **g** The same as **f** but showing the average margin D' of the classifier.

Finally, we investigate how the zero-shot generalization performance of our networks depends on both the input geometry and the number of trained tasks. We train the network to perform contextual tasks as before while holding the value of one irrelevant variable constant (fig. 7a, left). Then, we test the performance of the network when the value of that variable is changed (fig. 7a, right). First, we visualize the representational geometry of the learned configuration in a three-dimensional subspace found by PCA applied only to the trained points (fig. 7b, "trained"). Then, we project the never-before-seen points into the same subspace (fig. 7b, "tested"). We see that unstructured input geometries lead to a far greater change in the representation across learned and novel conditions (fig. 7b, left: low-dimensional; right: high-dimensional). The zero-shot generalization performance of the network is consistent with this visualization. We show that zero-shot performance depends primarily on the geometry of the input representations, rather than the number of tasks that the network is trained to perform (fig. 7c). This is because the nonlinear mixing in the input causes the decision variables to be represented in different subspaces given different values for the irrelevant variables.

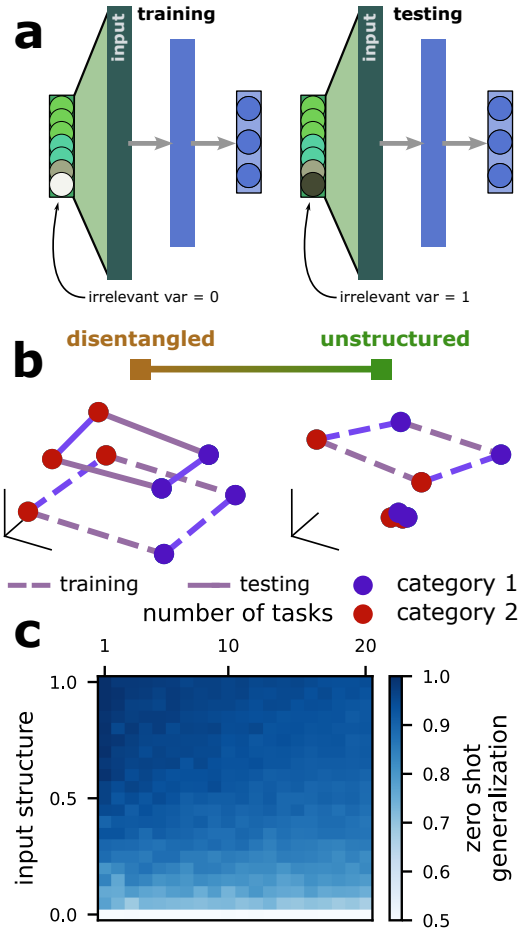

Figure 7: The zero-shot generalization performance of a model depends on the input geometry. **a** Schematic of a zero-shot generalization analysis for the full model, where the full contextual multi-tasking model is trained for a fixed value of an irrelevant variable, then its generalization performance is tested for the opposite value of that variable. **b** Visualization of how representations change within a single context from the learned set of stimuli to the novel set of stimuli, shown for fully disentangled (left) and unstructured (right) input geometries. **c** Zero-shot generalization performance of the models trained and tested with the procedure in **a**, shown for models trained on different numbers of tasks and with different input geometries.

2.6 The learned modular structure facilitates rapid learning

In the previous section, we showed how contextual multi-tasking models can generalize to novel situations without retraining depending on the geometry of their input representations. Here, we ask how the representational geometry learned by these models influences the speed with which they learn novel tasks or contexts.

First, we examine the learning of a novel task in two previously learned contexts (fig. 8a). In this case, we train each network to perform P tasks in $C = 2$ contexts. Then, after training on the original tasks, we train the network to perform a P+1th task using the same relevant latent variables. We measure how task performance changes across training on this novel task (fig. 8b, top). We show that the learning speed for a novel task depends on both the input geometry and the number of tasks that have been previously learned (fig. 8b, bottom). That is, both lower-dimensional input geometries and a larger number of previously trained tasks increase the speed of novel task learning. This is consistent with previous work showing that learning from lower-dimensional representations

is typically faster than learning from higher-dimensional representations[42, 50], so long as the low-dimensional representations allow for the classification to be learned.

Next, we investigate how the speed of learning for different kinds of novel contexts depends on both the input geometry and the number of previously trained tasks (fig. 8c). First, we investigate the learning of a related context (fig. 8c, "related") in which the novel context has a different contextual input than the previously learned contexts but shares all of the task decision rules with one of the two previously learned contexts (fig. 8c, "related" and "context 2"). We show that the learning speed for a related context depends primarily on the geometry of the input representations, where higher-dimensional input representations lead to slower learning in the novel, but related context.

Then, we investigate the learning speed of an unrelated context (fig. 8c, "unrelated"), in which the novel context has both a different contextual input than the previously learned contexts as well as different, randomly selected decision rules. In this case, the highest-dimensional input geometry provides faster learning than the low-dimensional input geometry (fig. 8e, top). However, the overall effect is more heterogeneous (fig. 8e, bottom). In particular, both decreasing the number of tasks and decreasing the dimensionality of input representations tend to increase learning speed.

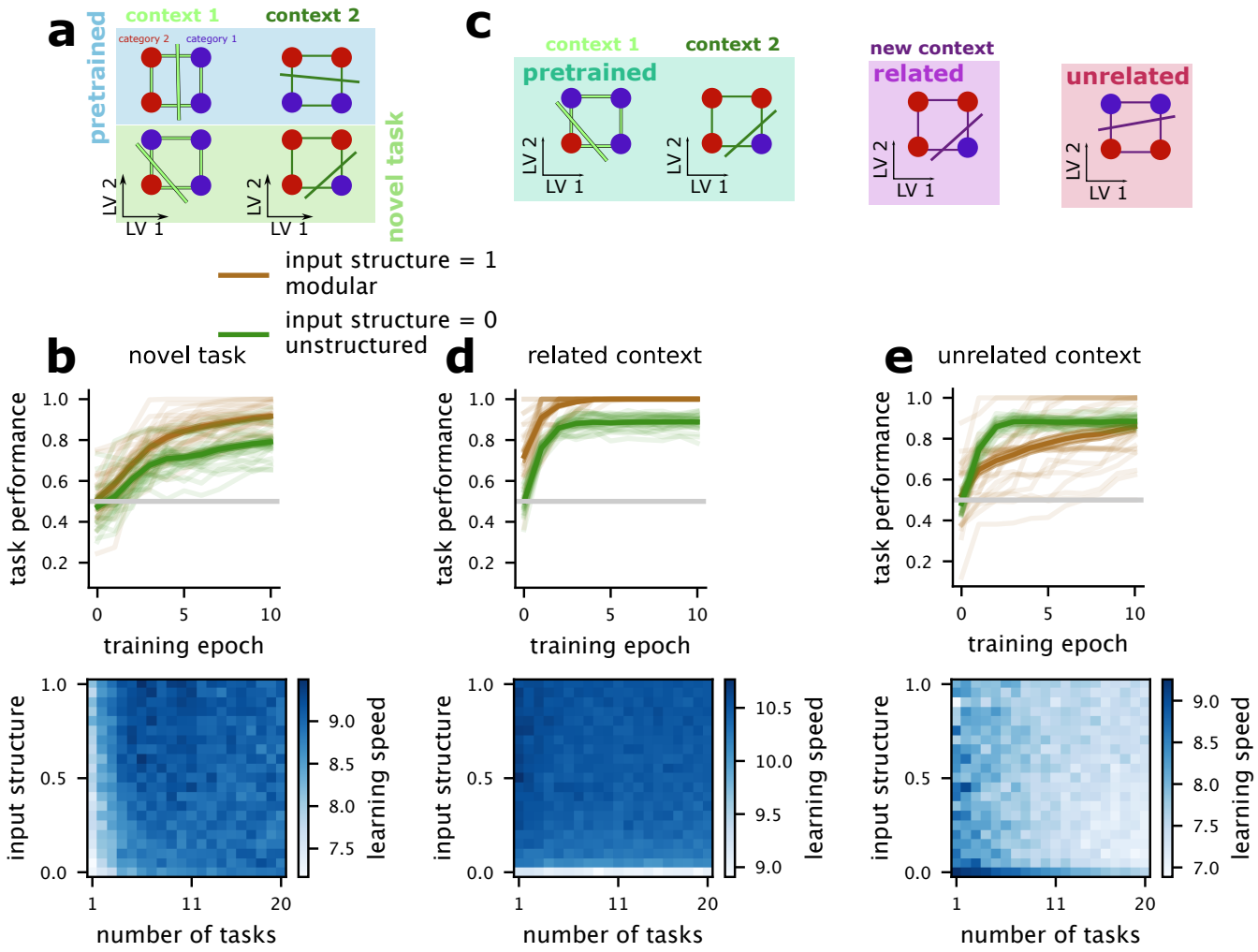

Figure 8: Modular representations provide rapid learning of new tasks and related contexts, but slower learning of unrelated contexts. **a** Schematic of the novel task learning analysis. The network is pretrained on P tasks and then trained to perform a $P+1$th task. **b** The learning trajectory of example networks with $P=1$ pretrained tasks and either fully disentangled or unstructured representations (top). The learning speed of networks for the $P+1$th task, given pretraining on different numbers of tasks and different input geometries. **c** Schematic of the related and unrelated context learning analyses. In both cases the network is pretrained in two contexts, and must learn to perform a third context. (left) The context is related to a previously learned context, and shares all the same task boundaries as one of the two pretrained contexts. (right) The context is unrelated to either pretrained context, as has randomly selected task boundaries. **d** The same as **b** but for the related context analysis. **e** The same as **b** but for the unrelated context analysis.

3 Discussion

We have shown how the learned representational geometry of an artificial neural network trained to perform contextual behavior depends on the geometry of its input and the number of tasks it must perform in each context. These learned geometries have been observed in the brain, and range from strikingly modular representations where individual units are only active in a specific context to completely unstructured, high-dimensional representations, which enable any task to be performed. We provide intuitive explanations for the transitions between these different regimes. We also explained the computational advantage of modularity in contextual tasks: context dependence almost always introduces non-linear separability, and the different modules reflect a possible decomposition of the full context-dependent task into subtasks (a subset of conditions) that are linearly separable. We have also demonstrated how our results change under different learning regimes (e.g., rich or lazy learning[28, 51]; see *Learned representations depend on the learning regime* in *Supplement* and fig. S1). Then, we show that these different learned representations have distinct computational benefits, enabling different forms of generalization. These results demonstrate that low- to moderate-dimensional input representations to networks trained to perform a handful of tasks yield benefits for both learning novel tasks and generalizing unseen stimuli. Finally, we demonstrate how these learned representations shape the learning of future contexts – and demonstrate differential benefits for learning related and unrelated contexts.

3.1 Predictions for experimental data

Our work makes several distinct predictions for experimental data. First, we predict that the representation of contextual and decision-related variables prior to training or in preceding brain regions will strongly influence how training shapes the representational geometry in the animal. In particular, for the high-dimensional representations often observed in frontal cortex[26, 27], relatively little change in the representations is necessary to facilitate task performance and therefore changes may be subtle – that is, our framework predicts that high-dimensional, unstructured representations will remain high-dimensional and unstructured unless the animal learns to perform several different tasks (or a more complex, multi-dimensional behavior) within each context. For these representations, it is sufficient to modify the readout, and even a simple linear readout can perform a very large number of different tasks without modifying the representations. In contrast, we predict much larger changes in representations when the original or preceding representations are relatively low-dimensional or do not already make the trained task linearly separable. We expect this condition to hold for some stimuli in frontal cortex, and representation in higher sensory cortices (such as visual cortex), where representations can be strikingly disentangled[43, 44]. In particular, not all representations in the brain need to change to support task performance – so, for instance, we do not expect large changes in earlier sensory regions, where task variables may not be linearly separable. These predictions emphasize the importance of longitudinal and multi-region recordings in neural data.

Further, our framework predicts that these different situations will lead to different levels of robustness to changes to the experimental framework. We expect unstructured representations to be associated with highly brittle behavior and a failure to generalize to subtle changes in the experimental setup. In contrast, we expect modular representations to be associated with a greater robustness to irrelevant changes (e.g. changes in latent variables that are not relevant for performing the task). This set of predictions could provide an explanation for what have remained largely anecdotal findings of brittleness to certain kinds of changes in experiments – for instance, animals

are often found to fail to generalize to even subtle changes in experiments yet regularly generalize in more naturalistic conditions. This could be because the variables that are relevant to more naturalistic behavior are low-dimensional and disentangled (as predicted by previous work [42]) while representations of various experimental variables are more high-dimensional. In addition, due to different training histories, different animals could have different input, or initial, representational geometries when they are being trained to perform the same task. These different initial geometries could lead to differences in representational geometry once the animal is fully trained; such differences have previously been shown to correlate with different behavioral strategies[52].

Finally, our framework predicts that different learned geometries will give rise to different learning speeds for novel behaviors – and that this effect will depend on whether the behavior is related to unrelated to previously learned behaviors. This prediction can be tested in experiments where an animal sequentially learns a series of interrelated tasks. In particular, if we find that the representational geometry underlying task performance is either explicitly or implicitly modular, then we expect the animal to rapidly learn related tasks but struggle to learn unrelated tasks; in contrast, if we find that the representation geometry is highly unstructured, then we expect that the animal will learn related and unrelated tasks at the same rate.

Our work significantly extends previous work focused on neural networks trained to perform contextual behavior[28, 34, 35, 53]. In particular, previous work has only considered low-dimensional input and single output (i.e., single task) conditions – while we consider the full interaction space between these two parameters. We believe that considering a much wider variety of input geometries is essential precisely because such a diversity of representational geometries has been found in experimental work – ranging from low-dimensional face representations[43, 44] to high-dimensional representations of cognitive variables in frontal cortex[26, 27]. This work provides a unified framework for understanding learning from these different starting points – and indicates that such diversity may be essential to explain the wide diversity of representational geometries reported to underlie contextual behavior in the literature[9, 20, 26]. Similarly, we believe that considering an increased number of trained tasks within each context is important because natural behavior is often richer than a single binary decision – instead, multiple different decisions are made about a stimulus at once, and these decisions produce multi-dimensional output that has so far remained relatively unstudied in experimental neuroscience.

One aspect of previous work that we have not extensively characterized here is the notion of compositionality[34, 53] in the learned representations. In prior recurrent neural network studies, they found that, in some cases, the same neural subspace would be used to represent the continuous value of a particular stimulus feature across different contexts. The networks trained to perform a contextual task that is related to one of several previously learned tasks are compositional. In this case, they re-use the same subspace used to perform the original task to perform the new task, and this re-use likely explains the faster learning for explicitly modular, as opposed to implicitly modular or unstructured representations.

Our work also only focuses on one form of contextual decision-making: when different contexts are associated with different task rules[4, 9, 19, 20, 54]. However, different contexts can also be associated with different thresholds for behavioral response[55] or biases in the stimuli that are shown[56]. Elucidating the computational principles that shape neural representations in these cases will require further work. However, it is possible that the initial representational geometry will be crucial in these cases as well, since the interaction between representations of decision variables and the representation of the context cue will also shape the kinds of solutions learned

in artificial neural networks. Thus, the principle of an initial representation shaping the learned representation is likely to still be important in this case.

Overall, our results provide an understanding of why different representational structure emerges in different situations. We argue that knowledge of the geometry of the initial representations of the task-relevant variables is essential for predicting the learned structure in the representations. This core prediction has already been validated in some experimental datasets, and underlines the importance of collecting data that allows the characterization of this initial geometry. Further, our work makes predictions for how these learned representations will, in turn, shape the learning of future behaviors and the ability to generalize to novel stimuli.

Acknowledgments: We are grateful to Matteo Alleman, Samuel Lippl, and members of the Center for Theoretical Neuroscience for useful discussions. We are grateful to Allison Ong and Ciela Sophia Chavez-Gilbride for administrative support. This work was supported by the following grants and foundations: NIH NINDS K99NS138578 (WJJ), Simons Foundation 542983SPI, Gatsby Charitable Foundation GAT3708, the Kavli Foundation, and the Swartz Foundation. We acknowledge computing resources from Columbia University's Shared Research Computing Facility project, which is supported by NIH Research Facility Improvement Grant 1G20RR030893-01, and associated funds from the New York State Empire State Development, Division of Science Technology and Innovation (NYSTAR) Contract C090171, both awarded April 15, 2010.

Author contributions: WJJ and SF designed the framework. WJJ performed the simulations and analyzed the models. WJJ made the figures. WJJ and SF wrote and edited the paper.

Competing interests: The authors declare no competing interests.

M1 Methods

M1 Latent variables

The input consists of $L = D+I+C$ binary latent variables, where D are the decision-related variables (i.e., variables that are used in classification tasks), I are irrelevant variables (i.e., variables that do not influence the target output), and C are context variables. Unless otherwise noted, there are $D = 3$ decision variables, $I = 4$ irrelevant variables, and $C = 2$ context variables. The decision and irrelevant variables are each independently sampled from a binomial distribution with $p = .5$. In contrast, the context variables are constrained to be one-hot across the C context dimensions, and are sampled with equal probability of each context being active. Instead of taking on the values 0 and 1 all variables are then remapped to take on the values -1 and 1 for notational convenience.

M2 Input model

Within the input model, all L latent variables are treated identically. We consider a spectrum of input geometries. Low-dimensional, disentangled representations are at one end of the spectrum and for samples $x \sim X$ where X is the distribution of latent variables (see *Latent variables* in *Methods*), the representation of the stimuli in the disentangled input model are given by,

$$r_{\text{disentangled}}(x) = Ax$$

where A is a $N_{\text{input}} \times L$ matrix, where N_{input} is the number of units in the input model and L is the number of latent variables. The rows of A are chosen to be orthogonal to each other with $N_{\text{input}} > L$

and the magnitude of the elements of A are chosen so that the sum of variance across $r_{\text{disentangled}}$ is 1.

High-dimensional, unstructured representations are at the other end of the spectrum, they are given by,

$$r_{\text{unstructured}}(x) = Mf(x)$$

where $f(x)$ produces vectors with length 2^L according to,

$$f_i(x) = [x_1 = t_i(1)]...[x_L = t_i(L)]$$

for $i \in [1, ..., n^L]$ where $[x = y]$ is the indicator function which is 1 when the x and y are the same and 0 otherwise, as used in prior work[26, 45, 57], and t_i is either -1 or 1. The matrix M is $N_{\text{input}} \times 2^L$ and $N_{\text{input}} > 2^L$ with all rows of M chosen to be orthogonal with scaling chosen so that the sum of variance across $r_{\text{unstructured}}$ is 1. Thus, each unique stimulus described by the latent variables L is represented in an independent dimension.

Finally, to produce a full spectrum of codes, we linearly interpolate between these two extreme cases, such that,

$$r_{\text{full}} = P\left(tr_{\text{disentangled}}(x) + (1-t)r_{\text{unstructured}}(x)\right)$$

where $t \in [0, 1]$ and P provides an overall scaling of the representation. Throughout the manuscript $P = 1$.

M3 Contextual tasks

The contextual tasks are given by,

$$t(x) = \begin{cases} \operatorname{sign} T_1 x & C_1 = 1 \\ \operatorname{sign} T_2 x & C_2 = 1 \end{cases}$$

where T_i are randomly sampled unit-length, L-dimensional vectors. Each T_i is sampled independently. Note that only one C_i can be 1 for any stimulus x, so the cases are mutually exclusive.

M4 Contextual multi-tasking model

The contextual multi-tasking model is a feedforward network with a single hidden layer. The inputs and target outputs are described above. The main parameters of the model are:

input dimensions	600
hidden units	200
output units	number of tasks, T
nonlinearity	ReLU
training examples	2000
training epochs	20
batch size	100
weight initialization	$\mathcal{U}(-a, a)$ with $a = \sqrt{\frac{6}{\text{in+out}}}$

The network is trained with the optimizer Adam and a learning rate of 1e-3.

M5 Dimensionality of unstructured solutions

The unstructured representation that we consider in the manuscript is primarily the maximum-dimensionality unstructured representation, where units have conjunctive responses for combinations of all L latent variables. This representation has dimensionality that scales exponentially with the number of latent variables, as $\dim(x) = 2^L$. While such high-dimensional interactions are necessary for a linear decoder to be able to learn any binary labeling of the stimuli, there are lower-dimensional but still unstructured representations that would allow a linear decoder to learn additional classes of labelings.

These other classes of unstructured representations can be described by the order of their interactions. A representation with order $O = L$ is the maximally unstructured representation from before. A representation with order $O = 2$ has only second-order interactions between latent variables. In particular, it has $\binom{L}{2}n^2$ interaction terms (e.g., an interaction between latent variable i and j, $[x_i = 0]\,[x_j = 0]$ where the brackets represent the indicator function as in *Input model* in *Methods*) and where n is the number of values that each latent variable can take on. In general, an unstructured representation with order O tuning will have $\binom{L}{O}n^O$ interaction terms. Importantly our measure of subspace specialization is robust to unstructured representations of different orders – that is, an unstructured representation of any order will still have subspace specialization equal to zero.

An $O = 2$ unstructured representation can be used to solve any contextual task. Thus, we calculate the dimensionality (described by the participation ratio) of this class of unstructured representation. The participation ratio can be written as [58],

$$\begin{aligned}\dim(x) &= \frac{(\operatorname{Tr} Z)^2}{\operatorname{Tr} Z^2} \\ &= \frac{D\,\langle Z_{ii}\rangle^2}{\langle Z_{ii}^2\rangle + (D-1)\langle Z_{ij}^2\rangle}\end{aligned}$$

where D is the number of dimensions in the input and Z is the $D \times D$ input dimension covariance matrix, across all stimuli.

We calculate,

$$\langle Z_{ii}\rangle^2 = \langle Z_{ii}^2\rangle = \frac{1}{n^{2O}}\left(1 - \frac{1}{n^O}\right)^2$$

and

$$(D-1)\,\langle Z_{ij}^2\rangle = \frac{n^O - 1}{n^{4O}} + \sum_{r=1}^{O-1}\binom{O}{r}\binom{L-O}{O-r}\frac{n^r - 1}{n^{3O}}$$

So, putting it together, we obtain,

$$\dim(L, O) = \frac{\binom{L}{O}n^O\left(1 - \frac{1}{n^O}\right)^2}{1 + \frac{n^O - 1}{n^{2O}} + \frac{1}{n^O}\sum_{r=1}^{O-1}\binom{O}{r}\binom{K-O}{O-r}(n^r - 1)}$$

where L is the number of latent variables and O is the order of the interactions.

For $O = 2$, we find that,

$$\dim(L, 2) = \frac{\binom{L}{2} 9}{3 + 2(L - 2)}$$

which will grow proportionally to L for large L.

M6 Clustering analysis

We assess whether the units in the hidden layer of the contextual multi-tasking model are best explained by discrete clusters in the space of average activity within each context. To do so, we sample 1000 stimuli from each context and take the average activity of each unit within each context. This produces a matrix of dimensions $\mu_C = N_{\text{hidden}} \times C$. Then, we perform Gaussian mixture model clustering on this matrix, where each unit is assigned to one of K discrete clusters. We perform this procedure for $K = 1$ to $K = 5$. Then, we evaluate the goodness of fit of each clustering with the Bayesian information criterion (BIC)[59], and choose the clustering with the highest BIC. We use these clusters to color the points in fig. 2d and similar plots. Where only one color is present, this means a single cluster provided the best fit to the data.

M7 Contextual fraction

We define the context fraction in the same mean activity space described above for the clustering analysis. Here, after computing μ_C, we then count the rows of the matrix that have mean activity < 0.01 in all contexts except one, this count is then divided by the number of units in the hidden layer.

M8 Subspace specialization

We define subspace specialization to capture whether the population activity in the hidden layer of the contextual multi-tasking model has low-dimensional structure related to context or if it is simply high-dimensional and unstructured. The measure is designed to be 0 if there is no unique low-dimensional structure related to context and positive otherwise. The measure is based on the alignment index, which has been introduced previously in [60]. The alignment index quantifies the amount of overlap between the subspaces described by two sets of basis vectors, U_1 and U_2, which are both $N \times P_i$ matrices where P_i is the number of basis vectors. The alignment index for U_1 and U_2 is,

$$a_{12} = \frac{\text{Tr}\left(U_1^T U_2 U_2^T U_1\right)}{\min P_1 P_2}$$

A value of 1 indicates that the two sets basis vectors span precisely the same subspace, while a value of 0 indicates that they span non-overlapping subspaces of the full population activity space.

To compute our subspace specialization measure, we compute $D + I$ non-context alignment indices and $\binom{C}{2}$ context-related alignment indices. For each latent variable (or context pair), we sample 1000 stimuli and then split into two subsets, where the value of the latent variable is zero (or the first context is active) in one subset and where the value of the latent variable is 1 (or the second context is active) in the other subset. Then, we perform PCA separately on the two subsets, and keep all dimensions for each that explain more than 10^{-10} of the variance. This gives us our sets

of basis vectors, U_1 and U_2. We take the alignment index between these sets of vectors, as defined above. Now, we have $D + I$ overlap measurements for non-context variables, $o_{\text{noncontext}}$, and $\binom{C}{2}$ overlap measurements for context pairs, o_{context}. We summarize these measurements into a single subspace specialization measure by taking the average of each group, $\bar{o}_{\text{noncontext}}$ and $\bar{o}_{\text{context}}$, followed by the rectified log-ratio:

$$s = \left[\log \frac{\bar{o}_{\text{noncontext}}}{\bar{o}_{\text{context}}}\right]_+$$

for the subspace specialization index. This index will be zero when the alignment for the context pairs and non-context variables is the same. It will also be zero (due to rectification) when the context variables have greater alignment than the non-context variables. However, it will be positive when the context variables have lower alignment than the non-context variables. This indicates that the activity in different contexts is confined to subspaces that are distinct relative to that of other variables encoded by the same system.

M9 Alternate decompositions

We want to calculate the probability that a the decomposition along a given dimension is eliminated (i.e., made so that decomposing along that dimension does not yield linearly separable sub-tasks) by a randomly selected task. We simplify our tasks for this calculation by assuming that they are all aligned with a single decision-relevant latent variable. We do not make this assumption when testing our prediction.

For a given task and with D relevant latent variables, there are three possible situations. With probability $\frac{1}{2D}$, the same decision-related variable and direction along that variable will be selected. This whole task is already linearly separable and it does not eliminate any dimensions. With probability $\frac{1}{2D}$, the same decision-related variable but different directions will be selected. This eliminates all variables as possible decompositions except for the selected variable. With probability $1 - \frac{1}{D}$, two different decision-related variables will be selected. This eliminates all variables except for these two variables.

To put these together, we compute the probability that a particular variable is not eliminated after a new task is added. This is the sum of the probability that this particular variable is chosen at least once or it is not chosen at all, but the the same variable and direction is chosen. The probability that it is chosen at least once is,

$$1 - \left(\frac{D-1}{D}\right)^2$$

and the probability that the same other variable is chosen twice in the same direction is,

$$(D-1)\frac{1}{2D^2}$$

So, together, the probability that a given variable is not eliminated after a single task is selected is,

$$p_{\text{safe}} = 1 - \left(\frac{D-1}{D}\right)^2 + (D-1)\frac{1}{2D^2}$$

and, because each task is selected independently, the probability that it eliminated after the selection of T tasks is,

$$p_{\text{elim}} = 1 - p_{\text{safe}}^T$$

where T is the number of tasks. Finally, the expected number of alternate decompositions is then

$$D p_{\text{safe}}^T$$

M10 Dimensionality of the required output

We compute the participation ratio of the required output, x averaged across all stimuli. To approach this, we employ the expression for the participation ratio developed in [58],

$$\begin{aligned}\dim(x) &= \frac{(\operatorname{Tr} Z)^2}{\operatorname{Tr} Z^2} \\ &= \frac{T\,\langle Z_{ii}\rangle^2}{\langle Z_{ii}^2\rangle + (T-1)\langle Z_{ij}^2\rangle}\end{aligned}$$

where T is the number of tasks and Z is the $T \times T$ task covariance matrix, across samples from all contexts. The diagonal elements of the covariance matrix Z_{ii} are all defined to be 1, so $\langle Z_{ii}^2\rangle = \langle Z_{ii}\rangle^2 = 1$. The off-diagonal elements require calculation. So, we want to calculate $\langle Z_{ij}^2\rangle$ for

$$Z_{ij} = \frac{1}{C}\sum_c^C \frac{1}{N}\sum_k^N t_i(x_k^c) t_j(x_k^c)$$

where C is the number of contexts, N is the number of samples taken from each context, and x_k^c is the k-th sample from context c.

First, we simplify our tasks so that they are aligned with the decision-related variables in each context. We do not make this assumption elsewhere in the paper, nor when verifying the predictions that arise from this calculation. In this simplified setting, the average product within each context can take on three different values: 1 when tasks i and j are the same (i.e., they are aligned with the same decision variable D and pointing in the same direction), and this happens with probability $\frac{1}{2D}$; 0 when the tasks are orthogonal (i.e., aligned with different decision variables), and this happens with probability $1 - \frac{1}{D}$; and -1 when the tasks are anti-parallel (i.e., aligned with the same decision variable, but pointing in different directions), and this happens with probability $\frac{1}{2D}$.

So, we want to calculate,

$$\begin{aligned}\langle Z_{ij}^2\rangle &= \left\langle \left(\frac{1}{C}\sum_c^C \theta_c\right)^2 \right\rangle \\ &= \frac{1}{C^2}\left\langle \sum_c^C \theta_c^2 + \sum_{m\neq n}^{C(C-1)} \theta_m\theta_n \right\rangle \\ &= \frac{1}{C^2}\sum_c^C \langle\theta_c^2\rangle + \sum_{m\neq n}^{C(C-1)} \langle\theta_m\theta_n\rangle\end{aligned}$$

where

$$\theta_c = \begin{cases} 0 & p = 1 - \frac{1}{D} \\ 1 & p = \frac{1}{2D} \\ -1 & p = \frac{1}{2D} \end{cases}$$

So,

$$\begin{aligned} \left\langle \theta_c^2 \right\rangle &= 1^2 \frac{1}{2D} + (-1)^2 \frac{1}{2D} \\ &= \frac{1}{D} \end{aligned}$$

and

$$\begin{aligned} \langle \theta_m \theta_n \rangle &= \langle \theta_m \rangle \langle \theta_n \rangle \\ &= 0 \end{aligned}$$

which yields

$$\begin{aligned} \langle Z_{ij}^2 \rangle &= \frac{1}{C^2} \frac{C}{D} \\ &= \frac{1}{CD} \end{aligned}$$

Putting everything together, we find

$$\dim(x) = \frac{T}{1 + \frac{T-1}{CD}}$$

M11 Cross-condition generalization performance

To compute the cross-condition generalization performance (CCGP) within a particular context, we first select one decision-relevant latent variable for decoding. Then, we train a decoder to decode the value of that variable given 8 stimulus representations sampled from the same context and where all other decision-related variables have a fixed value. Finally, we test the same decoder on 8 stimulus representations sampled from the same context but where the other decision-related values never have the their fixed value from before. We repeat this 10 times and report the average performance of this decoder as the CCGP. All samples have the same noise level as used in network training.

M12 Shattering dimensionality

To compute the shattering dimensionality within each context, we select all balanced dichotomies of the stimuli based on their decision-relevant features. Then, we train a linear decoder to perform that classification task using 1000 samples and report its performance on 1000 test samples with different noise. All samples have the same noise level as used in network training.

References

1. O'Neill, P.-K. *et al.* The representational geometry of emotional states in basolateral amygdala. *bioRxiv,* 2023–09 (2023).
2. Boyle, L. M., Posani, L., Irfan, S., Siegelbaum, S. A. & Fusi, S. Tuned geometries of hippocampal representations meet the computational demands of social memory. *Neuron* **112,** 1358–1371 (2024).
3. Nogueira, R., Rodgers, C. C., Bruno, R. M. & Fusi, S. The geometry of cortical representations of touch in rodents. *Nature Neuroscience* **26,** 239–250 (2023).
4. Bernardi, S. *et al.* The geometry of abstraction in the hippocampus and prefrontal cortex. *Cell* **183,** 954–967 (2020).
5. Courellis, H. S. *et al.* Abstract representations emerge in human hippocampal neurons during inference. *Nature,* 1–9 (2024).
6. Hirokawa, J., Vaughan, A., Masset, P., Ott, T. & Kepecs, A. Frontal cortex neuron types categorically encode single decision variables. *Nature* **576,** 446–451 (2019).
7. Hocker, D. L., Brody, C. D., Savin, C. & Constantinople, C. M. Subpopulations of neurons in lOFC encode previous and current rewards at time of choice. *Elife* **10,** e70129 (2021).
8. Hardcastle, K. *et al.* A Multiplexed , Heterogeneous , and Adaptive Code for Navigation in Medial Entorhinal Cortex. *Neuron* **94,** 1–13. ISSN: 0896-6273. `http://dx.doi.org/10.1016/j.neuron.2017.03.025` (2017).
9. Lee, J. J., Krumin, M., Harris, K. D. & Carandini, M. Task specificity in mouse parietal cortex. *Neuron* **110,** 2961–2969 (2022).
10. Sun, W. *et al.* Learning produces a hippocampal cognitive map in the form of an orthogonalized state machine. *bioRxiv,* 2023–08 (2023).
11. Strotzer, M. One century of brain mapping using Brodmann areas. *Clinical Neuroradiology* **19,** 179 (2009).
12. Van Essen, D. C. & Maunsell, J. H. Hierarchical organization and functional streams in the visual cortex. *Trends in neurosciences* **6,** 370–375 (1983).
13. Rakic, P. Specification of cerebral cortical areas. *Science* **241,** 170–176 (1988).
14. Konkle, T. Emergent organization of multiple visuotopic maps without a feature hierarchy. *bioRxiv,* 2021–01 (2021).
15. Bullmore, E. & Sporns, O. The economy of brain network organization. *Nature reviews neuroscience* **13,** 336–349 (2012).
16. Bakhtiari, S., Mineault, P., Lillicrap, T., Pack, C. & Richards, B. The functional specialization of visual cortex emerges from training parallel pathways with self-supervised predictive learning. *Advances in Neural Information Processing Systems* **34,** 25164–25178 (2021).
17. Desimone, R. & Duncan, J. Neural mechanisms of selective visual attention. *Annual review of neuroscience* **18,** 193–222 (1995).
18. Miller, E. K. & Cohen, J. D. An integrative theory of prefrontal cortex function. *Annual review of neuroscience* **24,** 167–202 (2001).
19. Okazawa, G. & Kiani, R. Neural Mechanisms that Make Perceptual Decisions Flexible. *Annual Review of Physiology* **85,** 191–215 (2023).
20. Mante, V., Sussillo, D., Shenoy, K. V. & Newsome, W. T. Context-dependent computation by recurrent dynamics in prefrontal cortex. *nature* **503,** 78–84 (2013).
21. Cole, M. W. *et al.* Multi-task connectivity reveals flexible hubs for adaptive task control. *Nature neuroscience* **16,** 1348–1355 (2013).

22. Koida, K. & Komatsu, H. Effects of task demands on the responses of color-selective neurons in the inferior temporal cortex. *Nature neuroscience* **10,** 108–116 (2007).
23. Chowdhury, S. A. & DeAngelis, G. C. Fine discrimination training alters the causal contribution of macaque area MT to depth perception. *Neuron* **60,** 367–377 (2008).
24. Rodgers, C. C. & DeWeese, M. R. Neural correlates of task switching in prefrontal cortex and primary auditory cortex in a novel stimulus selection task for rodents. *Neuron* **82,** 1157–1170 (2014).
25. Okazawa, G., Hatch, C. E., Mancoo, A., Machens, C. K. & Kiani, R. Representational geometry of perceptual decisions in the monkey parietal cortex. *Cell* **184,** 3748–3761 (2021).
26. Rigotti, M. *et al.* The importance of mixed selectivity in complex cognitive tasks. *Nature* **497,** 585–590 (2013).
27. Fusi, S., Miller, E. K. & Rigotti, M. Why neurons mix: high dimensionality for higher cognition. *Current opinion in neurobiology* **37,** 66–74 (2016).
28. Flesch, T., Juechems, K., Dumbalska, T., Saxe, A. & Summerfield, C. Orthogonal representations for robust context-dependent task performance in brains and neural networks. *Neuron* **110,** 1258–1270 (2022).
29. Panichello, M. F. & Buschman, T. J. Shared mechanisms underlie the control of working memory and attention. *Nature* **592,** 601–605 (2021).
30. Xie, Y. *et al.* Geometry of sequence working memory in macaque prefrontal cortex. *Science* **375,** 632–639 (2022).
31. Khosla, M., Williams, A. H., McDermott, J. & Kanwisher, N. Privileged representational axes in biological and artificial neural networks. *bioRxiv,* 2024–06 (2024).
32. Dubreuil, A., Valente, A., Beiran, M., Mastrogiuseppe, F. & Ostojic, S. The role of population structure in computations through neural dynamics. *Nature neuroscience* **25,** 783–794 (2022).
33. Ostojic, S. & Fusi, S. Computational role of structure in neural activity and connectivity. *Trends in Cognitive Sciences* (2024).
34. Driscoll, L., Shenoy, K. & Sussillo, D. Flexible multitask computation in recurrent networks utilizes shared dynamical motifs. *bioRxiv* (2022).
35. Saxe, A., Sodhani, S. & Lewallen, S. J. *The neural race reduction: Dynamics of abstraction in gated networks* in *International Conference on Machine Learning* (2022), 19287–19309.
36. Kaufman, M. T. *et al.* The implications of categorical and category-free mixed selectivity on representational geometries. *Current Opinion in Neurobiology* **77,** 102644 (2022).
37. Prince, J. S., Alvarez, G. A. & Konkle, T. Contrastive learning explains the emergence and function of visual category-selective regions. *Science Advances* **10,** eadl1776 (2024).
38. Minxha, J., Adolphs, R., Fusi, S., Mamelak, A. N. & Rutishauser, U. Flexible recruitment of memory-based choice representations by the human medial frontal cortex. *Science* **368,** eaba3313 (2020).
39. Saxena, S. & Cunningham, J. P. Towards the neural population doctrine. *Current opinion in neurobiology* **55,** 103–111 (2019).
40. Ebitz, R. B. & Hayden, B. Y. The population doctrine in cognitive neuroscience. *Neuron* **109,** 3055–3068 (2021).
41. Chung, S. & Abbott, L. Neural population geometry: An approach for understanding biological and artificial neural networks. *Current opinion in neurobiology* **70,** 137–144 (2021).
42. Johnston, W. J. & Fusi, S. Abstract representations emerge naturally in neural networks trained to perform multiple tasks. *Nature Communications* **14,** 1040 (2023).
43. Chang, L. & Tsao, D. Y. The code for facial identity in the primate brain. *Cell* **169,** 1013–1028 (2017).

44. She, L., Benna, M. K., Shi, Y., Fusi, S. & Tsao, D. Y. Temporal multiplexing of perception and memory codes in IT cortex. *Nature,* 1–8 (2024).
45. Johnston, W. J., Fine, J. M., Yoo, S. B. M., Ebitz, R. B. & Hayden, B. Y. Semi-orthogonal subspaces for value mediate a binding and generalization trade-off. *Nature Neuroscience,* 1–13 (2024).
46. Alleman, M., Lindsey, J. W. & Fusi, S. Task structure and nonlinearity jointly determine learned representational geometry. *arXiv preprint arXiv:2401.13558* (2024).
47. Swaminathan, S. K. & Freedman, D. J. Preferential encoding of visual categories in parietal cortex compared with prefrontal cortex. *Nature neuroscience* **15,** 315–320 (2012).
48. Sarma, A., Masse, N. Y., Wang, X.-J. & Freedman, D. J. Task specific versus generalized mnemonic representations in parietal and prefrontal cortices.
49. Freedman, D. J. & Assad, J. A. Experience-dependent representation of visual categories in parietal cortex. *Nature* **443,** 85 (2006).
50. Higgins, I. *et al.* Scan: Learning hierarchical compositional visual concepts. *arXiv preprint arXiv:1707.03389* (2017).
51. Farrell, M., Recanatesi, S. & Shea-Brown, E. From lazy to rich to exclusive task representations in neural networks and neural codes. *Current Opinion in Neurobiology* **83,** 102780 (2023).
52. Fascianelli, V. *et al.* Neural representational geometries reflect behavioral differences in monkeys and recurrent neural networks. *Nature Communications* **15,** 6479 (2024).
53. Yang, G. R., Joglekar, M. R., Song, H. F., Newsome, W. T. & Wang, X.-J. Task representations in neural networks trained to perform many cognitive tasks. *Nature neuroscience* **22,** 297–306 (2019).
54. Siegel, M., Buschman, T. J. & Miller, E. K. Cortical information flow during flexible sensorimotor decisions. *Science* **348,** 1352–55. ISSN: 0036-8075. arXiv: `arXiv:1011.1669v3`. `http://www.sciencemag.org/content/348/6241/1352.full` (2015).
55. Crapse, T. B., Lau, H. & Basso, M. A. A role for the superior colliculus in decision criteria. *Neuron* **97,** 181–194 (2018).
56. Hanks, T. D., Mazurek, M. E., Kiani, R., Hopp, E. & Shadlen, M. N. Elapsed decision time affects the weighting of prior probability in a perceptual decision task. *Journal of Neuroscience* **31,** 6339–6352 (2011).
57. Johnston, W. J., Palmer, S. E. & Freedman, D. J. Nonlinear mixed selectivity supports reliable neural computation. *PLoS computational biology* **16,** e1007544 (2020).
58. Litwin-Kumar, A., Harris, K. D., Axel, R., Sompolinsky, H. & Abbott, L. Optimal Degrees of Synaptic Connectivity. *Neuron* **0,** 1153–1164.e7. `http://linkinghub.elsevier.com/retrieve/pii/S0896627317300545` (2017).
59. Neath, A. A. & Cavanaugh, J. E. The Bayesian information criterion: background, derivation, and applications. *Wiley Interdisciplinary Reviews: Computational Statistics* **4,** 199–203 (2012).
60. Elsayed, G. F., Lara, A. H., Kaufman, M. T., Churchland, M. M. & Cunningham, J. P. Reorganization between preparatory and movement population responses in motor cortex. *Nature communications* **7,** 13239 (2016).

S1 Supplement

S1 Learned representations depend on the learning regime

We have focused on a rich learning regime in the rest of the paper. In this regime, the representations change appreciably as the network is trained. Intuitively, modular representations are less likely to

emerge in the lazy learning regime, as the representation in the network will not change as much over training. To give a sense of how our results depend on the learning regime, we train three sets of networks with different weight initialization statistics. The first set is close to the one investigated in most of the paper. This rich regime network is has its weights initialized following $w_ij \sim \mathcal{N}(0, \sigma)$ with $\sigma = .001$ (fig. S1a). The second set has larger initializations and is closer to the lazy regime ($\sigma = .02$; fig. S1b). The third set is trained in an even lazier regime ($\sigma = .8$; fig. S1c). The change of learning regime shifts the threshold input geometry where the network transitions from learning explicitly modular representations to learning unstructured or implicitly modular representations; however the results are otherwise qualitatively similar.

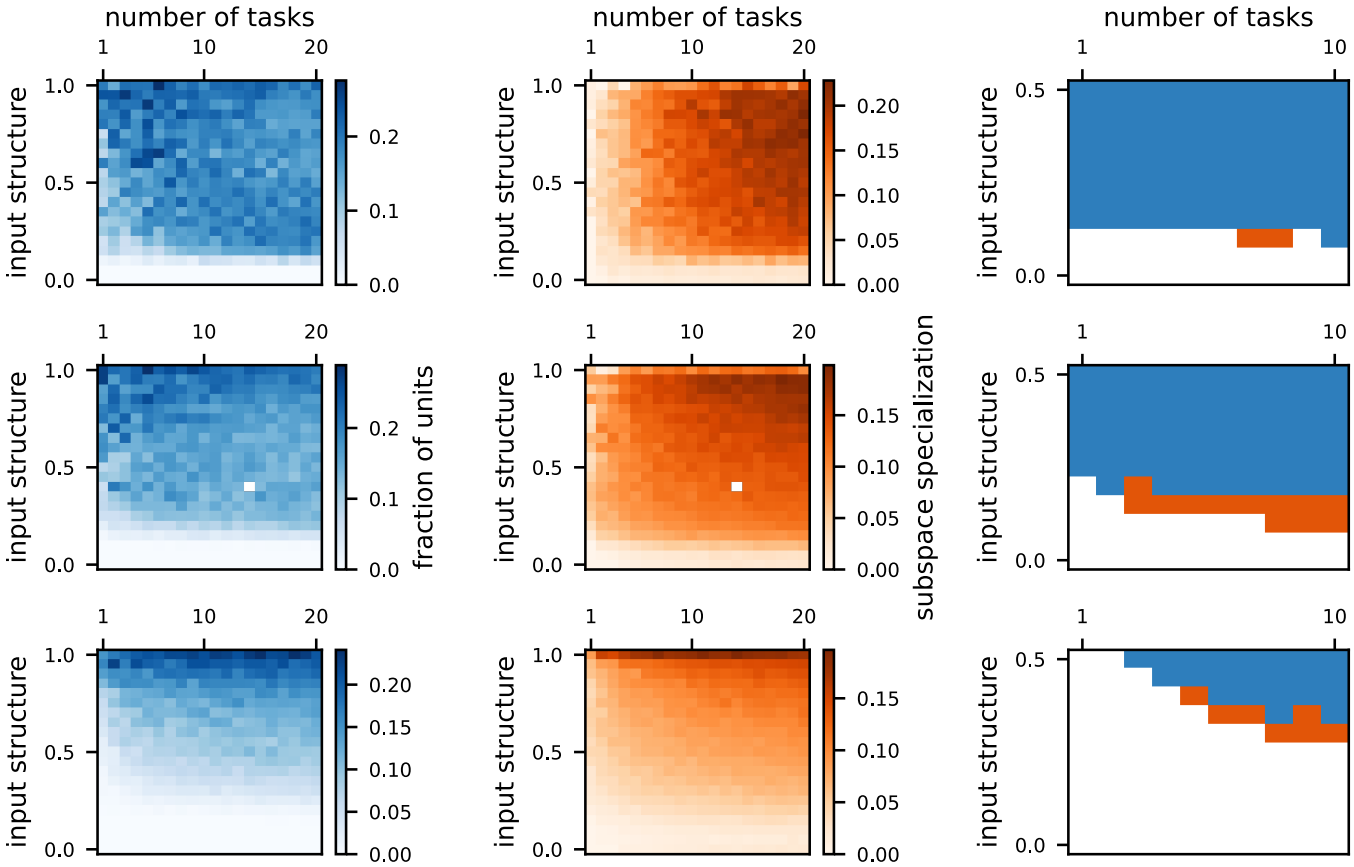

Figure S1: The learning regime of the network influences the learned representations. **a** Results for networks trained with small weight initialization ($\sigma = .001$). (left) The explicit modularity. (middle) The implicit modularity. (right) Threshold transition plot, focusing on a subset of the full parameter spacer. **b** The same as **a** but for larger weight initialization ($\sigma = .02$). **c** The same as **a** but for an even larger weight initialization ($\sigma = .8$).