Neural dynamics and geometry for transitive inference

Kenneth Kay[1,2,3], Xue-Xin Wei[4], Ramin Khajeh[1,2], Manuel Beiran[1,2], Christopher J. Cueva[5], Greg Jensen[1,6,7], Vincent P. Ferrera[1,6,8] & L.F. Abbott[1,2]

Abstract

The ability to make inferences using abstract rules and relations has long been understood to be a hallmark of human intelligence, as evidenced in logic, mathematics, and language. Intriguingly, modern work in animal cognition has established that this ability is evolutionarily widespread, indicating an ancient and possibly foundational role in natural intelligence. Despite this importance, it remains an open question how inference using abstract rules is implemented in the brain — possibly due to a lack of competing hypotheses at the level of collective neural activity and of behavior. Here we report the generation and analysis of a collection of neural networks (NNs) that perform transitive inference (TI), a classical cognitive task that requires inference of a single abstract relation between novel combinations of inputs (if A > B and B > C, then A > C). We found that NNs generated using standard training methods (i) generalize fully (i.e. to all novel combinations of inputs), (ii) generalize when inference requires working memory (WM), a capacity thought to be essential for inference in living subjects, (iii) express multiple emergent behaviors long documented in humans and animals, in addition to novel behaviors not previously studied, and (iv) adopt different solutions that yield alternative predictions for both behavior and collective neural activity. Further, a subset of NNs expressed a "subtractive" solution that was characterized in neural activity space by a simple dynamical pattern (an oscillation) and geometric arrangement (ordered collinearity). Together, these findings show how collective neural activity can accomplish generalization according to an abstract rule, and provide a series of testable hypotheses not previously established in the study of TI. More broadly, these findings suggest new ways to understand how neural systems realize abstract rules and relations.

1 Introduction

Cognitive faculties such as logical reasoning, mathematics, and language have long been recognized to be characteristic of human-level intelligence. Common to these faculties is abstraction: the ability to generalize prior knowledge and experience to novel circumstances. Importantly, abstraction typically entails understanding particular relationships, e.g. "adjacent to", "same as", "relevant to") between items (e.g. stimuli, objects, behaviors, words, variables), which can then be used to infer equivalent relationships between items not previously observed together, i.e. *completely novel combinations of items*. Such relational inferences – which can be understood as entailing the learning and use of abstract rules – are the basis of a structured form of knowledge, often termed a "schema," that is thought to enable humans to generalize in systematic and meaningful ways [1, 2, 3, 4, 5, 6], and thus has been posited as essential to advanced cognition.

Intriguingly, recent insights from landmark work in animals [7, 8, 9, 10, 11, 12] indicates that cognition based on relations is more prevalent and thus possibly more fundamental than previously appreciated. This prevalence is evidenced by the observation that cognitive abilities that entail relational inference – such as navigation [13, 14, 15], learning-to-learn [9, 16, 17], and concept/structure learning [11, 12, 18] – are in fact widespread across the animal kingdom. Further, these abilities have been linked to memory systems in the brain – variously termed "relational memory", "cognitive maps", "learning sets", among others – that enable humans and animals alike to make systematic inferences [19, 20, 21, 5], generalize across different domains [16, 22, 23], learn rapidly [24, 25, 26, 27, 28], and plan and envision new experiences [29, 30, 31, 32, 33, 34, 35, 36]. These findings and insights extend the scope of relational inference to a wide range of species and cognitive abilities, and, further, imply that there exists a deep interrelationship between relational inference and memory.

[1]Mortimer B. Zuckerman Mind Brain Behavior Institute, Columbia University, NY, [2]Center for Theoretical Neuroscience, Columbia University, NY, [3]Grossman Center for the Statistics of Mind, Columbia University, NY, [4]Departments of Neuroscience and Psychology, The University of Texas at Austin, TX, [5]Department of Brain and Cognitive Sciences, MIT, MA, [6]Department of Neuroscience, Columbia University Medical Center, NY, [7]Department of Psychology at Reed College, OR, [8]Department of Psychiatry, Columbia University Medical Center, NY. Correspondence: kk3291@columbia.edu

Despite this unifying importance, it remains an open question how relational inference is implemented in neural systems, whether in artificial networks (e.g. those performing linguistic [37, 38, 39] or symbolic [40, 41] tasks) or in the brain. Toward answering this question, a fundamental scientific aim is to identify or generate putative neural implementations that can be used to derive multiple experimentally testable hypotheses. In particular, hypotheses at the level of behavior and of collective (population-level) neural activity may be crucial given that these levels have proved decisive in clarifying whether and how neural systems in the brain implement various cognitive functions (e.g. vision, movement, timing, decision-making [42, 43, 44, 45, 46]). Notably, in both neurobiology and machine intelligence, relational inference is often studied in relatively complex cases (e.g. spatial [16, 21, 24, 47, 48, 49] and linguistic [38, 37, 50] knowledge), for which it may be relatively difficult to formulate or generate hypotheses derived from neural implementations, especially at the level of neural populations and of behavior. In this way, studying simpler cases may be advantageous or even crucial towards understanding relational inference in the brain. We therefore took a two-part approach intended to yield such hypotheses: first, we stipulated a task paradigm that distills relational inference into a simple yet essential form, and second, we adopted a methodology suited to discover possible population-level and behavior-relevant neural implementations thereof.

2 Transitive inference: a classic cognitive task

We first sought to operationalize relational inference in a task that was (i) reduced to a single abstract relation and (ii) based on stimuli (e.g. images) that can be presented with temporal precision. We reasoned that each of these properties might be critically important, in that (i) reduces complexity, which could ultimately enable discovering neural implementations, and (ii) delimits periods of sensory-driven neural activity (and by extension neural activity underlying inference), thus facilitating subsequent interpretation of observed neural activity.

A classic task paradigm capturing these properties is **transitive inference** (**TI**) [10, 51, 52, 53, 54], which tests a subject's ability to use premises A $>$ B and B $>$ C to infer A $>$ C — an instance of an abstract rule (i.e. choose items that are 'higher', based on the transitive '$>$' relation). TI operationalizes a simple yet powerful schema (**Fig. 1a**) that enables generalization from N premises (training cases) to order N^2 probes (test cases) in accordance with the relation-based rule (i.e. choose the 'higher' item), thus testing generalization based on the schema ('schematization') rather than interpolation or extrapolation (manifested in the pattern of correct responses in the task, **Fig. 1b**). The transitive '$>$' relation is itself a component of various types of relations, and is of fundamental importance in symbolic reasoning – together suggesting that TI is not essentially based on specific stimuli (or stimulus features) in isolation.

At a broader level, TI can be understood as a simplified alternative to other paradigms testing relational inference but involving multiple abstract rules (e.g. linguistic syntax and semantics) and/or task stimuli that are relatively challenging to isolate (e.g. spatial paradigms [16, 21, 24, 47]). Further, in contrast to approaches that focus mainly on lower-level neurobiological phenomena (e.g. neural firing that has abstract correlates [14, 22, 55]), TI requires direct behavioral report of successful inference, thus affording identification of potentially critical relationships between behavior and underlying neural implementations.

Remarkably, though TI is a classic task in behavioral psychology [10, 51, 53, 56, 57] and indeed a cornerstone of symbolic reasoning, the neural basis of TI has remained unclear [53, 58, 59, 60], potentially due to a lack of putative neural implementations and testable hypotheses.

3 A neural approach to transitive inference

Despite the long history of TI as a cognitive task, investigation of its neural basis in the brain is relatively recent [52, 61, 62, 63, 64, 65]. Prior work implies that an approach seeking to identify biologically accurate neural implementations would benefit from three criteria:

First, an approach should impose **minimal architectural constraints**. TI is observed in an extremely broad range of species, including primates, birds, rodents, and insects [53, 59, 60, 66, 67, 68]; possibly the result of convergent evolution [69]), a striking ubiquity that implies that highly specialized neural architecture may not be essential.

Second, an approach should explicitly require memory across time, particularly **working memory (WM)** [70, 71, 72, 73]. In living subjects, relational inferences such as TI characteristically rely on memory since subjects must assess relationships between events not experienced simultaneously (e.g. sensory stimuli separated in time) – memory enables such events to be in effect brought together. WM in particular is thought to be essential to relational inference [74, 75, 76, 77], not only because WM is generally required in real-world cases of relational

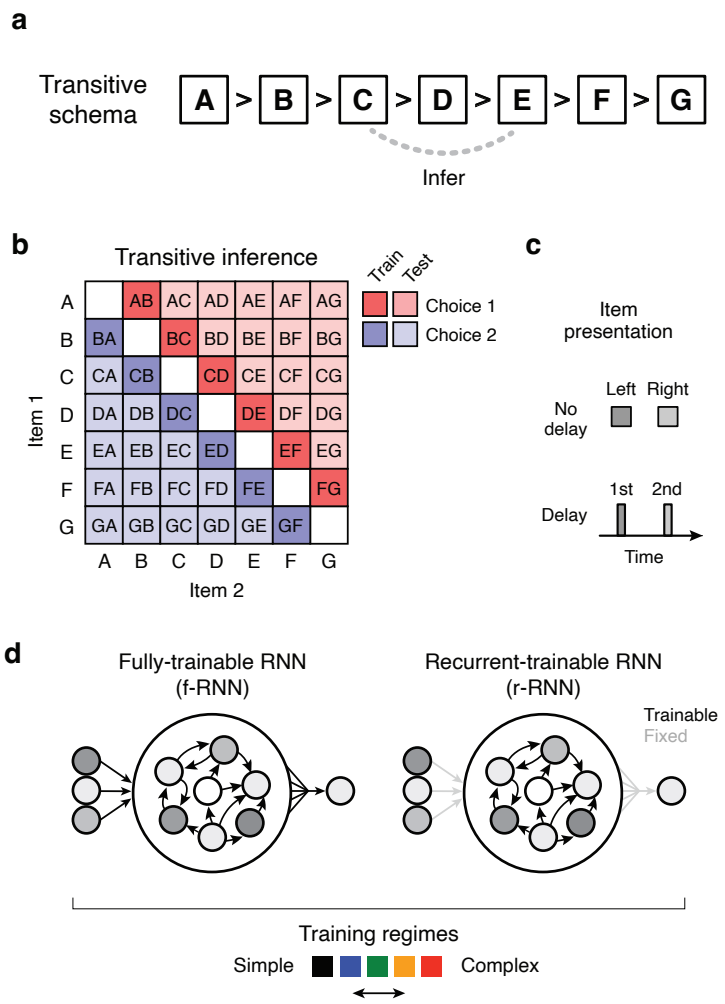

Figure 1: **Cognitive task and neural approach.** **a**, Diagram of the relational structure (schema) underlying transitive inference (TI). Subjects learn correct responses to premise pairs (A vs. B, B vs. C, C. vs. D, etc.; training trials), and must infer correct responses in previously unobserved (novel) pairs (C vs. E, etc.; testing trials). Correct responses are specified by an abstract rule: for every possible pair, the "higher" item in the schema should be chosen, in accordance with the transitive relation. Importantly, the items in the schema ("A", "B", etc.) correspond to arbitrary real-world items. **b**, Trial types and their correct responses in seven-item TI. Each trial type consists of a permutation of two items (item 1 and item 2). **c**, Item presentation task formats: no-delay vs. delay. In the no-delay format, task items are presented simultaneously and are chosen on the basis of their presentation position (e.g. left vs. right). In the delay format, task items are presented with an intervening delay and are chosen on the basis of their presentation order (1st vs. 2nd). The delay version explicitly requires working memory (WM), a cognitive capacity thought to be essential for relational inferences (such as TI) in living subjects. **d**, Neural models in the present study. Among neural network (NN) architectures, recurrent neural networks (RNNs) are able to implement WM. Two variants of RNNs were studied: fully-trainable RNNs (f-RNNs), for which both feedforward and recurrent synaptic weights were modifiable in training, and recurrent-trainable RNNs (r-RNNs), for which only recurrent synaptic weights were modifiable in training (feedforward weights chosen randomly and fixed) (trainable vs. fixed weights diagrammed as black vs. grey arrows, respectively). In conjunction, RNNs were trained with different levels of regularization and initial synaptic strengths ('training regime', indicated in later figures by color; parameters in **Table 1**). All networks had 100 recurrent units, which all had the same non-linear activation function (tanh). See Methods for additional details.

inference (e.g. language comprehension, spatial navigation), but also because prior work suggests that relational inference is accomplished in the brain by a neural system that intrinsically supports and/or relies upon WM (e.g. prefrontal cortex, possibly via a process akin to deliberation or reasoning [70, 72, 78]). Surprisingly, though TI exemplifies relational inference, prior work has only investigated indirect relationships between TI and WM (either by having separate WM vs. TI tasks [79], or by linking each to a common brain region [61, 80]), with no TI study to our knowledge directly testing WM by imposing an intervening delay between presented items whose relationship is to be inferred (e.g. item 1 - delay - item 2; schematic in **Fig. 1c** and **Fig. S1**). Notably, various behavioral studies of TI involve items that are sufficiently separated in space so as to require sequential observations (e.g. [52, 67, 68]), thus implicitly requiring WM.

Third, an approach should explicitly model **dynamics**. Even beyond the fact that WM necessitates dynamics over the course of a task trial, collective dynamical processes in the brain govern spontaneous neural activity (prominent in awake subjects; [81, 82, 83]) and likely also behavior in tasks, particularly tasks that expressly allow for self-determined responses. Such self-determined responses – and therefore the collective dynamics that underlie them – are relevant when tasks do not rigidly cue responses (e.g. via a 'go' cue), such as timing [84, 85] and reaction-time tasks [86, 87, 88], and are more broadly relevant in naturalistic or real-world settings [89].

Importantly, all three of these criteria are suited for the methodology of generating and analyzing task-optimized **recurrent neural networks** (**RNNs**), an approach that has proved successful in discovering neurobiologically insightful implementations of other cognitive abilities, and, further, in generating testable predictions at the level of collective neural activity [90, 91, 92]. We therefore adopted this approach, and, further, expanded upon it in two ways. First, in conjunction with RNNs, we also assessed whether and how trained models that cannot implement WM, yet have neurally relevant feedforward structure, might transitively generalize. To do so, we investigated two archetypal models: logistic regression (LR [93]) and multi-layer perceptron (MLP [94]) (schematic in **Fig. S2a**; see Methods), each tested on a task format having no delay (**Fig. 1c**). Second, to identify multiple potential solutions to TI, we investigated two neurobiologically relevant types of RNN variants:

Learnable connectivity. In NNs and the brain, network architecture is comprised of two types of connectivity: feedforward and recurrent. This distinction is important since the operations implemented by each type are not simply interchangeable: recurrent connectivity entails dynamics of the network's internal state (also interpretable as the network's working memory), which has no equivalent in networks that are solely feedforward. With regard to learning and performing tasks, the role of changes in feedforward vs. recurrent connectivity remains generally unclear (e.g. [95, 96, 97]). This is indeed the case for TI (a paradigm that requires learning of premise trials, e.g. A vs. B) — that is, it is not known whether the neural system implementing TI in the brain relies on learned connectivity that is feedforward, recurrent, or some configuration of both. It is worth emphasizing that this distinction is in fact critical for relational inference tasks such as TI, where the significance of the stimuli in the task (items A, B, C, etc.) has no *a priori* relationship to stimulus features, unlike tasks based on particular stimulus modalities and features (e.g. tactile frequency [98, 99], visual frequency and orientation [100], object categories [101]) that are known to be encoded in feedforward inputs from upstream brain regions. To address this gap in understanding, we investigated RNNs for which feedforward connectivity was either trainable or not trainable (fully-trainable RNN (f-RNN) vs. recurrent-trainable RNN (r-RNN), respectively; diagrammed in **Fig. 1d**).

	h_0	g_0	α	β
simple (high)	1.0	0.5	1.0	1.0
simple (low)	1.0	0.5	0.01	0.01
intermediate	1.0	1.0	0.01	0.01
complex (low)	1.0	2.0	0	0
complex (high)	1.5	4.0	0	0

Table 1: **Training regime parameters.** h_0, input gain; g_0, recurrent gain; α, weight regularization; β, metabolic regularization.

Training regime. The accuracy of trained NNs in matching neural systems in the brain has been found to depend on including biologically relevant constraints on the training procedure, such as penalties (regularizations) that serve to limit excess neural activity [48, 102, 103, 104]. Further, differences in the initial strength of connectivity – the magnitude of connection weights prior to training [105, 106] – have been found to yield NNs having striking differences in implementation (internal representations and dynamics) that are concomitant to differences in neurobiological accuracy [48, 103, 104, 107, 108]. These findings indicate that it is critically important to investigate both factors in trained NNs — training constraints and initial strength of connectivity — when seeking to identify biologically accurate neural implementations. We thus trained NN variants using systematically varied parameter sets corresponding to these factors, referring to each set as a "training regime" ('simple' to 'complex', **Table 1**).

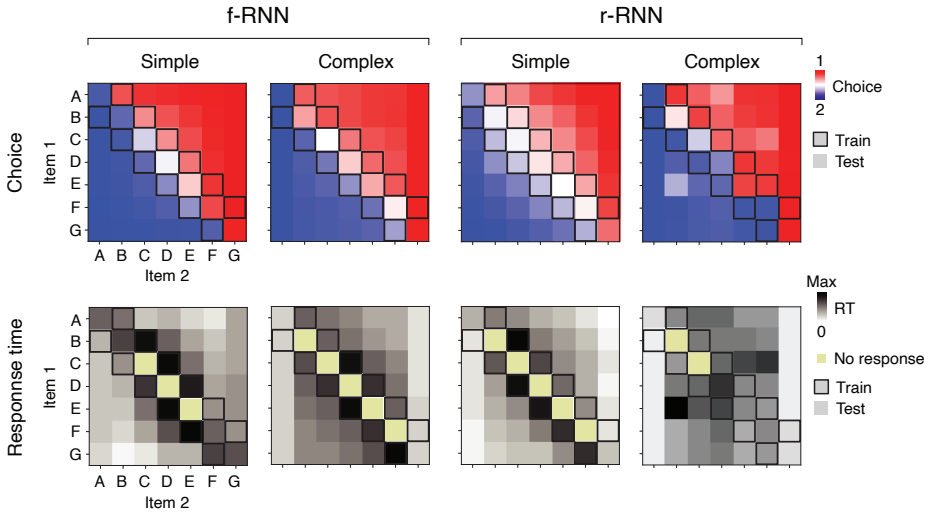

Figure 2: **A collection of recurrent neural networks (RNNs) perform transitive inference (TI).** Four example RNNs that perform delay-format TI (see **Fig. S1**). Each example RNN is plotted in a column. At left are fully-trainable RNNs (f-RNNs); at right are recurrent-trainable RNNs (r-RNNs); for each, one RNN from each of two training regimes (simple and complex; in particular, simple (high) and complex (high) in **Table 1**) are plotted. Top row, network output by trial type. Plotted for each trial type (in squares, defined by items 1 and 2; e.g. AB, BA, AC) is the value of output unit 1 minus that of output unit 2 (Choice, red and blue shades, corresponding to choice 1 and 2, respectively; see **Fig. S1**). Note RNNs were optimized (i.e. trained) only on training trial types (boxed squares). Bottom row, response time (RT) by trial type. In each trial, the choice response of the RNN was defined by the identity of the output unit (linear readout; one for each of two choices) that was first to reach a fixed activity threshold (85%); RT was defined as the time taken for the output unit to reach the threshold in the period following presentation of both items (choice period, see **Fig. S1a**). The RTs were normalized to the maximum duration (max) of the time period allowed for response (duration of the choice period). See Methods for additional details.

4 A variety of neural models perform TI

We first sought to determine whether trained RNNs could successfully perform TI. Unlike perceptual tasks or tasks based on statistical inference, TI expressly requires generalization to novel recombinations of inputs. Such generalization requires additional knowledge regarding the underlying relationship between inputs (the transitive schema, **Fig. 1a**). Thus TI is a task that requires an *a priori* inductive bias – here, for transitivity – for which it is not generally known whether relatively unconstrained models, such as trained RNNs, manifest [92, 109, 110].

	f-RNN	r-RNN
simple (high)	100	100
simple (low)	100	93
intermediate	98	82
complex (low)	95	57
complex (high)	79	41

Table 2: Number of RNNs (out of 100 trained instances) that generalized fully. All trained instances, regardless of generalization performance, responded correctly on all training trials.

Mirroring TI as encountered by living subjects (7-item TI with items A to G; each item represented as a random high-dimensional (100-D) input; see Methods), we trained RNNs (100 recurrent units, tanh nonlinearity) exclusively on "premise" (training) trials (A vs. B, B vs. C, C vs. D, etc.) and evaluated whether RNNs generalized to test trials (B vs. D, etc.; all trial types shown in **Fig. 1b**). Training was conducted using gradient descent optimization and backpropagation-through-time; furthermore, for RNNs, all trials required working memory (WM) (in delay format, **Fig. 1c** and **Fig. S1**). Response choice was defined by which of two output units (linear readouts; corresponding to choice 1 vs. 2) was activated in the trial, and response time (RT) was defined as the time at which activity of the output unit met a fixed threshold (85% of maximum activation). For an initial assessment of whether RNNs could generalize, a simulation of all trial types was performed under noiseless conditions.

We found that trained RNNs often generalized perfectly, i.e. responded correctly to all test trials (example RNNs in **Fig. 2**; summary in **Table 2**). Interestingly, it was also common for trained RNNs to respond correctly on all premise pairs (training trials) yet not generalize (example RNNs in **Fig. S3**; **Table 2**), whereas feedforward models trained on a no-delay task format never failed to generalize (examples in **Fig. S2**; LR: 100 out of 100 instances; MLP: 100 out of 100 instances; see also [111] for additional results showing generalization in MLPs), highlighting an essential difference posed by the requirement for WM (see additional discussion in Methods, Task).

Notably, we also observed that, among RNNs, fully-trainable RNNs (f-RNN) and simple training regimes more frequently yielded RNNs that generalized (**Table 2**), providing an initial hint that RNN variants might have functionally important differences.

5 Trained RNNs show a collection of emergent behaviors

Decades of work have established that living subjects performing TI show striking and commonly shared patterns of behavior, manifesting both as patterns of errors and response times (RT) [53, 60]. These behavioral patterns are based on trial types, where each trial type is defined by items (e.g. AB, BA), and where each item is defined by its implicit position — or "rank" — in the transitive schema (**Fig. 1a**; the rank of A is highest, while G is lowest). As recognized previously [60], empirically observed behaviors are not only important constraints on explanatory accounts of TI, but also potential sources of insight into underlying implementations.

We therefore investigated whether these behaviors (described below), and possibly others, were expressed by RNNs that successfully perform TI (**Fig. 2**). Importantly, expression of these behaviors would effectively be emergent, since the RNNs were neither pre-configured nor trained to express particular behaviors beyond that of responding correctly on training trials. Further, prior work on TI has either omitted or left unanalyzed delay periods that intervene between item presentation – necessitating WM – leaving unknown whether and how previously described TI behaviors relate to WM. Indeed, imposing a delay requiring WM also raises the possibility of observing, either in models or subjects, behavioral patterns that have not been previously studied and that could reflect underlying implementations.

Given these considerations, we took the following approach to determine whether RNNs expressed either known or novel behavioral patterns. First, we focused on RNNs that generalized fully (correct responses for every test trial type). Next, for each individual RNN, we simulated the networks (5000 runs, each of all trial types) with progressively increasing levels of intrinsic noise until the average performance of the RNN was >50% on training trials and <95% on testing trials (i.e. sub-asymptotic performance, the level of performance for which behavioral patterns have been observed; see Methods for further details). Lastly, we ran simulations (5000 runs, all trial types) at this noise level, from which we then measured performance (% correct) and RTs. RTs were measured using a previously established criterion (time to a fixed threshold in output units [112]).

We found that RNNs exhibited not only previously observed TI behaviors, but also a novel behavioral pattern not previously studied (**Fig. 3**; an analogous analysis of behavior in feedforward models is shown in **Fig. S2**). Importantly, behavior was both qualitatively and quantitatively comparable to that of living subjects, both single RNNs and across RNNs (**Fig. S4**). In the following sections, we address each behavioral pattern.

The symbolic distance effect. A standard observation across behavioral studies of TI is the "symbolic distance" effect: the larger the difference in rank between items (e.g. A vs. D compared to A vs. B), the higher the accuracy and the lower the RT [51, 53, 59, 60]. We found that RNNs performing TI invariably exhibited the symbolic distance effect (**Fig. 3**, second column; see also **Fig. S4** for effect across all trial types).

The end item effect. Along with the symbolic distance effect, a standard behavioral observation in subjects is the "end item" (or "terminal item") effect: trials containing either the highest- or lowest-rank item (A and G, respectively; both "end items") are associated with higher performance and lower RTs [53, 60, 113]. We found that RNNs performing TI invariably exhibited the end item effect (**Fig. 3**, third column; see also **Fig. S4** for effect across all trial types).

The "end order" effect: a novel behavior with two alternative versions. In some behavioral studies of TI, subjects perform better in trials containing the highest rank item (item A) compared to trials containing the lowest rank item (item G), a pattern termed "lexical marking" ([111, 114]; related to a 'magnitude' effect [66]). Further, a previous study of MLPs that perform TI in the non-delay format (**Fig. 1c**) has suggested that MLPs characteristically show lexical marking [111]. In contrast to these results, we did not observe a consistent lexical marking effect, either in feedforward models (LR and MLPs, **Fig. S2**; see **Fig. S2** for further discussion) or RNNs (**Fig. 3**, fourth column). This discrepancy may be due to differences in the relative size of MLPs, as the number of hidden layer units is comparable to number of task items in [111], while in the present study the number of hidden layer units (100 units) is an order of magnitude larger than task items (7 items) (further details in Methods).

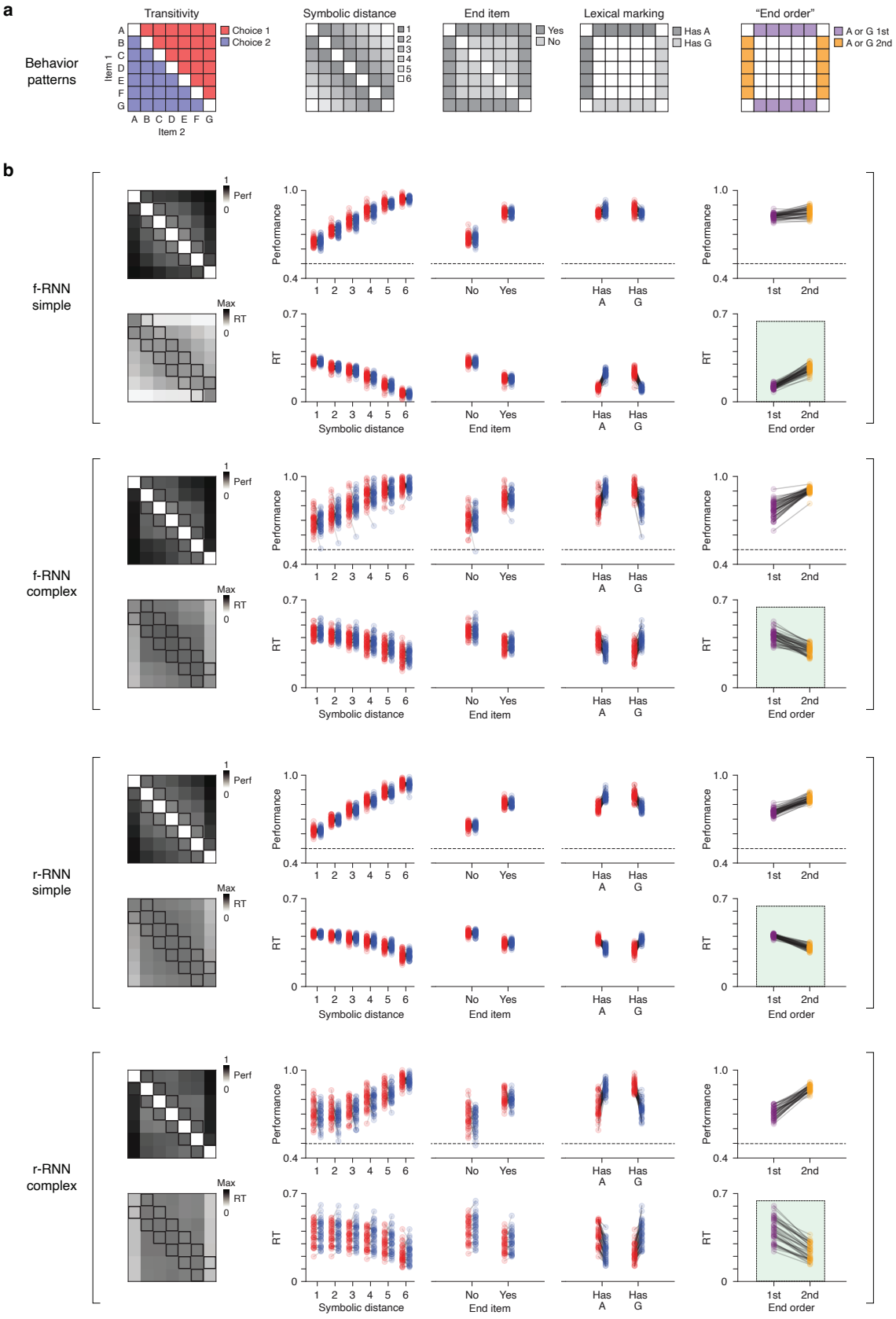

Figure 3: **RNNs performing TI show multiple emergent (untrained) behaviors.** See caption further below.

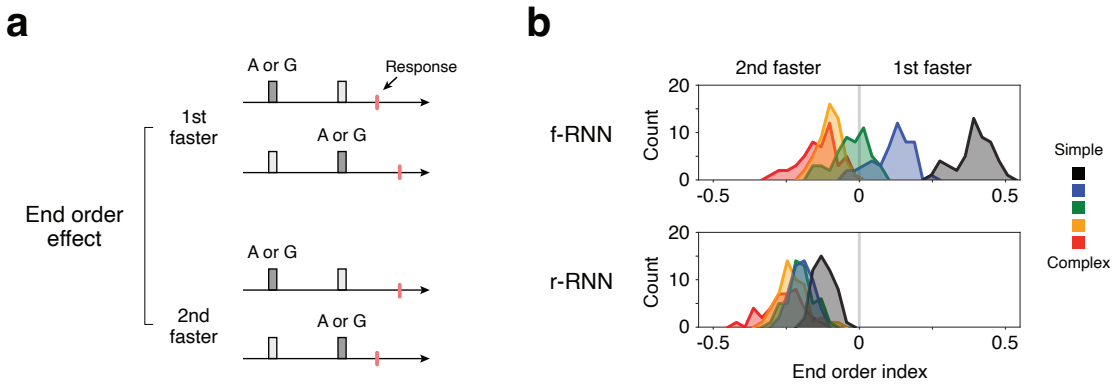

Figure 4: **RNN variants systematically show alternative response-time (RT) behavior. a**, Schematic of the end-order effect, a response-time (RT) behavior. The behavioral pattern is observed in RNNs in two alternative versions: 1st-faster vs. 2nd-faster. An additional schematic that shows all trial types used to quantify the effect is in **Fig. 3a** (last column). **b**, Histograms of end-order RT behavior across RNNs (counts are of RNN instances; colors correspond to different training-regime RNN variants). Plotted are counts of RNN instances. The behavior was quantified as the difference of average RTs divided by their sum (end order index; average RTs calculated for trials where end items (A and G) occurred 1st vs. 2nd). RNN variants (f-RNN vs. r-RNN and simple vs. complex training regimes) follow that diagrammed in **Fig. 1d**. These results summarize those highlighted in **Fig. 3** (fifth column, green zones).

While investigating lexical marking behavior, we made an unexpected observation: RNNs showed lower RTs if an end item (A or G) was presented first rather than second (**Fig. 3**, fourth column, f-RNN simple). We further unexpectedly found that other RNNs showed the inverse behavioral pattern, namely, lower RTs if an end item (A or G) was presented 2nd (rather than 1st) (**Fig. 3**, fourth column, f-RNN complex, and both r-RNN simple and complex). We termed the shared pattern the "end order" effect (i.e. end-item order; quantified separately in the sixth column of **Fig. 3**, fifth column), and its former and latter patterns as 'first-faster' and 'second-faster' versions, respectively.

Crucially, in examining this behavior across networks using a quantitative index (end order index, ranging from -1 (second-faster) to 1 (first-faster); see Methods) (**Fig. 4**), we found that the two alternative versions of the

Figure 3. **RNNs performing TI show multiple emergent (untrained) behaviors. a**, Schematic of behavior patterns. Each schematic shows the trial types (squares) that define the behavioral pattern. Transitivity: the correct choice (item 1 or 2) for transitive inference (i.e. the item 'higher' on a transitive schema, **Fig. 1a**). Symbolic distance: the size of the difference in rank between item 1 and 2; rank is an item's discrete position in the transitive schema (A: 1, B: 2, C: 3, etc.). End item: whether the trial type contains an 'end item', the item at either end of the transitive schema (here, A or G). Lexical marking: whether the trial type has the 'highest' or 'lowest' end item (here A vs. G, respectively). End order: whether the end item (A or G) is the 1st item (item 1) to be presented, or the 2nd item (item 2); defined only for the TI task format with an explicit WM delay (**Fig. S1**). **b**, Behavioral results across four RNN variants (n = 40 or 50 instances / variant). The four RNN variants were defined by different learnable connectivity (f-RNN vs. r-RNN) and training regime (simple vs. complex, corresponding here to simple (high) and complex (high), respectively, see **Table 1**). Results from each RNN variant are presented in a block of two rows (top row: performance (proportion correct); bottom row: response time (RT) (proportion of maximum time allowed for response)). Column 1: Averages across RNN instances (n = 40 or 50 instances) by trial type. Columns 2-5: Averages across trials by trial type, for all RNN instances (5000 simulated trials / trial type; each point corresponds an RNN instance, n = 40 or 50 instances). Trial types follow those in panel a (column 2: symbolic distance; column 3: end item; column 4: lexical marking, column 5: "end order"), in addition to distinguishing between choice 1 vs. choice 2 trial types (red vs. blue, respectively; diagrammed in panel a, transitivity). Highlighted (green zones) is an alternative behavior across variants (end order response times).

behavior systematically differed with respect to both types of RNN variants (learnable connectivity (f-RNN vs. r-RNN) and training regime (simple to complex), **Fig. 1d**). First, we found that r-RNNs always showed the second-faster pattern (**Fig. 4b**, bottom row), whereas f-RNNs showed both (**Fig. 4b**, top row). Second, we found that complex-regime RNNs nearly always showed the second-faster pattern (**Fig. 4b**, red histograms in top and bottom rows). In contrast, simple-regime f-RNNs consistently showed the first-faster pattern (**Fig. 4b**, both simple (high) and simple (low), black and blue histograms, respectively, in top row), with f-RNNs trained in the simple (high) regime always showing the first-faster pattern (**Fig. 4b,** black histograms in top row). Notably, intermediate-regime f-RNNs showed mixed and lower magnitude patterns, consistent with a systematic relationship between training regime and behavior.

More broadly, the finding that RNNs performing TI consistently show alternative patterns of behavior (first- vs. second-faster end order behavior; **Fig. 3**, fifth column) implied that the networks had systematically different underlying implementations. This implication was further substantiated by the finding that these differing behaviors systematically varied across networks (**Fig. 4**). Together, these findings provided additional motivation to investigate the underlying implementations of these networks.

6 A simple dynamic and geometry implementing transitive comparison

Neural recordings in living subjects performing TI indicate that single neurons in the brain can encode variables relevant to TI, including symbolic distance [62, 63]. It remains unclear what collective neural process or activity pattern implements the comparison operation – akin to a '>' operator – that generalizes transitivity to novel combinations of items (test cases).

For initial insight, we analyzed how solely feedforward models generalized transitivity (no-delay format in logistic regression and multi-layer perceptron models, **Fig. S5**). In these models, examination of unit activations indicated that transitive comparison was implemented using a "subtractive" solution, where the rank (A, B, C, etc.) and position (left vs. right) of each item was mapped to the magnitude and sign, respectively, of unit activation (**Fig. S5**). Notably, in either feedforward model (LR or MLP), this subtractive solution was sufficiently realized by the (trained) feedforward weights operating directly on the input [111], clarifying that a direct operation on otherwise arbitrary inputs (items A, B, C, etc. encoded as high-dimensional (100-D) random vectors in input activity space; see Methods) is adequate to perform TI when feedforward input can be learned or trained. This further raised the question of how TI is performed when modifiable feedforward connectivity is not sufficient (e.g. when memory over time is required, as in WM) and/or not available (e.g. a putative neural system in the brain underlying TI, and as the case in r-RNNs).

We thus sought to clarify neural implementations in RNNs solving TI. Given the above findings in feedforward models (**Fig. S5**), we broadly hypothesized that other neural systems having trainable feedforward connectivity — such as f-RNNs — might adopt a similar subtractive implementation. Beyond this hypothesis, we sought to address three unknowns. First, it was not clear whether or how the subtractive implementation observed in feedforward models – which relies on simultaneous input and thus requires twice the number of input connections as RNNs – was relevant to the RNNs solving TI with delay requiring WM. Second, it was also not clear whether f-RNNs would consistently adopt the same implementation, across either different instances (different random initializations of network weights) or different training regimes (simple vs. complex, **Table 1**). Lastly, it remained unclear whether the implementation of TI in r-RNNs, which have no trained feedforward connectivity, would show any similarity to feedforward implementations.

	f-RNN	r-RNN
simple (high)	98.1 ± 0.4	87.5 ± 1.7
simple (low)	97.1 ± 1.4	88.8 ± 2.2
intermediate	92.1 ± 1.6	85.6 ± 2.5
complex (low)	82.0 ± 3.5	78.8 ± 3.9
complex (high)	86.3 ± 3.0	91.4 ± 1.8

Table 3: PCA % variance explained by top 3 PCs across RNNs. Mean ± s.d. (n=50 instances; n=40 instances for complex (high) r-RNN). PCA was calculated for all activity state vectors for every time point across all trial types (12 training and 30 test, **Fig. 1b**).

To address these points, we began by first investigating the implementation of TI in fully-trainable RNNs (f-RNNs) trained in the simple regime (**Fig. 5**; RNN types and training regimes in **Fig. 1d**), as we thought these networks might show the most similarity to the feedforward models characterized previously (**Fig. S5**). We initially observed that population activity was consistently low-dimensional (% variance explained by top 3 PCs, **Table 3**). Next, in examining population trajectories (example network in **Fig. 5a**), we observed two signature patterns: (1) a **linearly-arranged rank-ordered** response to item presentation (**Fig. 5a**, observable in green-shaded circles, corresponding to the network response to item 1), and (2) a **rotational pattern** during the delay period (**Fig. 5a**,

bottom), suggesting an oscillatory dynamic.

Pattern 1 suggested a similarity to the subtractive implementation, as the pattern was characterized by an intrinsically 1-D ordering of item ranks (observed in single units in feedforward models, whereas in N-D population activity space in RNNs) and was due to trained feedforward connectivity. Pattern 2 suggested that a low-dimensional dynamical process — specifically a single (2-D) oscillation — might be sufficient to account for how these networks implemented transitive comparison across time. Importantly, RNN activity during the delay period was effectively described by linear dynamics (ordinary least-squares fit) characterized by an oscillatory mode correspondent with pattern 2 (**Fig. 5b-d**, $R^2 \sim$ 0.8-0.9, see **Fig. S7**). Given these clues, we next sought to identify the underlying implementation explicitly. Importantly, prior work has found that, in trained RNNs, analysis of network-level neural dynamics with respect to fixed points (FPs) can identify dynamical components that have specific functional roles [92, 115, 116, 117] and are jointly sufficient to perform cognitive tasks.

Taking this approach, we found that simple-regime f-RNNs had an "early-trial" fixed point located near activity trajectories at the beginning of trials (**Fig. 5a**, orange cross). This fixed point had an oscillatory mode of frequency ∼0.5 cycles/delay (highlighted by open arrowheads in **Fig. 5b**) and was orthogonal to the mean trajectory across trial types (i.e. the cross-condition mean (XCM), a population activity component observed across trial-based tasks [118, 119, 120]; plotted as yellow line in **Fig. 5a**; quantification of orthogonality in **Fig. S8**). Some RNN instances also showed additional FPs that were overtly associated with other task functions (**Fig. 5a**, black crosses): two "choice" FPs that are stable FPs (attractors) toward which trajectories corresponding to each of the two choices travelled and would eventually settle after item 2 presentation (not shown), and a saddle FP, located between the two choice FPs, that was stable except for a single unstable axis oriented toward each choice FP (**Fig. 5a**, choice FPs: black crosses respectively near end of choice 1 and choice 2 trajectories; saddle FP: center black cross, saddle unstable axis in pink line). This unstable axis is effectively a "choice axis," as it was aligned with a subsequent alignment of activity trajectories with either one of the two choice FPs, following presentation of item 2 (**Fig. 5a**, item 2 presented at the time point corresponding to the colored stars), and was parallel to the direction between choice 1 and choice 2 activity states. More generally, these latter patterns of activity in these networks indicate an effective choice axis (direction pointing from choice 1 vs. choice 2 activity states; see Methods) that is parallel to the oscillatory mode (quantification in **Fig. S8)**.

Together, these dynamical components indicated that the transitive comparison could be performed within the 2D linear subspace (plane) of the oscillatory mode of the early-trial FP. To test this possibility, we evaluated whether TI could be performed solely by activity and dynamics within the identified oscillation (constituting a linear approximation of the oscillation seen in the RNN). We found that this oscillation (see Methods), when presented with trial-structured input (item 1 - delay - item 2), yielded activity trajectories for which correct choice was linearly separable (**Fig. 5d**).

This clarified a simple implementation of transitive comparison over time (**Fig. 5e**): namely, a single oscillation that uniformly re-orients activity states corresponding to item 1 (A, B, C, etc.), doing so via a common angular displacement. In the delay period (following the presentation of item 1), this re-orientation serves to shift the activity state in the direction opposite to the direction of the activity displacement due to item presentation. Note that the activity displacement due to item presentation is rectilinear – a consequence of its implementation as feedforward input – whereas the dynamics-driven activity displacement in the delay period is angular. The re-oriented activity state is thereby "subtractive" with respect to the activity shift elicited by the subsequent item presentation (item 2).

Notably, a particular angular shift of ∼0.5 cycles (resulting from an oscillatory frequency of ∼0.5 cycles/delay) re-orients the activity state to be opposite (diametric) to that imposed by the presentation of the item prior to the delay. This can be likened to a change-of-sign along a "comparison" axis in population activity space (**Fig. 5e**), and is analogous to the mapping of signs (- vs. +) to item position (left vs. right) in the subtractive implementation of TI in feedforward models (**Fig. S5**). The resulting activity along the comparison axis accounts for transitive comparison and the symbolic distance effect, as long as the output implemented by the network (the readout mechanism, e.g. "choice axis" of saddle-point dynamics, **Fig. 5a**) is aligned with the comparison axis.

7 Geometric signatures of alternative neural implementations

The above findings identify a neural implementation of TI (**Fig. 5**), but do not address other putative neural implementations. As outlined further above ('A neural approach to transitive inference'), prior studies have found that RNN variants performing a task may show strikingly different levels and signatures of neurobiological accuracy [48, 103, 121, 97], implying that variants adopt different underlying neural implementations. Likewise, we previously observed that RNN variants performing TI systematically showed alternative behavioral patterns (**Fig. 4**), similarly

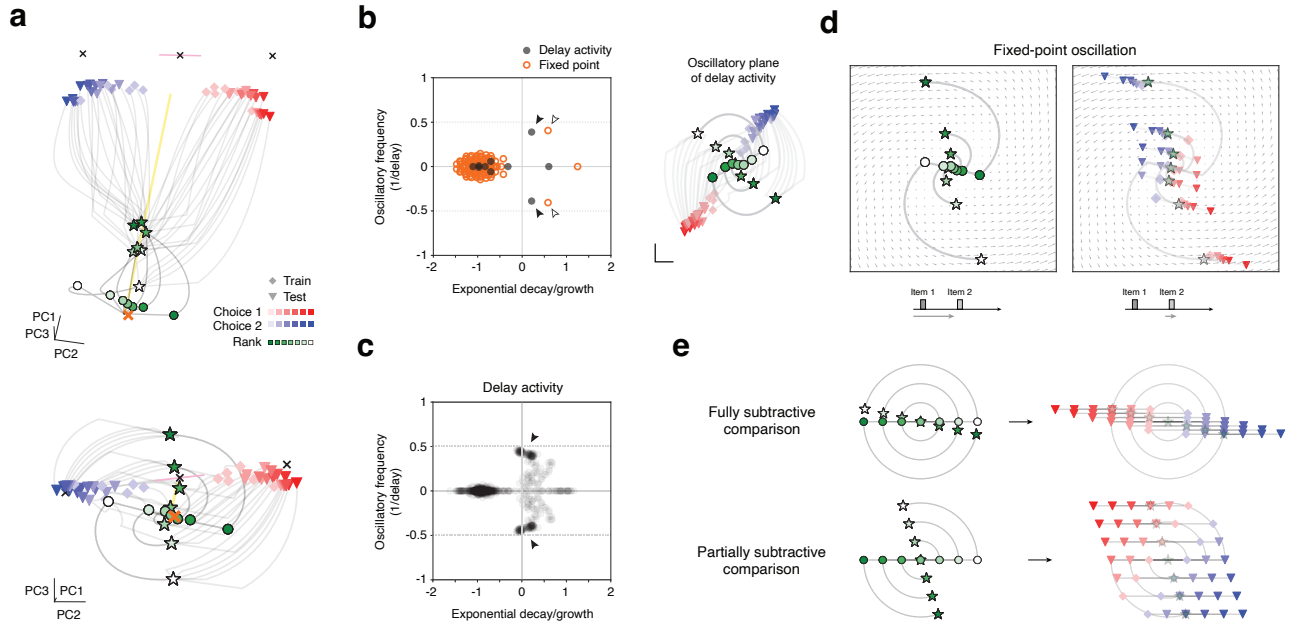

Figure 5: **A 'subtractive' neural implementation of TI in simple-regime f-RNNs.** **a**, Population activity trajectories in an RNN (simple-regime f-RNN) that performs TI. Top and bottom plots show two different views. Shown are trajectories from all 42 trial types (Fig. 1b). To clarify the operation of the network, three trial times are highlighted: (i) presentation of item 1 (green circles; shade indicating item rank: A (dark green) to G (white)), (ii) the last time point of the delay period (green stars; same color convention), (iii) last time point of the trial (red/blue symbols; red: choice 1 trials, blue: choice 2 trials, light to dark shading indicating symbolic distance (1 to 6); diamonds: training trials, triangles: test trials). Also shown: cross-condition mean (XCM; the average trajectory across all trial types) (yellow line) and fixed points (FPs) (crosses). Two FPs were attractors (lone black crosses), one FP was a saddle point (black cross with pink line indicating axis of unstable mode), and one FP (orange cross) was located near trajectories during the delay period ('early-trial' FP). Note two prominent activity patterns: linearly arranged rank-ordered activity upon presentation of item 1 (green circles) and the oscillatory evolution of trajectories in the delay period (circles to stars). **b**, Linear dynamics of RNN in panel a. Left, eigenvalue spectra of the RNN. The spectra were calculated in two ways: first, from delay-period neural activity (black points; inferred via least-squares linear fit, $R^2 = 0.65$) and second, from linearization of the network with respect to the early-trial FP (orange circles; FP shown as orange cross in panel a). Right, population activity trajectories of the RNN plotted in the plane of the oscillation inferred from delay period activity (filled arrowheads in spectra plot). **c**, Linear dynamics of simple-regime f-RNNs (n=100 instances; 50 instances of simple (high), 50 instances of simple (low)). Eigenvalue spectra of delay-period neural activity (grey translucent points; inferred via least-squares linear fit, $R^2 = 0.77 \pm 0.11$ (mean $\pm$ s.d.) across 100 instances; see also Fig. S7). Note the density of oscillatory modes with frequency $\sim$0.5 cycles / delay (filled arrowhead). **d**, Activity trajectories in the oscillatory mode of the linearized RNN. The oscillatory mode is that of the linearization of the early-trial FP (open arrowheads in panel b). Plotted in background are flow field vectors (not to scale; shown for motion direction). To clarify how the activity evolves, trajectories are plotted in two stages of the task trial (left and right panels; schematic at bottom of each): early trial (left) and presentation of item 2 (right). Three trial times are highlighted: (i) presentation of item 1 (green circles; shade indicating item rank: A (dark green) to G (white)), (ii) the last time point of the delay period (green stars; same color convention), (iii) presentation of item 2 (red/blue symbols; red: choice 1 trials, blue: choice 2 trials; diamonds: training trials, triangles: test trials). Note the separation of choice 1 vs. 2 trials (red vs. blue symbols), indicating that transitive responses were evolved in the activity space. **e**, Schematic of solution to TI expressed in RNNs: population-level "subtraction". Plotted are activity trajectories generated by simulating a 2D linear dynamical system defined by an oscillation of frequency $\sim$0.5 (top row) vs. $<$0.5 (bottom row) cycles / delay, with initial condition at the origin and input vectors encoding task items (A, B, C, etc.) in ordered collinear arrangement in state space (compare to panels a and d). Trial-based input (item 1 - delay - item 2, see Fig. S1a) was applied to the system. Plotting conventions are the same as that of panel d. Note that in either case, the systems evolve separation (linear separability) between choice 1 (red symbols) vs. 2 (blue symbols) trials.

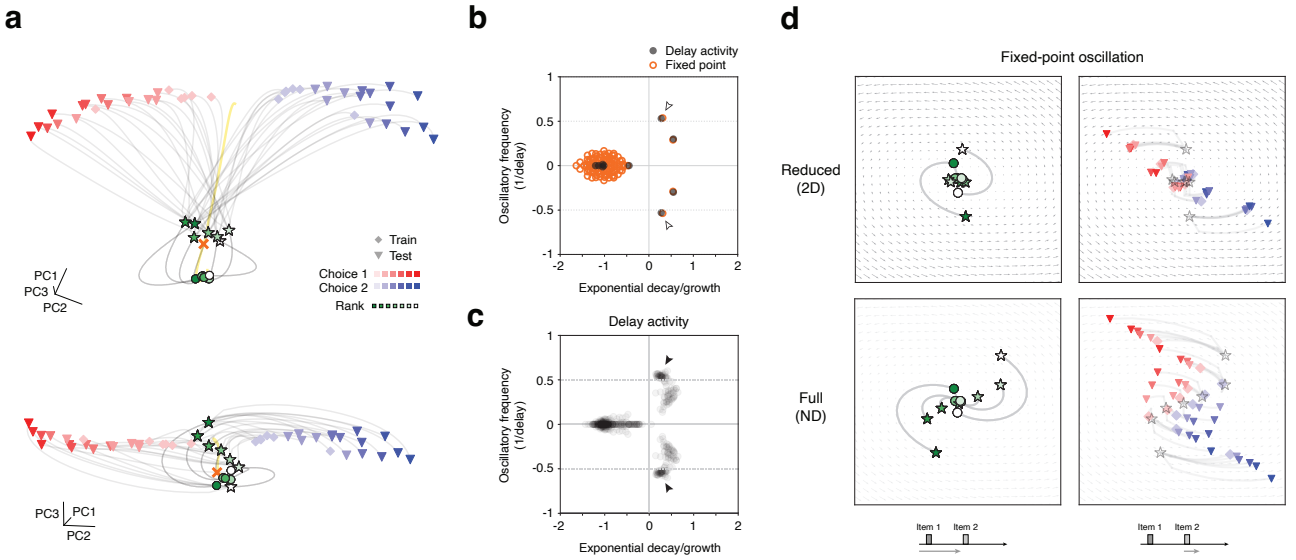

Figure 6: **Low-dimensional oscillatory dynamics in simple-regime r-RNNs. a**, Population activity trajectories in an RNN (simple-regime r-RNN) that performs TI. Top and bottom plots show two different views. Shown are trajectories from all 42 trial types (Fig. 1b). To clarify the operation of the network, three trial times are highlighted as follows: (i) presentation of item 1 (green circles; shade indicating item rank: A (dark green) to G (white)), (ii) the last time point of the delay period (green stars; same color convention), (iii) last time point of the trial (red/blue symbols; red: choice 1 trials, blue: choice 2 trials, light to dark shading indicating symbolic distance (1 to 6); diamonds: training trials, triangles: test trials). Also shown: cross-condition mean (XCM; the average trajectory across all trial types) (yellow line) and fixed point (FP) (orange cross). The FP was located near trajectories during the delay period ('early-trial' FP, compare to Fig. 6a). Note the oscillatory evolution of trajectories in the delay period (circles to stars) despite the absence of a linearly arranged rank-ordered activity upon presentation of item 1 (green circles; compare to Fig. 6a). **b**, Linear dynamics of RNN in panel a. Two eigenvalue spectra of the RNN are plotted: first, the spectrum calculated from delay-period neural activity (black points; inferred via least-squares linear fit, $R^2 = 0.78$) and second, the spectrum from linearization of the network with respect to the early-trial FP (orange circles; FP shown as orange cross in panel a). **c**, Linear dynamics of simple-regime f-RNNs (n=50 instances, simple (high) training regime, Table 1). Eigenvalue spectra of delay-period neural activity (grey translucent points; inferred via least-squares linear fit, $R^2 = 0.81 \pm 0.02$ (mean $\pm$ s.d.) across the 50 instances, see also Fig. S7). Note the density of oscillatory modes with frequency ~0.5 cycles / delay (filled arrowhead; compare to Fig. 5c). **d**, Activity trajectories in the oscillatory mode of the linearized RNN. The oscillatory mode is that of the linearization of the early-trial FP (open arrowheads in panel b). Two sets of trajectories are plotted (rows): one where all activity (initial conditions and inputs) has been reduced to the 2 dimensions of the oscillatory mode (top row, reduced) and one where the full dimensionality (N-D; here N=100) of initial conditions and inputs have been retained (bottom row, full). In both cases, the dynamics are governed solely by the oscillatory mode (see Methods), though only in the reduced (2D) case do trajectories strictly follow the flow field vectors. To clarify the progression of activity, trajectories are plotted in two stages of the task trial (left and right columns; schematic of the stages at the bottom of each panel): early trial (left) and late trial (right). To clarify how the activity evolves, specific trial times are highlighted as follows: (i) presentation of item 1 (green circles; shade corresponding to item rank: A (dark green) to G (white)), (ii) the last time point of the delay period (green stars; same color convention), (iii) a quarter of the time period following presentation of item 2 (i.e. choice period, see Fig. S1; red/blue symbols; red: choice 1 trials, blue: choice 2 trials; diamonds: training trials, triangles: test trials). Note that separation (linear separability) of choice 1 vs. 2 trials (red vs. blue symbols) does not occur in the reduced-activity case, but does occur in the full-activity case.

implying alternative neural implementations.

To clarify these alternative implementations – in particular, the population-level patterns of neural activity they respectively predict – we took the 'subtractive' implementation shown by simple-regime f-RNNs as a point of reference (**Fig. 5**). We sought to identify an activity pattern that characterizes this implementation while also being easily quantifiable (whether in neural models or experimental data). Such a characteristic or signature activity pattern could then be a means by which to survey or discover alternative implementations. From the results above (**Fig. 5**), two candidate activity patterns were (1) the linearly-arranged rank-ordered arrangement of activity states (pattern 1), which here refer to as "ordered collinearity," and (2) the oscillation associated with transitive comparison (pattern 2). Across RNN variants, we assessed these two patterns in turn.

Surprisingly, we found that pattern 1 was not unique to the subtractive implementation observed in simple-regime f-RNNs. A striking indication that this was the case came from analyzing r-RNNs (recurrent-trainable RNNs), an RNN variant that necessarily adopts an alternative overall implementation, since the subtractive implementation (**Fig. 5e**) depends in part on the ordered collinear activity made possible by learned feedforward connectivity (not available in r-RNNs, see **Methods**). Despite this fundamental difference, neural activity in r-RNNs appears remarkably similar to that of simple-regime f-RNNs (**Fig. 6a**, compare to **Fig. 5a**): in particular, r-RNNs showed oscillatory activity reminiscent of the single oscillation associated with transitive comparison previously seen in simple-regime f-RNNs (frequency ∼0.5 cycles/delay, inferred either from fixed-point linearization or from linear dynamics inferred from delay period activity, **Fig. 6b-c**). As with f-RNNs previously (**Fig. 5d**), linearization analysis of these networks revealed that a single linear oscillatory mode was sufficient to accomplish transitive comparison in time (**Fig. 6d**). Interestingly, this sufficiency was only apparent when the oscillation was applied to full-dimensional activity (**Fig. 6d**, bottom row; see Methods for further explanation) and not activity restricted to 2-D subspace of the oscillation (**Fig. 6d**, top row), an indication that the implementation of TI in these networks also depends on higher-dimensional activity components. These results indicate that though r-RNNs cannot adopt the subtractive solution (**Fig. 5e**), the implementation in r-RNNs nonetheless can rely on a single dynamical pattern (here an oscillation) to accomplish transitive comparison over time.

Unlike pattern 1, pattern 2 (ordered collinearity) differed substantially across alternative implementations of TI. To assess pattern 2 – a simple geometric pattern in population activity space (seen in particular activity subspaces, e.g. PCs and oscillatory subspaces, **Fig. 5a, c**) – we first defined a measure that quantifies the degree to which neural activity conforms to this pattern, which we termed the ordered collinearity index (OCI) (ranging from -1 to +1, schematic in **Fig. 7a**; see Methods). The OCI is measured in the full activity space of individual networks (ambient N-D space reduced to top 10 PCs, see Methods). As expected from previous results (**Figs. 5** and **6**), simple-regime f-RNNs expressed OCIs that were relatively high (>0.5) or near 1 (**Fig. 7a**, black and blue histograms in left and middle columns; OCI, early delay: simple (high): 0.78 ± 0.04, simple (low): 0.73 ± 0.04; OCI, late delay: simple (high): 0.95 ± 0.02, simple (low): 0.90 ± 0.04; mean ± s.d., n=50 for each variant) and that were virtually always higher than that of r-RNNs in the early part of the delay period (**Fig. 5a**, first column; r-RNN: 0.22 ± 0.11, n=240 instances, simple-regime f-RNN: 0.76 ± 0.05, n = 100 instances; mean ± s.d.) (note that early vs. late delay were defined as the first vs. last quarter of the delay period, respectively). These results establish that the OCI – a measure that is evaluable for any neural system performing TI over a delay period – can be used to distinguish between alternative implementations.

Measuring the OCI yielded further insight into activity geometry across implementations. In examining OCIs at different time points during the delay (**Fig. 7a**, left and middle columns), we noticed that OCI could increase dramatically across the delay period (e.g. ∼0.1 to ∼1 for r-RNNs trained in the simple (high) regime, **Fig. 7a**, black histograms in bottom row; also previously observable as an "unfolding" of activity trajectories during the delay period in **Fig. 6a**; see **Fig. S6** for example time traces of OCI). This observation led us to examine change in OCI across RNN variants (**Fig. 7a,** right column). Intriguingly, f-RNNs exhibited systematic differences depending on training regime: in particular, simple- and intermediate- regime networks invariably increase OCI during the delay (**Fig. 7a**, black, blue, and green histograms in top row; OCI change: simple (high): 0.17 ± 0.03, simple (low): 0.17 ± 0.05, intermediate: 0.08 ± 0.06; mean ± s.d., n=50 instances for each variant), whereas complex-regime networks consistently decrease OCI during the delay (**Fig. 7a**, both complex (high) and complex (low), red and orange histograms, respectively, in top row; OCI change: complex (high): -0.13 ± 0.10, complex (low): -0.09 ± 0.07; mean ± s.d., n=50 instances for each variant). These qualitatively different patterns of activity (increasing vs. decreasing OCI) were consistent with alternative underlying implementations in these network variants, as implied previously by the alternative patterns of behavior expressed by these variants (**Fig. 4**). More broadly, these findings raised the possibility that, besides ordered collinearity (which characterizes the subtractive implementation of TI, **Fig. 5f**), there might exist other geometric arrangements of activity, expressed during the delay period, that distinguish between alternative implementations of TI and thus constitute additional neural predictions.

Leveraging this insight, we identified one such predictive geometric activity pattern. We were struck by the fact that alternative implementations of TI – namely, the subtractive solution (**Fig. 5f**) vs. the solution in simple-regime r-RNNs (**Fig. 6**; lacking ordered collinearity from feedforward input) – nonetheless both appeared to depend upon a similar dynamical pattern: an single oscillation enabling transitive comparison over time. We conjectured that a single oscillation could be a relatively simple case in a putative wider set of dynamical patterns enabling transitive comparison over time, and, further, that the essential operation of this wider set of dynamics is rotation.

Given this possibility, we hypothesized that activity geometry based on angular rather than distance relationships (i.e. angles (vs. distances) between activity states in A vs. B trials, A vs. C, B vs. C, etc.) might effectively distinguish between alternative implementations of TI. To test this possibility, we evaluated two geometric measures – mean angle (**Fig. 7b**) and mean distance (**Fig. 7**c) – that were similar to the OCI in that they quantify geometric relationships of neural activity across trial types (see Methods), but different in that they do not measure conformity specifically to ordered collinearity. Strikingly, we found that change in mean angle (**Fig. 7b**, left column), but not mean distance (**Fig. 7b**, right column), categorized RNN variants (values <0 vs. >0) in overall accordance with the two types of alternative implementations previously implied: (1) the subtractive implementation exhibited in simple-regime f-RNNs vs. the implementation in r-RNNs (which do and do not rely on learned feedforward connectivity, respectively) (**Fig. 7b**, left column, black histogram in top row vs. histograms in bottom row; simple-regime f-RNN: -0.043 $\pm$ 0.010 (mean $\pm$ s.d., n=50 instances), vs. 0, $p < 10^{-9}$; r-RNN, all variants: 0.015 $\pm$ 0.022 (mean $\pm$ s.d., n=240 instances), vs. 0, $p < 10^{-22}$; signed-rank tests), and (2) the alternative implementations of simple- vs. complex- regime f-RNNs (as implied from these networks' alternative behaviors, **Fig. 4**) (**Fig. 7b**, left column and top row, values <0 vs. >0 in black and blue vs. orange and red histograms; simple-regime f-RNNs: -0.025 $\pm$ 0.022 (mean $\pm$ s.d., n=100 instances), vs. 0, $p < 10^{-13}$; complex-regime f-RNNs: 0.018 $\pm$ 0.021 (mean $\pm$ s.d., n=100 instances), vs. 0, $p < 10^{-11}$; signed-rank tests). In this way, mean angle change constitutes a geometric signature of alternative implementations of TI, a prediction that is directly testable on population-level activity from any neural system performing delay-format TI.

8 Behavior systematically predicts activity geometry

The observation that RNN variants performing TI showed signature differences in activity geometry (**Fig. 7**) was reminiscent of the alternative behavioral patterns shown by these network variants (**Fig. 4**). This parallel raised the possibility that solely behavioral results (whether from living subjects or neural models) could be used to make predictions regarding underlying activity geometry. More specifically, explicit quantification of the relationship between behavior and activity geometry would clarify joint neural-behavioral predictions across the RNNs, thus helping adjudicate between alternative neural implementations given empirical data. We therefore examined whether and how activity geometry corresponded with behavior, focusing on the end order effect (**Fig. 3a**, last column), for which RNN variants expressed alternative response-time behavior (**Fig. 4**).

We found that, across RNNs, end order behavior varied systematically with activity geometry (**Fig. 8**). In particular, two patterns of activity geometry (ordered collinearity and mean angle change; **Fig. 8a**, first and fourth columns, respectively) were predicted by behavior across all RNN variants, i.e. all networks regardless of differences in learnable connectivity (f-RNN and r-RNN) or training regime (simple to complex) (**Fig. 8b**). Importantly, specific quantitative ranges of behavior predicted, jointly, both activity geometry and alternative implementations of TI. Behavior values (end order index) >0.25 were uniquely associated with maximal ordered collinearity, thereby constituting evidence for the subtractive implementation. In contrast, behavior values <0.25 were associated with minimal ordered collinearity (and also r-RNNs, see also **Fig. 4b**), thereby constituting evidence for alternative implementations. Further, across all RNNs, alternative behavioral patterns (first vs. second- faster, correponding to behavioral values, >0 vs. <0, respectively) corresponded to the sign of mean angle change (>0 vs. <0, indicating angular alignment and anti-alignment, respectively), suggesting that networks showing second-faster behavior (<0, networks below y-axis in **Fig. 8b**, right) may have similar underlying implementations. Together, these findings clarify the structure of a predictive relationship between an observable behavior and underlying neural activity.

Since particular patterns of activity geometry (**OCI** and **mean angle change**, **Fig. 7**) distinguished between alternative neural implementations of TI, explicit characterization of the relationship between behavior and activity geometry (**Fig. 8**) thus provides additional grounds by which to adjudicate between alternative neural implementations.

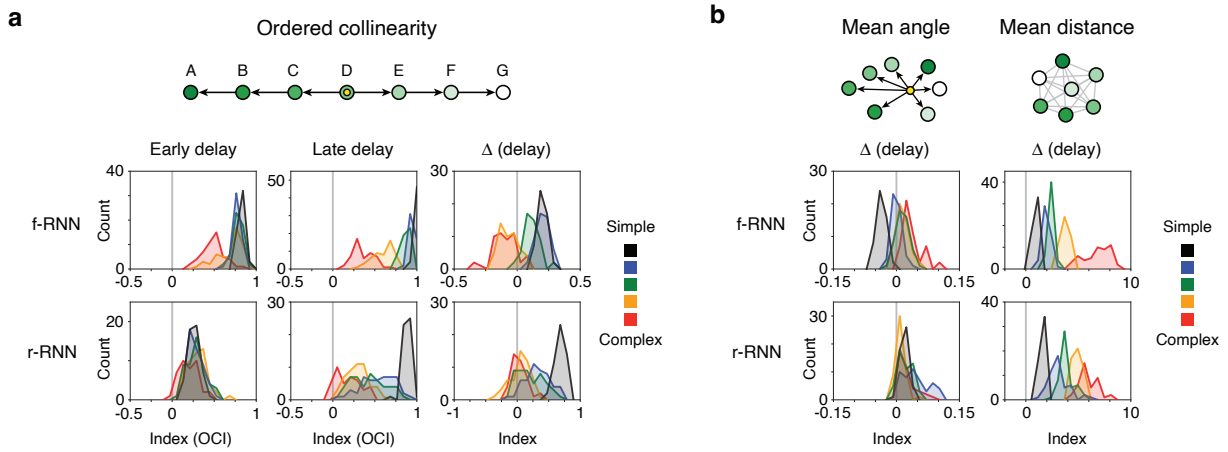

Figure 7: **Geometric signatures of alternative neural implementations of TI. a**, Ordered collinearity: schematic (top) and quantification across RNNs performing TI (bottom). Top, schematic of geometric arrangement in population activity space. The corresponding index (ordered collinearity index, OCI; see Methods) quantifies whether neural activity during the delay matches this geometric arrangement. Angles were measured relative to the cross-condition mean (XCM, yellow circle; compare to Figs. 5a and 6a). Bottom, OCI across RNN variants. OCI was quantified in three ways (columns): early delay (first quarter of delay), late delay (last quarter of delay), and change during delay (last - first quarter). Plots show histograms of instances for each RNN variant. **b**, Mean angle and mean distance: schematic (top) and quantification across RNNs performing TI (bottom). Top, schematic of geometric arrangement in population activity space. For both mean angle and mean distance, all pairwise angles and pairwise distances (not including self-pairs), respectively, were calculated in population activity space (between trial types based on item 1, i.e. A, B, C, etc.; green circles) and averaged. Angles were measured relative to the cross-condition mean (XCM; yellow circle). Bottom, mean angle and mean distance across RNN variants. Specifically, for both activity geometries, the change during delay (last quarter - first quarter) is plotted. All plots show histograms of instances for each RNN variant (n=50 instances / variant, 40 for complex-regime r-RNN; f-RNN and r-RNN, and across training regimes); all quantifications were performed in the top 10 PCs of neural activity during the delay.

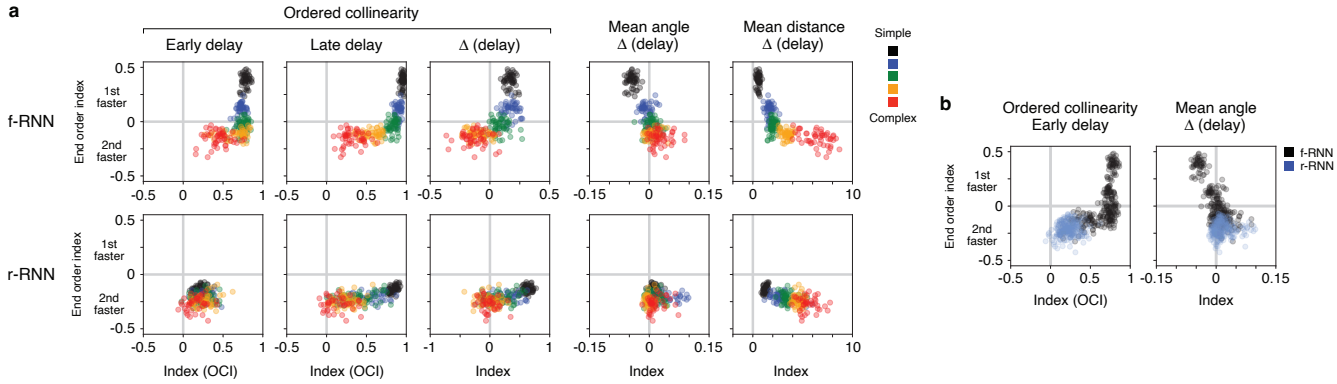

Figure 8: **Behavior systematically predicts the geometry of neural activity. a**, Behavior vs. activity geometry across RNN variants. Behavior (y-axis) is the end order RT pattern (Figs. 3 and 4; quantified by the end order index; see Methods), for which RNNs show alternative versions (1st vs. 2nd-faster; >0 and <0 index values, respectively). Activity geometry (x-axis) correspond to the patterns schematized and quantified in Fig. 7. **b**, Behavior vs. activity geometry across all RNNs. Plots contain the same data as in panel a, but do not show training regime. Note that behavior shows a monotonic relationship to both measures (ordered collinearity in early delay and mean angle change), and that, for mean angle change, alternative behaviors (index values >0 vs. <0) correspond to qualitatively different geometries (<0 vs. >0).

	Behavior	Neural activity			Implication
	"End order" effect	Ordered collinearity	Mean angle change	Oscillation of delay period	
Simple-regime f-RNN	1st-faster, >0.25••	Early delay: >0.75* Late delay: 1*	Negative	∼0.5 cycles/delay XCM cosine: ∼0 Choice cosine: ∼0.5	Feedforward input learned, efficient activity
Simple-regime r-RNN	2nd-faster, <0	Early delay: <0.5* Late delay: 1*	Positive	∼0.5 cycles/delay XCM cosine: ∼0 Choice cosine: ∼0.5	Recurrent dynamics learned, efficient activity
Complex-regime RNNs	2nd-faster, <0.25••	Early delay: <0.75 Late delay: <0.5••	Positive	(variable dynamics)	Diverse dynamics, strong recurrence

Table 4: **Main predictions.** •• a prediction that is a signature of this model, * two predictions that jointly constitute a signature of this model. "End order" effect: **Fig. 4**. Ordered collinearity: **Fig. 7a**. Mean angle change: **Fig. 7b**. Oscillation of delay period: **Figs. 5c, 6c, S8**.

9 Discussion

In this study, we generated and investigated a collection of neural models of transitive inference (TI), a classical cognitive task that distills relational inference into a simple yet essential form. We found that trained recurrent neural networks (RNNs) not only could perform TI — i.e. generalized fully to all novel (untrained) combinations of inputs (**Fig. 2** and **Table 2**) — but also showed behavioral patterns that have long been documented in living subjects performing TI (**Fig. 3**). In addition to these behaviors, these networks also showed a previously unstudied pattern of behavior that manifested in two alternative versions ("end order" effect, **Figs. 3** and **4**). Importantly, RNNs optimized to be efficient (via training regularization and low initial connectivity strength; 'simple' training regime) expressed a solution to TI characterized by relatively simple dynamics and geometry in population activity space (population "subtraction," **Fig. 5**). Lastly, we found that RNNs expressing this simple solution showed systematically different behavior and neural activity patterns compared to other RNNs that also performed TI (**Figs. 4, 6, 7, 8**; summary of predictions in **Table 4**).

Prior work on relational inference in the brain has often focused on task paradigms that rely on stimuli that are challenging to isolate (e.g. spatial tasks [21, 24, 47, 122, 123]), test multiple relations at once (e.g. tasks with linguistic responses and/or episodic elaboration, e.g. [124, 125, 126, 127]), or do not require behavioral report of inference. Possibly as a result, there are relatively few hypotheses and available models that clarify or explain how neural systems accomplish relational inference (i.e. generalize according to the relation) — most notably at the critical explanatory levels of population-level neural activity and behavior. We therefore adopted a task paradigm (TI) and a neural approach (task-trained neural networks) suited to meet these challenges.

Crucial to our approach was to identify, where possible, multiple putative implementations of TI. To do so, we investigated whether and how NNs performed TI when two neurobiologically relevant factors varied — learnable connectivity (i.e. fully-trainable RNNs (f-RNNs) vs. recurrent-trainable (r-RNNs)) and training regime (regularization and initial synaptic strength; simple vs. complex training regimes) (**Fig. 1d**, **Table 1**). We found that each of these NN variants could perform TI (though differing in how often they generalized, **Table 2**), and identified three characteristic types: (1) simple-regime f-RNNs, (2) simple-regime r-RNNs, and (3) complex-regime RNNs, each of which made different testable predictions (summarized in **Table 4**). Behaviorally, these NN variants expressed alternative patterns of response times (RTs) (**Fig. 4**, **Table 4**, second column). Neurally, we found that simple-regime f-RNNs expressed two prominent patterns of neural activity that were jointly sufficient for a population-level implementation of TI (population "subtraction", **Fig. 5e**): first, a rank-ordered linearly arranged set of activity states — which we refer to as "ordered collinearity" — and, second, a single oscillation appropriately oriented in activity space (**Fig. S8**). By investigating these (and other) patterns across networks, we clarified neural activity predictions that distinguished NN variants (**Fig. 7**) and also systematically varied with behavior (**Fig. 8**).

At the same time, the neural activity expressed in different networks performing TI suggested a common principle across networks: namely, rotation. This was at first suggested by the implementation of TI expressed in simple-regime f-RNNs (**Fig. 5**, diagram in **Fig. 5e**; "subtractive" solution), in which a single oscillation serves to rotate activity states in activity space. In this solution, the function of the oscillation is to transform the initially input-driven internal representation of item 1 — a 1-D arrangement of activity states in these networks (ordered collinear arrangement, seen variously in **Fig. 5**, quantified in **Fig. 7a**) — in the direction opposite to that of the input-driven activity shift (corresponding to the presentation of item 1). This re-orientation, which is angular by virtue of the oscillation (in contrast to the input-driven activity shift, which is rectilinear), effectively results in a "subtraction" of item 1 from item 2 along a linear axis in activity space. In light of this solution, we were struck by the fact that a single oscillation also looked to be essential to implementation of transitive comparison in simple-regime r-RNNs (**Fig. 6**), even though these latter networks do not show the same (ordered collinear) arrangement of activity states upon presentation of item 1 (compare **Figs. 5a** and **6a**; quantified in **Fig. 7a**). This suggested that there were additional input-driven activity geometries, besides ordered collinearity, that could nonetheless be transformed by dynamics in a manner enabling generalization. In this way, a single oscillation could be a simple case of a wider set of dynamical patterns enabling transitive generalization, and, further, that the essential operation of this putative wider set of dynamics is rotation. Indeed analysis of dynamics in other RNN variants performing TI suggest a diversity of dynamics (**Fig. S7**). While a non-rotational solution may also be possible, constraints relevant to neurobiology (e.g. metabolic efficiency or the dimensionality of outputs) may favor rotational solutions [103, 128, 117, 129]. Notably, we used the generalized idea of rotational transformation during the delay to identify an additional neural prediction distinguishing RNN variants (mean angle change, **Fig. 7b**; summarized in **Table 4**).

Apart from this broader implication, we note that each of the predictions — behavioral or neural — established in this study can be readily tested in living subjects, and moreover can be used to adjudicate between different neural models of TI (**Table 4**). Surprisingly, the delay-based task format (**Fig. S1**; adopted to enable direct investigation of the interrelationship between WM and relational inference [74, 75, 76, 77]) has not, to our knowledge, been investigated in any prior study of TI. (Indeed it remains possible that prior behavioral and neural findings in TI are ultimately derived from an intrinsically WM-based process in the brain.) It will therefore first be necessary to establish that living subjects can perform TI in a delay format, after which particular behavioral patterns can be assessed. It is worth noting that the present models express behavioral patterns that have been consistently observed in past TI studies, namely the symbolic distance and end item effects (**Fig. 3**). Beyond these patterns, solely behavioral investigation can test the present neural models, specifically whether and to what degree living subjects show the models' novel behavioral prediction ("end order" effect, **Fig. 4**). Further, the alternative versions of this behavioral pattern (first- vs. second-faster, **Fig. 4a**) raises the possibility that this behavior, if observed, may help account for behavioral differences — and by extension differences in underlying neural implementation — between individual subjects. Notably, recent work focusing on other task paradigms also seeks to use behavior to to infer subject-specific latent variables [130, 131, 132].

Complementing behavior, the present models also make a set of population-level neural activity predictions (**Table 4**, columns 3-5). These predictions are most immediately relevant to neural data from brain regions previously linked to relational inference and/or WM, most notably prefrontal cortex (PFC) and hippocampus [133, 134, 135, 136, 137, 138, 139], both required for TI performance [61, 140]. Importantly, testing the predictions we have established has the potential to provide insight not only into how the brain implements TI, but also how these brain regions contribute to the ability to perform other cognitive tasks, particularly those involving WM and relational inference. In particular, each of the three classes of neural models we have identified are associated with different properties with regard to learnable connectivity, efficiency, and dynamics (**Table 4**, Implications). In this way, distinguishing between these models (by way of neural and behavioral data) would suggest that these properties pertain to the neural system in the brain responsible for TI. For example, empirical support for simple-regime r-RNN would suggest that the neural system in the brain underlying TI relies on learned recurrent dynamics rather than learned feedforward input — two fundamentally different implementations of relational knowledge (**Fig. 5e**).

The findings in the present study may also relate to the neural basis of other cognitive functions. Of direct interest are neural activity patterns relevant to abstraction — whether at the level of single cells (e.g. place and grid cell firing [14, 55] and other firing having abstract correlates [141, 142]) or neural populations (e.g. activity geometries [45, 107, 143, 144, 145] and dimensionalities [146, 147, 148, 149] suitable for generalization). In our approach to TI, we deliberately chose not seek to fit or capture these neural activity patterns, and instead stipulated relatively unconstrained neural models. Indeed there may exist important relationships between such activity patterns and those that we find in NNs performing TI. It is also worth emphasizing that our findings do not directly address learning processes, for which prior studies have proposed various models and mechanisms [60, 150, 151, 152, 153, 154]

(including for explicit variants of TI, where human subjects are informed of the transitive hierarchy [155, 156, 157]). Further, our analyses and neural activity predictions focus on delay period activity, leaving open the question of whether and how neural activity following presentation of both items may contribute to transitive generalization. More broadly, it is important to point out that there exist any number of other types of relational inferences (e.g. spatial navigation), and that the relevant brain regions linked to TI and relational inference also support or pertain to cognitive capacities such as structure learning, episodic and semantic memory, and imagination. This convergence of diverse cognitive functions indicates that, toward understanding their biological basis, there is a major need to synthesize approaches.

Acknowledgements

We thank J. Johnston, L. Tian, S. Lippl, M. Triplett, C. Monfredo, N. Biderman, B. Antin, J. Cunningham, and members of the Columbia Center for Theoretical Neuroscience for comments and discussion, and R. Yang for guidance on model training. This work was supported by the the Simons Collaboration for the Global Brain (521921 and 542981), NIH grants (R01 MH090188; R01 MH105174; R01 MH111703, V.F.), the NSF NeuroNex award (DBI-1707398), the Gatsby Charitable Foundation, and an NIMH K99 (MH126158-01A1, KK).

Author contributions

K.K. conceived study in discussion with X.W, V.F., and L.F.A. All authors contributed to study design. K.K. implemented models and performed analyses. K.K. wrote manuscript with input from all authors.

Code availability

Code and trained networks will be made available upon publication.

References

[1] Ortony, A. & Rumelhart, D. E. The representation of knowledge in memory. *Schooling and the acquisition of knowledge* 99–135 (1977).

[2] Rumelhart, D. E., Smolensky, P., McClelland, J. & Hinton, G. E. Schemata and sequential thought processes in pdp models, parallel distributed processing: explorations in the microstructure, vol. 2: psychological and biological models. *Chicago: Psychological and Biological Models* (1986).

[3] Tenenbaum, J. B., Kemp, C., Griffiths, T. L. & Goodman, N. D. How to grow a mind: Statistics, structure, and abstraction. *science* **331**, 1279–1285 (2011).

[4] Hummel, J. E. & Holyoak, K. J. A symbolic-connectionist theory of relational inference and generalization. *Psychological review* **110**, 220 (2003).

[5] Battaglia, P. W. *et al.* Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261* (2018).

[6] Bengio, Y., Lecun, Y. & Hinton, G. Deep learning for ai. *Communications of the ACM* **64**, 58–65 (2021).

[7] Dona, H. S. G. & Chittka, L. Charles h. turner, pioneer in animal cognition. *Science* **370**, 530–531 (2020).

[8] Tolman, E. C. Cognitive maps in rats and men. *Psychol Rev* **55**, 189–208 (1948).

[9] Harlow, H. F. The formation of learning sets. *Psychological review* **56**, 51 (1949).

[10] McGonigle, B. O. & Chalmers, M. Are monkeys logical? *Nature* **267**, 694–696 (1977).

[11] Lazareva, O. & Wasserman, E. Categories and concepts in animals. *Learning Theory and Behavior* 197–226 (2008).

[12] Avargues-Weber, A. & Giurfa, M. Conceptual learning by miniature brains. *Proceedings of the Royal Society B: Biological Sciences* **280**, 20131907 (2013).

[13] O'Keefe, J. & Nadel, L. *The hippocampus as a cognitive map* (Oxford University Press, London, 1978).

[14] Moser, E. I., Moser, M.-B. & McNaughton, B. L. Spatial representation in the hippocampal formation: a history. *Nature neuroscience* **20**, 1448 (2017).

[15] Redish, A. D. *Beyond the cognitive map: From place cells to episodic memory* (MIT Press, Cambridge, MA, 1999).

[16] Behrens, T. E. *et al.* What is a cognitive map? organizing knowledge for flexible behavior. *Neuron* **100**, 490–509 (2018).

[17] Zhou, J. *et al.* Evolving schema representations in orbitofrontal ensembles during learning. *Nature* (2020).

[18] Tervo, D. G. R., Tenenbaum, J. B. & Gershman, S. J. Toward the neural implementation of structure learning. *Current opinion in neurobiology* **37**, 99–105 (2016).

[19] Kemp, C. & Tenenbaum, J. B. The discovery of structural form. *Proceedings of the National Academy of Sciences* **105**, 10687–10692 (2008).

[20] Holyoak, K. J. Induction as model selection. *Proceedings of the National Academy of Sciences* **105**, 10637–10638 (2008).

[21] Eichenbaum, H. The role of the hippocampus in navigation is memory. *J Neurophysiol* **117**, 1785–1796 (2017).

[22] Whittington, J. C. *et al.* The tolman-eichenbaum machine: Unifying space and relational memory through generalization in the hippocampal formation. *Cell* **183**, 1249–1263. e23 (2020).

[23] Kumaran, D. What representations and computations underpin the contribution of the hippocampus to generalization and inference? *Front. Hum. Neurosci.* **6**, 157 (2012).

[24] Tse, D. *et al.* Schemas and memory consolidation. *Science.* **316**, 76–82 (2007).

[25] Tse, D. *et al.* Schema-dependent gene activation and memory encoding in neocortex. *Science* **333**, 891–5 (2011).

[26] Kumaran, D., Hassabis, D. & McClelland, J. L. What learning systems do intelligent agents need? complementary learning systems theory updated. *Trends Cogn Sci* **20**, 512–34 (2016).

[27] Wang, J. X. *et al.* Prefrontal cortex as a meta-reinforcement learning system. *Nature neuroscience* **21**, 860–868 (2018).

[28] Goudar, V., Peysakhovich, B., Freedman, D. J., Buffalo, E. A. & Wang, X.-J. Schema formation in a neural population subspace underlies learning-to-learn in flexible sensorimotor problem-solving. *bioRxiv* (2022).

[29] Eichenbaum, H. & Cohen, N. J. *From Conditioning to Conscious Recollection* (Oxford University Press, New York, 2001).

[30] Hassabis, D. & Maguire, E. A. Deconstructing episodic memory with construction. *Trends Cogn. Sci.* **11**, 299–306 (2007).

[31] Buckner, R. L. The role of the hippocampus in prediction and imagination. *Annual review of psychology* **61**, 27–48– C1–8 (2010).

[32] Addis, D. R. & Schacter, D. L. The hippocampus and imagining the future: where do we stand? *Front Hum Neurosci* **5**, 173 (2011).

[33] Schacter, D. L. *et al.* The future of memory: remembering, imagining, and the brain. *Neuron* **76**, 677–94 (2012).

[34] Redish, A. D. Vicarious trial and error. *Nat Rev Neurosci* **17**, 147–59 (2016).

[35] Miller, K. J., Botvinick, M. M. & Brody, C. D. Dorsal hippocampus contributes to model-based planning. *Nat Neurosci* **20**, 1269–1276 (2017).

[36] Stachenfeld, K. L., Botvinick, M. M. & Gershman, S. J. The hippocampus as a predictive map. *Nature neuroscience* **20**, 1643 (2017).

[37] Elman, J. L. Finding structure in time. *Cognitive science* **14**, 179–211 (1990).

[38] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S. & Dean, J. Distributed representations of words and phrases and their compositionality. In Burges, C., Bottou, L., Welling, M., Ghahramani, Z. & Weinberger, K. (eds.) *Advances in Neural Information Processing Systems*, vol. 26 (Curran Associates, Inc., 2013).

[39] Brown, T. B. *et al.* Language models are few-shot learners. *arXiv preprint arXiv:2005.14165* (2020).

[40] Lample, G. & Charton, F. Deep learning for symbolic mathematics. *arXiv preprint arXiv:1912.01412* (2019).

[41] Davis, E. The use of deep learning for symbolic integration: A review of (lample and charton, 2019). *arXiv preprint arXiv:1912.05752* (2019).

[42] Briggman, K. L., Abarbanel, H. D. & Jr, W. B. From crawling to cognition: analyzing the dynamical interactions among populations of neurons. *Curr Opin Neurobiol* (2006).

[43] Saxena, S. & Cunningham, J. P. Towards the neural population doctrine. *Current opinion in neurobiology* **55**, 103–111 (2019).

[44] Vyas, S., Golub, M. D., Sussillo, D. & Shenoy, K. V. Computation through neural population dynamics. *Annual Review of Neuroscience* **43**, 249–275 (2020).

[45] Chung, S. & Abbott, L. Neural population geometry: An approach for understanding biological and artificial neural networks. *Current opinion in neurobiology* **70**, 137–144 (2021).

[46] Dubreuil, A., Valente, A., Beiran, M., Mastrogiuseppe, F. & Ostojic, S. The role of population structure in computations through neural dynamics. *Nature Neuroscience* 1–12 (2022).

[47] Knierim, J. J. & Hamilton, D. A. Framing spatial cognition: neural representations of proximal and distal frames of reference and their roles in navigation. *Physiological reviews* **91**, 1245–1279 (2011).

[48] Cueva, C. J. & Wei, X.-X. Emergence of grid-like representations by training recurrent neural networks to perform spatial localization. *arXiv [q-bio.NC]* (2018).

[49] Summerfield, C., Luyckx, F. & Sheahan, H. Structure learning and the posterior parietal cortex. *Progress in neurobiology* **184**, 101717 (2020).

[50] Brown, T. B. *et al.* Language models are few-shot learners (2020).

[51] Bryant, P. E. & Trabasso, T. Transitive inferences and memory in young children. *Nature* **232**, 456–458 (1971).

[52] Dusek, J. A. & Eichenbaum, H. The hippocampus and memory for orderly stimulus relations. *Proc.Natl.Acad.Sci.U.S.A.* **94**, 7109–7114 (1997).

[53] Jensen, G. Serial learning. (2017).

[54] Jensen, G., Terrace, H. S. & Ferrera, V. P. Discovering implied serial order through model-free and model-based learning. *Frontiers in neuroscience* **13**, 878 (2019).

[55] Hartley, T., Lever, C., Burgess, N. & O'Keefe, J. Space in the brain: how the hippocampal formation supports spatial cognition. *Philos Trans R Soc Lond B Biol Sci* **369**, 20120510 (2014).

[56] Burt, C. *et al.* Experimental tests of higher mental processes and their relation to general intelligence (1911).

[57] Piaget, J. *Une forme verbale de la comparaison chez l'enfant: un cas de transition entre le jugement prédicatif et le jugement de relation* (éditeur non identifié, 1921).

[58] Vasconcelos, M. Transitive inference in non-human animals: An empirical and theoretical analysis. *Behavioural Processes* **78**, 313–334 (2008).

[59] Gazes, R. P. & Lazareva, O. F. Does cognition differ across species, and how do we know? lessons from research in transitive inference. *Journal of Experimental Psychology: Animal Learning and Cognition* **47**, 223 (2021).

[60] Jensen, G., Terrace, H. S. & Ferrera, V. P. Discovering implied serial order through model-free and model-based learning. *Frontiers in neuroscience* **13**, 878 (2019).

[61] DeVito, L. M., Lykken, C., Kanter, B. R. & Eichenbaum, H. Prefrontal cortex: role in acquisition of overlapping associations and transitive inference. *Learn. Mem.* **17**, 161–167 (2010).

[62] Munoz, F. *et al.* Neuronal responses in posterior parietal cortex during learning of implied serial order. *bioRxiv* 689133 (2019).

[63] Brunamonti, E. *et al.* Neuronal modulation in the prefrontal cortex in a transitive inference task: evidence of neuronal correlates of mental schema management. *Journal of Neuroscience* **36**, 1223–1236 (2016).

[64] Mione, V., Brunamonti, E., Pani, P., Genovesio, A. & Ferraina, S. Dorsal premotor cortex neurons signal the level of choice difficulty during logical decisions. *Cell Reports* **32**, 107961 (2020).

[65] Berens, S. C. & Bird, C. M. Hippocampal and medial prefrontal cortices encode structural task representations following progressive and interleaved training schedules. *bioRxiv* 2021–08 (2022).

[66] Terrace, H. The comparative psychology of ordinal knowledge. (2012).

[67] Grosenick, L., Clement, T. S. & Fernald, R. D. Fish can infer social rank by observation alone. *Nature* **445**, 429–432 (2007).

[68] Tibbetts, E. A., Agudelo, J., Pandit, S. & Riojas, J. Transitive inference in polistes paper wasps. *Biology letters* **15**, 20190015 (2019).

[69] Guez, D. & Audley, C. Transitive or not: a critical appraisal of transitive inference in animals. *Ethology* **119**, 703–726 (2013).

[70] Goldman-Rakic, P. S. Working memory and the mind. *Scientific American* **267**, 110–117 (1992).

[71] Eichenbaum, H. & Cohen, N. J. Working Memory and the Prefrontal Cortex. In *From Conditioning to Conscious Recollection: Memory systems of the brain* (Oxford University Press, 2004).

[72] Carruthers, P. Evolution of working memory. *Proceedings of the National Academy of Sciences* **110**, 10371–10378 (2013).

[73] D'Esposito, M. & Postle, B. R. The cognitive neuroscience of working memory. *Annual Review of Psychology* **66**, 115–142 (2015). PMID: 25251486.

[74] Robin, N. & Holyoak, K. J. Relational complexity and the functions of prefrontal cortex. (1995).

[75] Halford, G. S., Cowan, N. & Andrews, G. Separating cognitive capacity from knowledge: A new hypothesis. *Trends in cognitive sciences* **11**, 236–242 (2007).

[76] Halford, G. S., Wilson, W. H. & Phillips, S. Relational knowledge: The foundation of higher cognition. *Trends in cognitive sciences* **14**, 497–505 (2010).

[77] Holyoak, K. J. Analogy and relational reasoning. (2012).

[78] Evans, J. S. B. & Stanovich, K. E. Dual-process theories of higher cognition: Advancing the debate. *Perspectives on psychological science* **8**, 223–241 (2013).

[79] Libben, M. & Titone, D. The role of awareness and working memory in human transitive inference. *Behavioural processes* **77**, 43–54 (2008).

[80] Acuna, B. D., Eliassen, J. C., Donoghue, J. P. & Sanes, J. N. Frontal and parietal lobe activation during transitive inference in humans. *Cerebral Cortex* **12**, 1312–1321 (2002).

[81] Buzsáki, G., Peyrache, A. & Kubie, J. Emergence of cognition from action. In *Cold Spring Harbor symposia on quantitative biology*, vol. 79, 41–50 (Cold Spring Harbor Laboratory Press, 2014).

[82] Buzsáki, G. *The brain from inside out* (Oxford University Press, 2019).

[83] Liu, Y., Nour, M. M., Schuck, N. W., Behrens, T. E. & Dolan, R. J. Decoding cognition from spontaneous neural activity. *Nature Reviews Neuroscience* **23**, 204–214 (2022).

[84] Wang, J., Narain, D., Hosseini, E. A. & Jazayeri, M. Flexible timing by temporal scaling of cortical responses. *Nature neuroscience* **21**, 102–110 (2018).

[85] Remington, E. D., Egger, S. W., Narain, D., Wang, J. & Jazayeri, M. A dynamical systems perspective on flexible motor timing. *Trends in cognitive sciences* **22**, 938–952 (2018).

[86] Wong, K.-F. & Wang, X.-J. A recurrent network mechanism of time integration in perceptual decisions. *Journal of Neuroscience* **26**, 1314–1328 (2006).

[87] Roitman, J. D. & Shadlen, M. N. Response of neurons in the lateral intraparietal area during a combined visual discrimination reaction time task. *Journal of neuroscience* **22**, 9475–9489 (2002).

[88] Churchland, A. K. *et al.* Variance as a signature of neural computations during decision making. *Neuron* **69**, 818–831 (2011).

[89] Huk, A., Bonnen, K. & He, B. J. Beyond trial-based paradigms: Continuous behavior, ongoing neural activity, and natural stimuli. *Journal of Neuroscience* **38**, 7551–7558 (2018).

[90] Sussillo, D. Neural circuits as computational dynamical systems. *Current opinion in neurobiology* **25**, 156–163 (2014).

[91] Barak, O. Recurrent neural networks as versatile tools of neuroscience research. *Current opinion in neurobiology* **46**, 1–6 (2017).

[92] Yang, G. R. & Wang, X.-J. Artificial neural networks for neuroscientists: A primer. *Neuron* **107**, 1048–1070 (2020).

[93] Bishop, C. M. & Nasrabadi, N. M. *Pattern recognition and machine learning*, vol. 4 (Springer, 2006).

[94] Goodfellow, I., Bengio, Y. & Courville, A. *Deep learning* (MIT press, 2016).

[95] Yamins, D. L. & DiCarlo, J. J. Using goal-driven deep learning models to understand sensory cortex. *Nature neuroscience* **19**, 356 (2016).

[96] Kar, K., Kubilius, J., Schmidt, K., Issa, E. B. & DiCarlo, J. J. Evidence that recurrent circuits are critical to the ventral stream's execution of core object recognition behavior. *Nature neuroscience* **22**, 974–983 (2019).

[97] Michaels, J. A., Schaffelhofer, S., Agudelo-Toro, A. & Scherberger, H. A goal-driven modular neural network predicts parietofrontal neural dynamics during grasping. *Proceedings of the National Academy of Sciences* **117**, 32124–32135 (2020).

[98] Romo, R., Brody, C. D., Hernandez, A. & Lemus, L. Neuronal correlates of parametric working memory in the prefrontal cortex. *Nature* **399**, 470–3 (1999).

[99] Romo, R. & Salinas, E. Flutter discrimination: neural codes, perception, memory and decision making. *Nature Reviews Neuroscience* **4**, 203–218 (2003).

[100] Reinert, S., Hübener, M., Bonhoeffer, T. & Goltstein, P. M. Mouse prefrontal cortex represents learned rules for categorization. *Nature* **593**, 411–417 (2021).

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

[102] Olshausen, B. A. & Field, D. J. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature* **381**, 607–609 (1996). URL `PM:8637596`.

[103] Sussillo, D., Churchland, M. M., Kaufman, M. T. & Shenoy, K. V. A neural network that finds a naturalistic solution for the production of muscle activity. *Nature neuroscience* **18**, 1025–1033 (2015).

[104] Cueva, C. J., Wang, P. Y., Chin, M. & Wei, X.-X. Emergence of functional and structural properties of the head direction system by optimization of recurrent neural networks. *arXiv preprint arXiv:1912.10189* (2019).

[105] Schuessler, F., Mastrogiuseppe, F., Dubreuil, A., Ostojic, S. & Barak, O. The interplay between randomness and structure during learning in rnns. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. & Lin, H. (eds.) *Advances in Neural Information Processing Systems*, vol. 33, 13352–13362 (Curran Associates, Inc., 2020).

[106] Saxe, A., Nelli, S. & Summerfield, C. If deep learning is the answer, what is the question? *Nature Reviews Neuroscience* **22**, 55–67 (2021).

[107] 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).

[108] Orhan, A. E. & Ma, W. J. A diverse range of factors affect the nature of neural representations underlying short-term memory. *Nature neuroscience* **22**, 275–283 (2019).

[109] Richards, B. A. *et al.* A deep learning framework for neuroscience. *Nature neuroscience* **22**, 1761–1770 (2019).

[110] Turner, E., Dabholkar, K. V. & Barak, O. Charting and navigating the space of solutions for recurrent neural networks. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P. & Vaughan, J. W. (eds.) *Advances in Neural Information Processing Systems*, vol. 34, 25320–25333 (Curran Associates, Inc., 2021).

[111] De Lillo, C., Floreano, D. & Antinucci, F. Transitive choices by a simple, fully connected, backpropagation neural network: implications for the comparative study of transitive inference. *Animal Cognition* **4**, 61–68 (2001).

[112] Chaisangmongkon, W., Swaminathan, S. K., Freedman, D. J. & Wang, X.-J. Computing by robust transience: how the fronto-parietal network performs sequential, category-based decisions. *Neuron* **93**, 1504–1517. e4 (2017).

[113] Wynne, C. Pigeon transitive inference: Tests of simple accounts of a complex performance. *Behavioural Processes* **39**, 95–112 (1997).

[114] Clark, H. H. Linguistic processes in deductive reasoning. *Psychological review* **76**, 387 (1969).

[115] Sussillo, D. & Barak, O. Opening the black box: low-dimensional dynamics in high-dimensional recurrent neural networks. *Neural computation* **25**, 626–649 (2013).

[116] Mante, V., Sussillo, D., Shenoy, K. V. & Newsome, W. T. Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature* **503**, 78–84 (2013).

[117] Driscoll, L., Shenoy, K. & Sussillo, D. Flexible multitask computation in recurrent networks utilizes shared dynamical motifs. *bioRxiv* (2022).

[118] Kobak, D. *et al.* Demixed principal component analysis of neural population data. *Elife* **5** (2016).

[119] Kaufman, M. T. *et al.* The largest response component in the motor cortex reflects movement timing but not movement type. *eneuro* **3** (2016).

[120] Churchland, M. M. *et al.* Neural population dynamics during reaching. *Nature* **487**, 51–6 (2012).

[121] Kao, J. C. Considerations in using recurrent neural networks to probe neural dynamics. *Journal of Neurophysiology* **122**, 2504–2521 (2019).

[122] McKenzie, S. *et al.* Hippocampal representation of related and opposing memories develop within distinct, hierarchically organized neural schemas. *Neuron* **83**, 202–15 (2014).

[123] Wood, R. A. *et al.* The honeycomb maze provides a novel test to study hippocampal-dependent spatial navigation. *Nature* **554**, 102–105 (2018).

[124] Hassabis, D., Kumaran, D., Vann, S. D. & Maguire, E. A. Patients with hippocampal amnesia cannot imagine new experiences. *Proc Natl.Acad.Sci.U.S.A* **104**, 1726–1731 (2007).

[125] Duff, M. C. & Brown-Schmidt, S. The hippocampus and the flexible use and processing of language. *Frontiers in human neuroscience* **6**, 69 (2012).

[126] Duff, M. C., Kurczek, J., Rubin, R., Cohen, N. J. & Tranel, D. Hippocampal amnesia disrupts creative thinking. *Hippocampus* **23**, 1143–1149 (2013).

[127] De Brigard, F. & Gessell, B. S. *Time Is Not of the Essence: Understanding the Neural Correlates of Mental Time Travel* (Oxford University Press, New York, 2016).

[128] Sabatini, D. A. & Kaufman, M. T. A curved manifold orients rotational dynamics in motor cortex. *bioRxiv* (2021).

[129] Libby, A. & Buschman, T. J. Rotational dynamics reduce interference between sensory and memory representations. *Nature neuroscience* **24**, 715–726 (2021).

[130] Dezfouli, A., Griffiths, K., Ramos, F., Dayan, P. & Balleine, B. W. Models that learn how humans learn: the case of decision-making and its disorders. *PLoS computational biology* **15**, e1006903 (2019).

[131] Jaffe, P. I., Poldrack, R. A., Schafer, R. J. & Bissett, P. G. Discovering dynamical models of human behavior. *bioRxiv* (2022).

[132] Roy, N. A. *et al.* Extracting the dynamics of behavior in sensory decision-making experiments. *Neuron* **109**, 597–610 (2021).

[133] Waltz, J. A. *et al.* A system for relational reasoning in human prefrontal cortex. *Psychological Science* **10**, 119–125 (1999).

[134] Reber, T. P., Luechinger, R., Boesiger, P. & Henke, K. Unconscious relational inference recruits the hippocampus. *Journal of Neuroscience* **32**, 6138–6148 (2012).

[135] Eichenbaum, H. Memory: Organization and control. *Annu Rev Psychol* **68**, 19–45 (2017).

[136] Morton, N. W. & Preston, A. R. Concept formation as a computational cognitive process. *Current Opinion in Behavioral Sciences* **38**, 83–89 (2021).

[137] Zeithamova, D., Schlichting, M. & Preston, A. The hippocampus and inferential reasoning: building memories to navigate future decisions. *Frontiers in Human Neuroscience* **6** (2012).

[138] Zeithamova, D. *et al.* Brain mechanisms of concept learning. *Journal of Neuroscience* **39**, 8259–8266 (2019).

[139] Wendelken, C. & Bunge, S. A. Transitive inference: distinct contributions of rostrolateral prefrontal cortex and the hippocampus. *Journal of cognitive neuroscience* **22**, 837–847 (2010).

[140] Devito, L. M., Kanter, B. R. & Eichenbaum, H. The hippocampus contributes to memory expression during transitive inference in mice. *Hippocampus* **20**, 208–217 (2010).

[141] Shima, K., Isoda, M., Mushiake, H. & Tanji, J. Categorization of behavioural sequences in the prefrontal cortex. *Nature* **445**, 315–8 (2007).

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

[143] Bernardi, S. *et al.* The geometry of abstraction in the hippocampus and prefrontal cortex. *Cell* **183**, 954–967. e21 (2020).

[144] Sheahan, H., Luyckx, F., Nelli, S., Teupe, C. & Summerfield, C. Neural state space alignment for magnitude generalization in humans and recurrent networks. *Neuron* **109**, 1214–1226.e8 (2021).

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

[146] Mastrogiuseppe, F. & Ostojic, S. Linking connectivity, dynamics, and computations in low-rank recurrent neural networks. *Neuron* **99**, 609–623.e29 (2018).

[147] Cueva, C. J. *et al.* Low-dimensional dynamics for working memory and time encoding. *Proceedings of the National Academy of Sciences* **117**, 23021–23032 (2020).

[148] Beiran, M., Meirhaeghe, N., Sohn, H., Jazayeri, M. & Ostojic, S. Parametric control of flexible timing through low-dimensional neural manifolds. *bioRxiv* (2021).

[149] Farrell, M., Recanatesi, S., Moore, T., Lajoie, G. & Shea-Brown, E. Gradient-based learning drives robust representations in recurrent neural networks by balancing compression and expansion. *Nature Machine Intelligence* **4**, 564–573 (2022).

[150] Wu, X. & Levy, W. B. A hippocampal-like neural network model solves the transitive inference problem. In *Computational Neuroscience*, 567–572 (Springer, 1998).

[151] Delius, M. S. J. D. Algebraic learning and neural network models for transitive and non-transitive responding. *European Journal of Cognitive Psychology* **10**, 307–334 (1998).

[152] Frank, M. J., Rudy, J. W. & O'Reilly, R. C. Transitivity, flexibility, conjunctive representations, and the hippocampus. ii. a computational analysis. *Hippocampus* **13**, 341–354 (2003).

[153] Kumaran, D. & McClelland, J. L. Generalization through the recurrent interaction of episodic memories. *Psychol. Rev.* **119**, 573–616 (2012).

[154] Ciranka, S. *et al.* Asymmetric reinforcement learning facilitates human inference of transitive relations. *Nature Human Behaviour* **6**, 555–564 (2022).

[155] Park, S. A., Miller, D. S., Nili, H., Ranganath, C. & Boorman, E. D. Map making: constructing, combining, and inferring on abstract cognitive maps. *Neuron* **107**, 1226–1238 (2020).

[156] Russin, J., Zolfaghar, M., Park, S. A., Boorman, E. & OâReilly, R. C. Complementary structure-learning neural networks for relational reasoning. In *CogSci... Annual Conference of the Cognitive Science Society. Cognitive Science Society (US). Conference*, vol. 2021, 1560 (NIH Public Access, 2021).

[157] Nelli, S., Braun, L., Dumbalska, T., Saxe, A. & Summerfield, C. Neural knowledge assembly in humans and deep networks. *bioRxiv* (2021).

[158] Marois, R. & Ivanoff, J. Capacity limits of information processing in the brain. *Trends in Cognitive Sciences* **9**, 296–305 (2005).

[159] Davis, H. Logical transitivity in animals. In *Cognitive aspects of stimulus control*, 405–429 (Psychology Press, 2018).

[160] Paszke, A. *et al.* Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* **32** (2019).

[161] Murray, J. D. *et al.* A hierarchy of intrinsic timescales across primate cortex. *Nat Neurosci* **17**, 1661–3 (2014).

[162] Cavanagh, S. E., Towers, J. P., Wallis, J. D., Hunt, L. T. & Kennerley, S. W. Reconciling persistent and dynamic hypotheses of working memory coding in prefrontal cortex. *Nature communications* **9**, 1–16 (2018).

[163] Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization (2014). URL `https://arxiv.org/abs/1412.6980`.

[164] Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning representations by back-propagating errors. *Nature* **323**, 533–536 (1986).

[165] Krawczyk, D., McClelland, M. & Donovan, C. A hierarchy for relational reasoning in the prefrontal cortex. *Cortex; a journal devoted to the study of the nervous system and behavior* **47**, 588–97 (2011).

Methods

Task

Transitive inference (TI) is a classic cognitive task that requires subjects to infer an abstract ordered relation – here, transitivity (**Fig. 1a**) – between items not previously observed together, i.e. using A > B and B > C to infer A > C (**Fig. 1b**). TI defines test cases expressly as novel recombinations of training inputs, thus primarily testing relational rather than statistical inference. We focused on a 7-item version of TI, in which there are 12 training trials and 30 test trials (training: A vs. B, B vs. C, etc.; test: A vs. C, B vs. D, etc.; see **Fig. 1b** for diagram of trial types and correct responses), though in pilot work we found that our approach could be generalized to fewer or more items with qualitatively similar results.

Given the interrelationship between relational inference (exemplified by TI) and working memory (WM) [75, 76], we investigated TI in a task format that explicitly imposes a delay that necessitates WM (**Fig. S1a**, delay format). Further, for comparison to previous modeling work [60, 111] and for potential insight, we also studied a no-delay task format, for which the presentation of task items (A, B, C, etc.) is simultaneous (diagram in **Fig. 1c**). It is worth noting that in some prior TI studies, a delay between stimuli is implicit in the free-exploration afforded to subjects (e.g. [52]).

Besides WM, an important difference between the delay vs. no-delay formats is that the no-delay version requires twice as many input parameters as the delay version (e.g. twice as many input connections in a neural system). This difference may make the delay version not only more difficult to perform, but also more neurobiologically accurate with respect to neural systems underlying abstract cognition: these systems look to receive extremely diverse inputs, implying that the extent of input connectivity is relatively constrained [158].

Input stimuli (items)

A panel of input stimuli corresponding to items A, B, C, etc. was constructed for each model instance. In TI in living subjects, an item is an arbitrary sensory object (e.g. image, odor) with no features that are significant *a priori*. To capture this property, items were represented as randomly generated input vectors $\mathbf{u}$ ($\mathbf{u}^A$, $\mathbf{u}^B$, etc.), modeling sensory-driven activity in upstream neurons. For simplicity, each $\mathbf{u}$ was drawn from a multivariate standard normal distribution of dimension N^{in}.

The identity of items 1 and 2 varied by trial type (e.g. AB, BA, AC, BC, etc.; see **Fig. 1b** for all trial types). N^{in} was chosen to be 100, matching the size of the hidden layer of the neural models (MLP and RNN), which was set to N = 100 (see below). This ensured that input corresponding to each item presentation elicited patterns of activation that would be uncorrelated in the activity space of the neural models – at least prior to training – thereby simulating arbitrary sensory stimuli. Note that the TI (and relational inference more generally) is not defined in terms of particular stimulus features, and indeed is most rigorously tested in the absence of stimulus features indicating items' rank in the transitive hierarchy [159].

Model architectures

Three model architectures were studied: a recurrent neural network (RNN), logistic regression (LR), and a multi-layer perceptron (MLP). Each was implemented in Python using the NumPy and PyTorch [160] packages, in addition to custom code for RNNs and all subsequent analyses.

Recurrent neural network (RNN). To investigate population-level neural dynamics, we studied the standard continuous-time RNN:

$$\tau \dot{x_i}(t) = -x_i(t) + \sum_{k=1}^{N} J_{ik} r_k(t) + \sum_{k=1}^{N^{in}} B_{ik} u_k(t) + b_i + \eta_i(t)$$

where x_i are activity of the recurrent units, r_i are the corresponding rates, N is the number of recurrent units (100 for all networks), N^{in} is the number of input units, and τ is the time constant. The rates r_i derive from the activations x_i via a tanh nonlinearity, $r_i = \tanh(x_i)$.

The tanh non-linearity was chosen because we found it to be the most effective for generating and analyzing network dynamics; further, in pilot work we found that other non-linearities (e.g. rectified tanh) yielded RNNs that exhibited unrealistic behavioral patterns when simulated (see below for description of behavioral simulations; rectified-tanh RNNs exhibited an unrealistic bias in choice 1 vs. choice 2). We note that RNNs in the present study are intended to model neural activity (generate testable predictions) expressly at the population level.

The network units interact via the recurrent synaptic weight matrix $\mathbf{J}$. The input to the system is $\mathbf{u}$, the activity of the set of N^{in} input units that influence the network through input weights $\mathbf{B}$. The output of the system is $\mathbf{z}$, the activity of a set of N^{out} output units, each defined to be a linear readout of network activity:

$$z_i(t) = \sum_{k=1}^{N^{out}} W_{ik} r_k(t) + b_i$$

Each output unit z_i is a weighted sum of network rates with weights, $\mathbf{W}$, with a constant bias, b_i. In all models, three output units were implemented ($N^{out} = 3$), corresponding to three alternative behavioral actions (see below, Model outputs). All analyses of neural activity were performed on x_i. Note that analyses of r_i would yield similar predictions.

Network simulations were performed using Euler's method with discrete time step $\Delta t = \tau/10$. Intrinsic single-unit noise $\eta_i(t)$ was generated at each time step by drawing values from a Gaussian random variable with zero mean and s.d. of 0.2.

Prior to training, the parameters of the model were initialized as follows. The entries of $\mathbf{J}$ were initialized as draws from a Gaussian random variable with zero mean and variance g_0^2/N. The entries of $\mathbf{B}$ were initialized as draws from a Gaussian random variable with zero mean and variance h_0^2/N^{in}. The elements of $\mathbf{W}$, and all bias terms, were initialized to 0. Both h_0 and g_0 were hyperparameters that were systematically varied across RNNs (see **Table 1** and RNN variants below).

Logistic regression (LR). The LR model (schematic in **Fig. S2a**) was studied to clarify the possible role of feed-forward connectivity in performing TI. Each LR consisted of two linear readouts (corresponding to choice 1 and 2) each of which had coefficients for every input dimension N^{in} for each of the two items presented. In each simulation of the model, Gaussian noise (zero mean, s.d. of 0.2) was added to each of the readouts.

Multi-layer perceptron (MLP). In addition to the LR model, single-layer MLPs (schematic in **Fig. S2a**; N = 100 hidden units, fully connected; see also [111] for study of MLPs of smaller size, i.e. three hidden units solving five-item TI) were studied to clarify the possible role of feedforward connectivity in TI. Entries of the input weight matrix were initialized as Gaussian entries with 0 mean and variance h_0^2/N^{in}, with $h_0 = 1$; entries of the output weight matrix were initialized to 0; all biases were initialized to 0. In each simulation of the model, Gaussian noise (zero mean, s.d. of 0.2) was added to each of the hidden units.

Model input

RNN. The input to the RNN in trial m, $\mathbf{u}(t, m)$, consisted of the presentation of items 1 and 2 with an intervening delay, dividing three periods in each trial: rest, delay, and choice (**Fig. S1a**), with durations 0.5 τ, 2 τ, and 2 τ, respectively. These durations were sufficiently long to yield differing neural implementations of TI in trained networks, and also because this was the minimal (delay) duration relevant to WM [161, 162]; in pilot work, we found that RNNs trained with longer durations yielded similar results. Item presentations were modeled as instantaneous pulses (Δt), as TI (and relational inference more generally) is not dependent on sensory input of a particular duration. The input over the course of the trial is diagrammed in **Fig. S1b**.

MLP and LR. The input to the feedforward models (LR and MLP) in trial m, $\mathbf{u}(m)$, consisted of the joint (simultaneous) presentation of items 1 and 2, requiring twice the input dimensionality of RNNs (i.e. given a fixed dimensionality for individual input stimuli, i.e. items); thus the feedforward models had $N^{in} = 200$ rather than $N^{in} = 100$ as in the RNNs (diagrammed in **Fig. S2a**).

Model output

RNN. The output from the RNN was composed of three output units z_1, z_2, z_3 corresponding to three behaviors: choice 1, choice 2, and rest, respectively. In training, the target output $\hat{\mathbf{z}}(t, m)$ was defined for every time point t and for each trial type m such that the correct output unit was activated above resting levels during the period following the presentation of the second item (target values: resting: 0, activated: 5; diagram of target output in **Fig. S1b**).

The target values of output units are given by (diagrammed in **Fig. S1b**):

$$y = \frac{1}{MTN^{out}} \sum_{m,t,i=1}^{M,T,N} r_i(c,t)^2 \Delta t$$

where M is the number of training trial types (12 total, see **Fig. 1b**), T is the number of timesteps in each trial, and N^{out} is the number of output units. In example outputs (**Fig. 2**, top row, and **Fig. S3a**), the choice value plotted was the difference in the readout values for z_1 and z_2 averaged over the last half of the choice period and normalized to the magnitude of the largest such difference value across all trial types.

RNN behavior. The behavior – i.e. the choice response and response time (RT) – of an RNN in a trial was defined using an established criterion (see [112]). The z_1 (choice 1) and z_2 (choice 2) output units were passed through a simple monotonic saturating function ranging in value from 0 to 1:

$$\tilde{z}_i = \frac{1}{2}\tanh(z_i - \frac{\hat{z}_i}{2}) + \frac{1}{2}$$

where $\hat{z}_i$ is the target value of the output unit.

The **response** (i.e. choice in trial) was defined by the identity of the output unit (choice 1 vs. choice 2, see above) that first reached a fixed threshold value of 85% in the choice period, and normalized to the duration of the choice period (0 to 1, where 1 is 'max' in plots of RT).

The **RT** was defined as the time of the response, measured as the time elapsed from t_2 (the time of presentation of item 2). Under certain conditions (i.e. when additional noise was added to RNNs (see further below), or cases when the RNN was presented with same-item stimuli, e.g. AA, BB, CC, etc., **Fig. 2**; these trial types were not evaluated in this study), the threshold was not reached for output unit. These trials did not count as correct and are shown in plots as 'no response' trials (**Fig. 2**).

MLP and LR. For both feedforward models, the output was composed of two output units z_1, z_2, corresponding to choice 1 and choice 2, respectively. In training, the target output $\hat{\mathbf{z}}(m)$ was defined for each trial type m such that the correct output unit for each trial type (choice 1 vs. 2, **Fig. 1b**) was activated (value for active: 1, value for not active: 0). The response for a given trial was defined by the identity of the output unit which had the higher activity value. In example outputs (**Fig. S2b**), the choice value plotted was the difference in the readout values for z_1 and z_2 normalized to the magnitude of the largest such difference value across all trial types.

Model training

Models were optimized (trained) solely on training trials and not test (inference) trials. The ability of trained models to respond correctly to inference trials thereby mirrors that of living subjects that have only experienced or learned from training trials. In this way, analysis of models that respond correctly on test trials (i.e. perform inference) can be studied to identify putative neural implementations in the brain. Parameter updates were performed for batches of training trials, where each batch consisted of 128 trials randomly sampled from the training trial types defined by the task (diagrammed in **Fig. 1b**).

RNN. RNNs were trained to minimize E_{task}, the average squared difference between $\mathbf{z}(t,m)$, the readout of the network on trial m, and $\hat{\mathbf{z}}(t,m)$, the target output for that trial:

$$E_{task} = \frac{1}{MTN^{out}} \sum_{m,t,j=1}^{M,T,N^{out}} (z_j(t,m) - \hat{z}_j(t,m))^2$$

where m corresponds to different training trials, T corresponds to the length of the trial (in time steps), and N^{out} is the number of readout units. E_{task} stipulates that the optimization procedure generate networks that respond correctly in training trials.

The overall error function E was comprised of E_{task} and two additional terms that implement regularization, which has been found to promote neurobiologically accurate solutions in trained RNNs [Sussillo, Cuevas, Beiran]. The two terms were R_{L2}, a standard L2 regularization on input and output synaptic weights, and R_{FR}, a regularization on the network rates.

The overall error function was

$$E = E_{task} + \alpha R_{L2} + \beta R_{FR}$$

where the α and β hyperparameters set the strength of each type of regularization.

The first regularization term is a standard L2 penalty on input and output synaptic weights:

$$R_{L2} = \sum_{i,j=1}^{N,N^{in}} B_{ij}^2 + \sum_{i,j=1}^{N^{out},N} W_{ij}^2$$

The second regularization term is a metabolic penalty on rates in the network:

$$R_{FR} = \frac{1}{MTN^{out}} \sum_{m,t,i=1}^{M,T,N} r_i(c,t)^2 \Delta t$$

Both terms have been found to promote neurobiologically realistic responses in trained RNNs [48, 103].

The objective of training was to minimize E by modifying the network parameters **J**, **B**, **W**, **x**(t=0), and constant bias terms.

Training was implemented with the Adam optimizer [163], with updates to the network parameters calculated using backpropagation through time [164, 165]. Parameter updates were performed for training sets (batches) of 128 trials, where the trials in each batch were randomly sampled from the task-defined training trials (diagrammed in **Fig. 1b**). Training was stopped when the RNN responded correctly on all trial types (training and test; see above for response criteria) in the absence of noise (η_i set to 0), or when E_{task} fell below 0.1. Up to 30,000 training epochs were run.

MLP and LR. Each feedforward model was trained using the Adam optimizer with a cross-entropy loss. For the LR models, training was performed to convergence (i.e. until loss did not improve for 1000 training epochs); for MLPs, training was performed until the network responded correctly on all trial types (training and test; see above for response criteria) in the absence of noise (η_i set to 0). For both models, a weight-decay term (given by the L2-norm of all parameters) scaled by hyperparameter α was included in the loss function, with similar results across a range of values (presented are 0.1 for LR and 0.001 for MLP).

RNN variants

To identify multiple biologically relevant neural implementations of TI, two classes of RNN variants were studied:

First, RNN variants that differed in learnable connectivity: f-RNN and r-RNN. f-RNNs were RNNs where all connection weights (feedforward and recurrent) were trainable; r-RNNs were RNNs for which feedforward weights (**B** and **W**, see above) were not allowed to be modified from their initial random values (Gaussian draws) in training. In addition, we separately found that RNNs for which feedforward input weights (**B**) were not allowed to change showed similar results to r-RNNs. Since these latter RNNs were a relatively closer point of comparison to f-RNNs, these models were analyzed in **Fig. 5** and **Fig. S7** (third row).

Second, RNN variants that differed with respect to initial connectivity strength (prior to training) and regularization (during training) – jointly termed "training regime" and defined for five classes (**Table 1**): simple (high), simple (low), intermediate, complex (low), and complex (high). In general we use the terms "simple-regime" and "complex-regime" to refer to the simple (high) and complex (high) training regimes, respectively, though in some cases we use "simple-regime" to refer to both simple (high) and simple (low) (and similarly for "complex-regime"). If the case, this is made explicit where presented. Ten RNN variants were studied in all: two types of learnable connectivity variants (f-RNN and r-RNN) by five types of training regime variants.

In addition to these two types of variants, for each RNN variant (e.g. simple-regime f-RNN) a collection of individual instances (randomly initialized networks, given initial connectivity strength parameters in **Table 1**) were studied. In particular, we trained 100 instances of each RNN variant, subsequently studying only those models that performed TI perfectly (correct responses to all trial types) under noise-free conditions (η_i set to 0). This subset of instances were then subject to behavioral and neural analyses.

Behavior simulation and behavioral patterns.

To investigate behavioral patterns across models (**Fig. 3**, **4**, **8**, **S2d**, **S4**), models were simulated on all 42 trial types (12 training and 30 test, **Fig. 1b**). To simulate average levels of performance that were realistic to living subjects performing TI and showing characteristic behavioral patterns (>50% performance in training trials and <100% performance on trial types with large symbolic distance; see example monkey data in **Fig. S4a**), we took the following approach.

Simulation approach. For each model instance that performed perfectly (correct responses on all trial types) under noise-free conditions, we added progressively larger amounts of intrinsic noise to model units (s.d. of η_i, increased from 0.5 in increments of 0.1; LR: readout units, MLP: hidden-layer units, RNN: hidden-layer (recurrent) units) until the performance of the model (averaged over 500 simulations across all trial types) satisfied these basic performance criteria: >50% training performance (moreover for both choice 1 and 2 training trials) and <95% performance on the largest symbolic distance trials (AG and GA).

With the addition of noise, a subset of simulated trials (∼20%) did not meet the output activity threshold criterion (fixed at 85%) for a response (no response trials, see above). For simplicity, we designated the choice in these trials to be randomly either 1 or 2 in calculating average performance, and did not include these trials in RT analysis (note also that alternative approaches to defining the model response can be employed [148]). The first 50 RNN instances (where available) to pass these the basic performance criteria were subsequently analyzed. For subsequent behavioral analysis, 5000 simulations of all trial types at the identified noise level were run.

Symbolic distance effect. Trial types differing by the magnitude of the difference in rank between items (i.e. distance 1: AB, BA, BC, CB, CD, DC, DE, ED, EF, FE, FG, GF; 2: AC, CA, BD, DB, CE, EC, DF, FD, EG, GE; 3: AD, DA, BE, EB, CG, GC; 4: AE, EA, BF, FB, CG, GC, 5: AF, FA, BG, GB, 6: AG, GA). Schematic of trial types in Fig. 3a, second column.

End item effect. Trial types differing by whether or not they contain end items A or G. Schematic of trial types in Fig. 3a, third column.

Lexical marking effect. Trial types containing end item A vs. end item G. Schematic of trial types in Fig. 3a, third column. Note that trial types containing both A and G (i.e. AG and GA) were not included.

"End order" effect. Trial types for which end items (A or G) were presented 1st (item 1) vs. 2nd (item 2). Schematic of trial types in Fig. 3a, third column. Note that trial types containing both end items (i.e. AG and GA) were not included. The end order effect refers the (sequential) order of item presentation, and is therefore specific to the delay task format of TI (**Fig. 1c**, **Fig. S1a**).

The end order index (**Figs. 4, 8**) was defined as:

$$Index = \frac{RT_{2nd} - RT_{1st}}{RT_{1st} + RT_{2nd}}$$

where RT_{1st} is the average RT over trials where end items were presented first (item 1), and RT_{2nd} is the average RT over trials where end items were presented second (item 2).

Visualization of population activity

To clarify the neural implementation of TI in RNNs that performed the task, we visualized population-level neural activity by performing PCA of activity vectors across all time points and trial types (12 training and 30 test, **Fig. 1b**) under noise-free conditions (η_i set to 0) (% variance explained in **Table 3**). Activity was plotted in the top 3 PCs (**Fig. 5a**, **6a**).

Task–relevant activity axes

Visualization of population activity in RNNs performing TI (**Figs. 5**, **6**) suggested that the underlying neural implementations were characterized by specific arrangements (in activity space) of activity states with respect to two directions (axes), each defined on the basis of different trial types in the task (visual schematic of each axis in **Fig. S8a**).

The first was the **choice axis**, defined as the direction pointing from the activity states of Choice 1 vs. Choice 2 trial types (21 trial types each; red vs. blue, respectively, in **Fig. 1b**) during the choice period (period following the delay, **Fig. S1**a). The choice axis was calculated as a unit-normalized vector pointing from the mean of activity vectors across Choice 1 trial types to the mean of activity vectors in Choice 2 trial types, both for activity averaged over the first quarter of the choice period.

The **cross-condition mean** (**XCM**; yellow line in **Figs. 5a**, **5b**; [118, 119, 120]), was defined and calculated as the average trajectory across all trial types. The XCM was moreover calculated in two ways: for visualizations, the XCM was calculated for every time point (Fig. 5a, 5b); for quantifications, the XCM was calculated by taking the mean neural activity across a given time window (e.g. first or last quarter of the delay period). The **XCM axis**, defined as the direction pointing from the activity states at the beginning of the delay period to the end of the delay period, was calculated as a unit-normalized vector pointing from the XCM of the first quarter of the delay period to the XCM of the last quarter of the delay period.

Inference of linear dynamics

To identify dynamical components expressed in RNNs performing TI, we fit neural activity from the delay period to an unconstrained linear dynamics model ($\dot{X} = XA$; least-squares fit) from noise-free simulations (η_i set to 0) across all time points during the delay and across all trial types. In the delay period there are 7 trial types corresponding to each possible item 1 (A through G). The fit was performed for the top 10 PCs of delay-period activity. R^2 values of the fit were relatively high (0.5-0.9; see **Fig. S7**, first column, for values across RNNs). Eigenvalues of the A matrix were subsequently plotted for each type of RNN variant (**Fig. S7**, second through sixth columns).

Fixed point analysis and linearization.

To identify dynamical components of RNNs performing TI, we used fixed-point analysis and linear approximation methods [115, 121, 116].

Fixed-point finding was implemented using custom code in PyTorch, following an established method [115, 121]. Optimization via gradient descent (Adam optimizer) was used to identify activity states in which the speed of RNN dynamics was minimized (mean-squared error loss). The optimization was seeded using activity states from noise-free simulations of trial types in the TI task (specifically at these time points: 0, time of item 1 presentation, time of item 2 presentation, halfway through delay, the last time point, and the last timestep when trials were simulated with 100 additional timesteps), in addition to 5 batches of 50 activity state seeds, each of which were drawn randomly from activity states of trials in which item 1 and item 2 were randomly jittered in time across the trial. Each batch was optimized for 50000 epochs and stopped after 5000 epochs with no improvement in loss. Candidate FPs were those activity states for which the speed was lower than 10^{-5}. Redundant candidate FPs were eliminated by requiring that, between candidate FPs, the activity of every recurrent unit differed by more than 10^{-5}.

To obtain the Jacobian matrix A of linearization, two methods were used. The first was analytic (based on the weight matrix [115]); the second was numerical [92], using the function grad() in the PyTorch autograd library: at each fixed point, the function grad() was used to calculate the entries of A for the trained (and frozen) RNN (with no external input). Each approach yielded the same results.

Oscillation of transitive comparison.

To identify the putative oscillation associated with transitive comparison (comparison oscillation), for each RNN performing TI we detected the oscillatory mode (mode with eigenvalue having a non-zero imaginary component) of highest frequency (e.g. highlighted in arrowheads in **Fig. 5b** and **6b** for example networks), as this was the mode associated with transitive comparison in simple-regime RNNs (modes with frequency ∼0.5 cycles/delay, **Figs. 5d, 6d**). This detection was performed from the inferred linear dynamics matrix (with eigenvalue spectra across RNNs in **Fig. S7**), as this can also be equivalently performed for experimental neural data. The 2D linear subspace (plane) of the identified oscillatory mode was defined by the eigenvectors, which were orthogonalized and unit-normalized prior to being used to visualize population activity (**Fig. 5b**, right) and to quantify model predictions regarding activity geometry (angles of task-relevant axes with respect to the oscillation, **Fig. S8**).

Measuring activity geometry

To quantify patterns of population-level neural activity characteristic of different neural implementations of TI, we calculated the following geometric measures (indices). These indices were calculated for delay period activity, and were based on the following groupings of task items: S = 7 (all items: A, B, C, D, E, F, G). $S_{high} = 3$ (high-rank items: A, B, C). $S_{low} = 3$ (low-rank items: E, F, G). Note that each item defines a trial type during the delay period; thus for delay period activity there were 7 different trial types. All indices were calculated in the top 10 PCs of delay period activity. Note that the geometric indices were measured in the full activity space of networks

(N-D ambient space, reduced to top 10 PCs) to avoid assumptions or biases incurred from an intermediate step of estimating activity subspaces (such as the subspace of an oscillation).

Ordered collinearity index (OCI). The ordered collinearity index (OCI) measures the degree to which activity is rank-ordered and linearly arranged in activity space (schematic in **Fig. 7a**; characteristic of the "subtractive" solution, **Fig. 5f**), defined as

$$OCI = -\frac{1}{S_{high}S_{low}} \sum_{i}^{S_{high}} \sum_{j}^{S_{low}} \frac{\mathbf{v_i} \bullet \mathbf{v_j}}{\|\mathbf{v_i}\|\|\mathbf{v_j}\|}$$

Activity state vectors (v) were network activity states (x) measured with respect to the cross-condition mean (XCM; yellow circle in **Fig. 7a**; see also **Fig. 5a**). Note that the negative sign results in value +1 for activity corresponding to rank-ordered collinearity. OCI was measured during two periods of time (time-averaged over the period): over the 1st quarter of the delay (early delay OCI; **Fig. 7a**, left column) and over the last quarter of the delay (late delay OCI; **Fig. 7a**, middle column). OCI change (**Fig. 7a**, right column) was defined as the late delay OCI - early delay OCI.

Mean angle. To generalize OCI change to other possible angular arrangements of neural activity, we measured mean angle change (schematic of mean angle in **Fig. 7b**), defined as

$$Index = \frac{1}{\binom{S}{2}} \sum_{i=1}^{S} \sum_{j \neq i}^{S} \left[\frac{\mathbf{v}_{i,late} \bullet \mathbf{v}_{j,late}}{\|\mathbf{v}_{i,late}\|\|\mathbf{v}_{j,late}\|} - \frac{\mathbf{v}_{i,early} \bullet \mathbf{v}_{j,early}}{\|\mathbf{v}_{i,early}\|\|\mathbf{v}_{j,early}\|} \right]$$

Activity state vectors (v) were network activity states (x) measured with respect to the cross-condition mean (XCM), and averaged over either the first (early) or last late) quarter of the delay period.

Mean distance. To help distinguish between angular vs. non-angular rearrangements of activity states in activity space (during the delay period), we measured mean distance change (schematic of mean distance in **Fig. 7b**), defined as

$$Index = \frac{1}{\binom{S}{2}} \sum_{i=1}^{S} \sum_{j \neq i}^{S} [(\mathbf{x}_{i,late} - \mathbf{x}_{j,late}) - (\mathbf{x}_{i,early} - \mathbf{x}_{j,early})]$$

Axis angles. To quantify predicted angular relationships between activity patterns (**Fig. S8**), we calculated the relevant cosine angles (dot products). In particular, simple-regime RNNs (**Figs. 5** and **6**) predict two angular relationships: (1) orthogonality (cosine angle: 0) between the XCM and the plane of the oscillation associated with transitive comparison (plane of the comparison oscillation), and (2) alignment (cosine angle: $\pm$ 1) between the choice axis and the plane of the comparison oscillation. The plane of the comparison oscillation was defined by the two eigenvectors of the comparison oscillation (see above, Linear dynamics), which were orthogonalized and unit-normalized.

For (1) (**Fig. S8b**, left column), the dot product was calculated between the XCM and each of these plane vectors. As a conservative estimate, the cosine angle was defined as the dot product having the larger manitude.

For (2) (**Fig. S8b**, right column), the dot product was calculated between the choice axis and each of these plane vectors. As a conservative estimate, the cosine angle was taken to be the dot product having the smaller magnitude.

All angles were calculated after first reducing neural activity to the top PCs (PCs calculated from delay period activity; either top 3 or 10 PCs, indicated in **Fig. S8b**).

Statistical tests

All statistical tests were non-parametric and two-sided.

Supplementary Figures

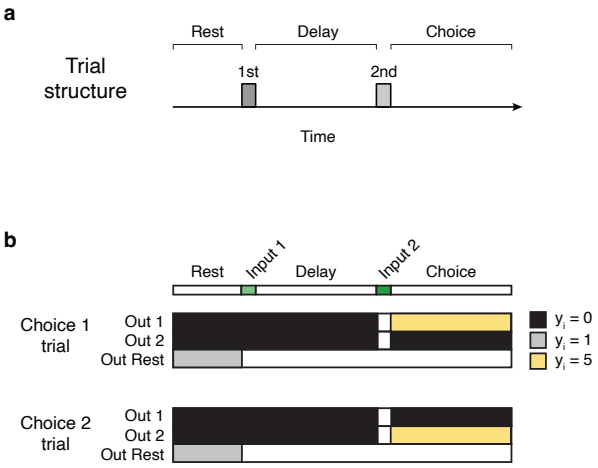

Fig. S1. Transitive inference (TI) with an explicit working-memory (WM) delay. a, Trial structure of delay-based item presentation. Each trial consists of three periods: rest, delay, and choice. Subjects respond during the choice period, following presentation of item 2. **b**, Target output of RNNs (see Methods).

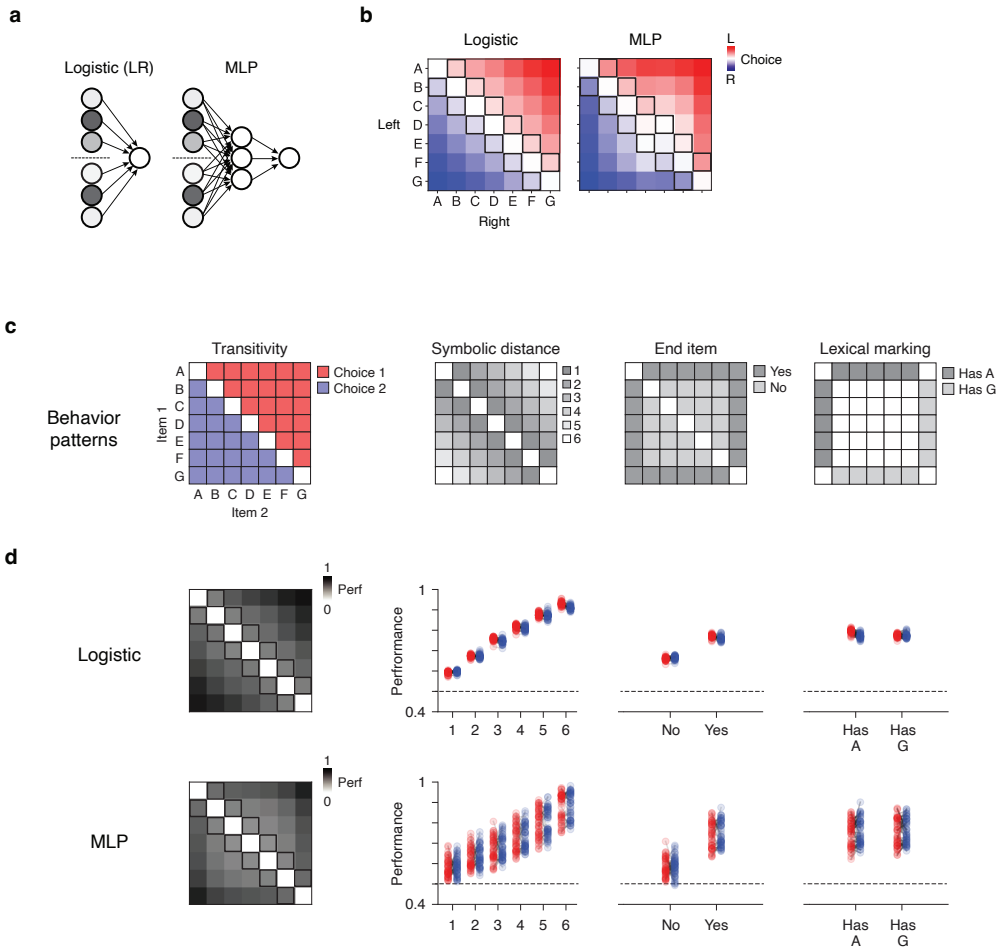

Fig. S2. Feedforward models can generalize transitivity when working memory (WM) is not required. **a**, Architecture of feedforward models. **b**, Example LR and MLP model instances that perform no-delay TI (left vs. right, Fig. 1b). **c**, Schematic of behavior patterns (see Fig. 3a). **d**, Behavioral results across feedforward models (n = 50 instances / model). All plots show performance (proportion correct). Column 1: Averages across model instances by trial type. Columns 2-5: Averages across trials (5000 simulations) for each model instance by trial type. Trial types follow that defined for each behavioral pattern in panel c (column 2: symbolic distance; column 3: end item; column 4: lexical marking), in addition to distinguishing between choice 1 vs. choice 2 trial types (red vs. blue, respectively; diagrammed in panel c, transitivity).

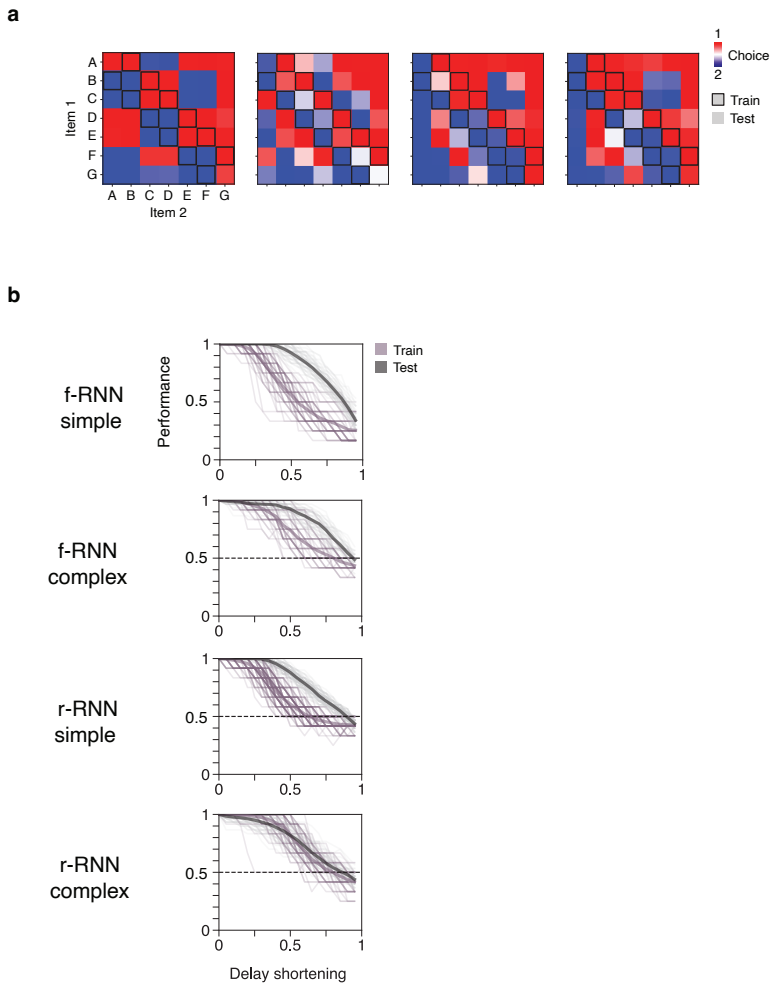

Fig. S3. RNNs trained on TI: failures to generalize and robustness to variable delay lengths. a, Four examples of RNNs that responded correctly in training trials, but failed to generalize. **b**, Performance (proportion correct) as a function of variable delay lengths. Trials were shortened relative to the (fixed) delay length used in training, and performance was measured for all trial types (separately averaged for training (purple) vs. test (grey) trial types). Averages across model instances (n=40 or 50 for each RNN variant) are shown as darker traces.

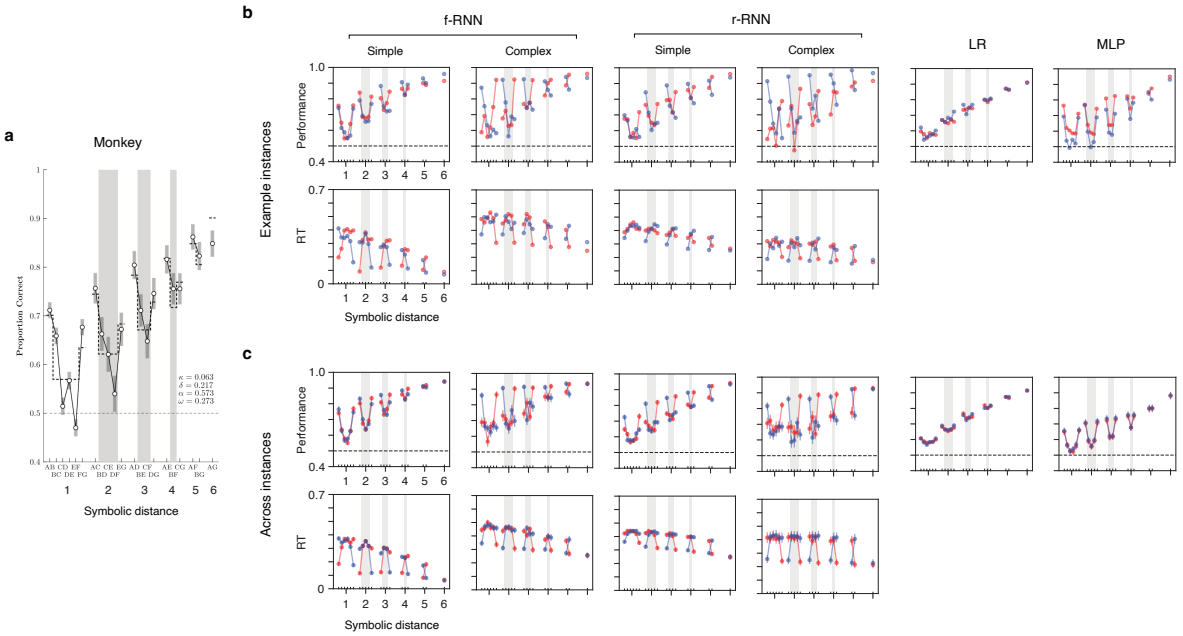

Fig. S4. RNNs show TI behavior similar to that of living subjects. a, Monkey behavioral performance by trial type (from [53]). Data are from no-delay trial format. **b**, Behavioral results from six examples across six model types (columns: f-RNN and r-RNN (simple: simple (high); complex: complex (high); see Table 1), LR, and MLP). Plotted are mean performance (top row) and mean RTs (bottom row) by trial type, following the same trial type plotting convention in panel a, with additional distinction of choice 1 vs. 2 trials (red and blue, respectively). Highlighted in each plot are critical test trial types (grey zones; test trials that do not contain end items (A or G)). **c**, Behavioral results averaged across instances for the same model types shown in panel b. Data plotted are the same as that of Fig. 3, but here explicitly show differences in individual trial types (AB, BA, AC, etc.). Error bars are ± 2 s.e.m. (n=50 for all model classes (columns) except n=40 for complex-regime r-RNN).

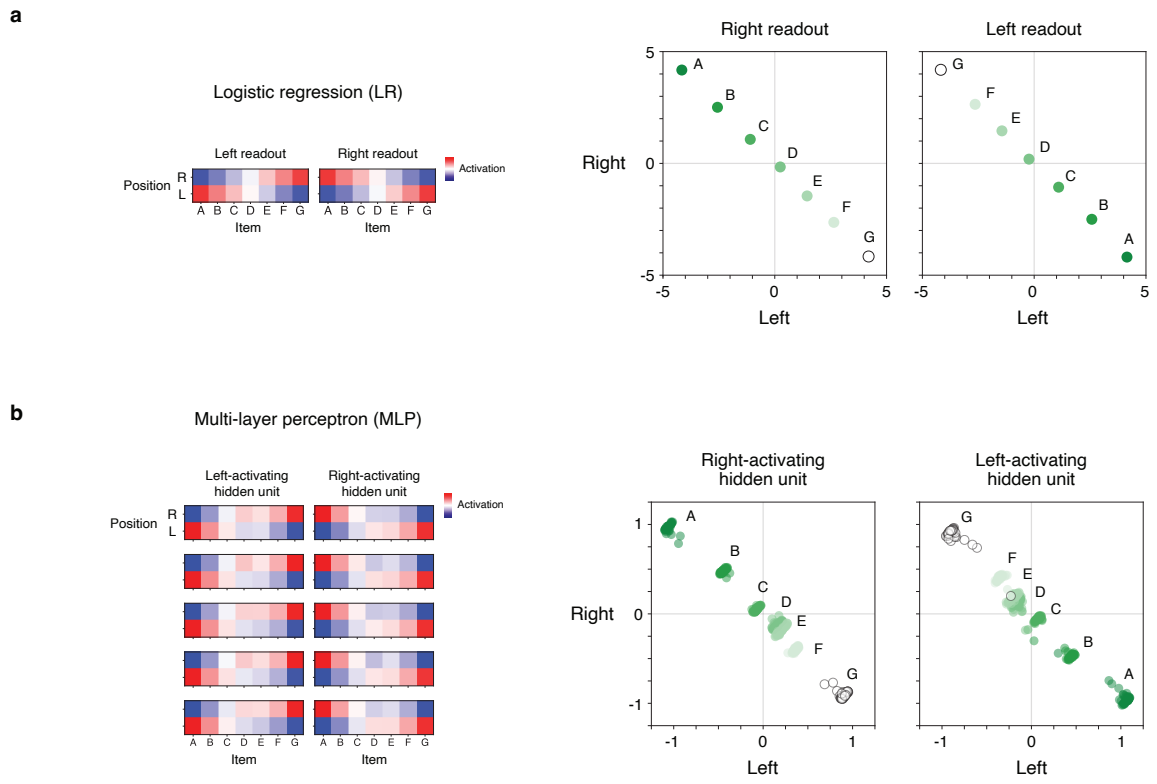

Fig. S5. Feedforward models show a 'subtractive' implementation of TI. a, Analysis of an example LR. At left, activation of readout 'units' (see Methods) as a function of input position (y-axis) and rank (x-axis). At right, relationship between position of inputs and readout unit activation. Note that activations by item position (left vs. right) were sign-inverted versions of each other. **b**, Analysis of an example MLP. At left, activation of hidden units as a function of input position (y-axis) and rank (x-axis). At right, relationship between position of inputs and unit activation, plotted for all hidden units (N=100 tanh units). Note that activations by item position (left vs. right) were approximately sign-inverted versions of each other, akin to the LR model.

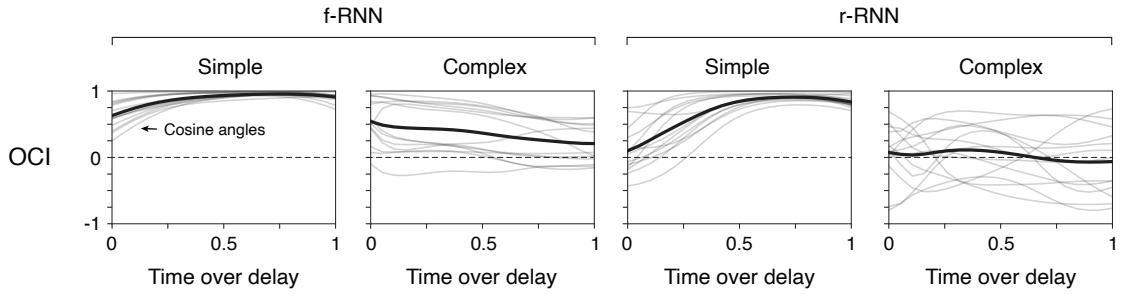

Fig. S6. Ordered collinearity in RNNs over the delay period. Ordered collinearity over the course of the delay period in four example RNNs. In each plot, two measures are plotted: the ordered collinearity index (OCI, black lines; schematized in **Fig. 7a**; see Methods for further details) and individual cosine angles between trial types (grey lines; e.g. A vs. B, B vs. C activity states, sign-adjusted based on ordered collinear arrangement). The OCI is the average across individual sign-adjusted cosine angles. Note that the OCI in simple-regime f-RNN is consistently high, while simple-regime r-RNNs begin near 0, and evolve to 1; note also the decreasing or relatively unchanged OCI in complex-regime RNNs.

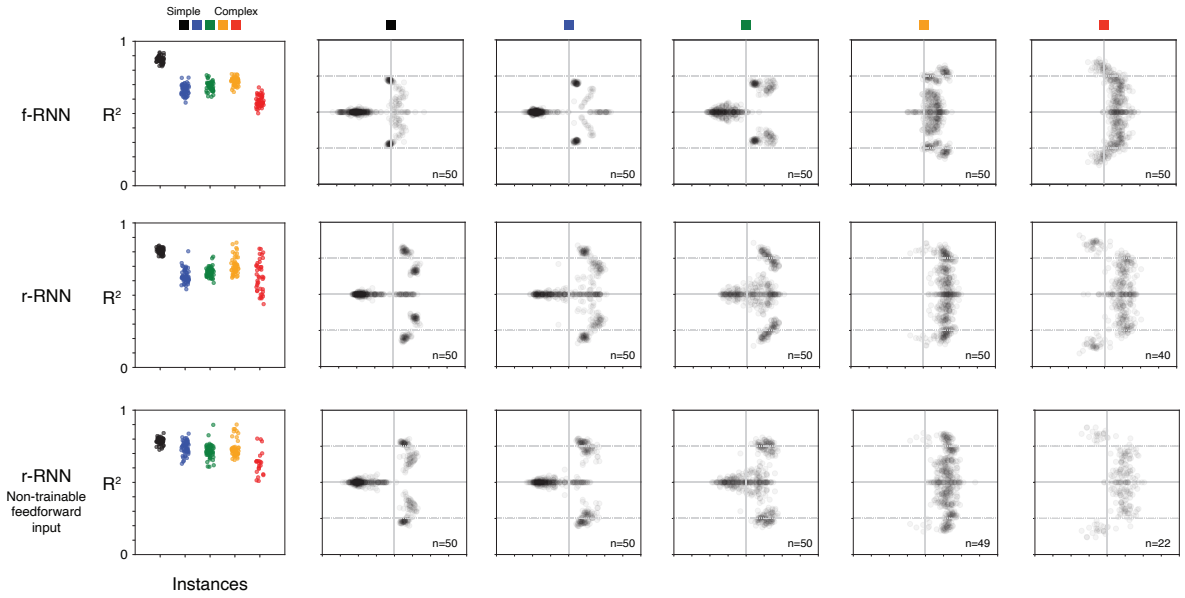

Fig. S7. Inferred linear dynamics in RNNs during the delay period. RNN activity during the delay period was fit to a linear dynamics model (least-squares). Rows show results for three RNN variants differing by learnable connectivity (f-RNN: fully-trainable RNN, r-RNN: recurrent-trainable RNN, and last row corresponds to r-RNN for which only feedforward inputs (not outputs) were not trainable). Column 1: R^2 values of the fit. Training regime variants plotted by color. Columns 2-5: eigenvalue spectra (grey points; calculated for each instance using top 10 PCs). Each column corresponds to a different training variant (simple to complex). For each variant, n=50 instances (with these exceptions: 40 for complex (high) regime r-RNNs (red), 49 for complex (low) regime r-RNNs in third row, and 22 for complex (high) regime r-RNNs in third row). Note that spectra shown in Fig. 5c corresponds to two of the spectra here combined (f-RNN simple (high) (black, second column) and simple (low) (blue, third column)), and that spectra shown in Fig. 6c is the same as that shown in the third row, second column.

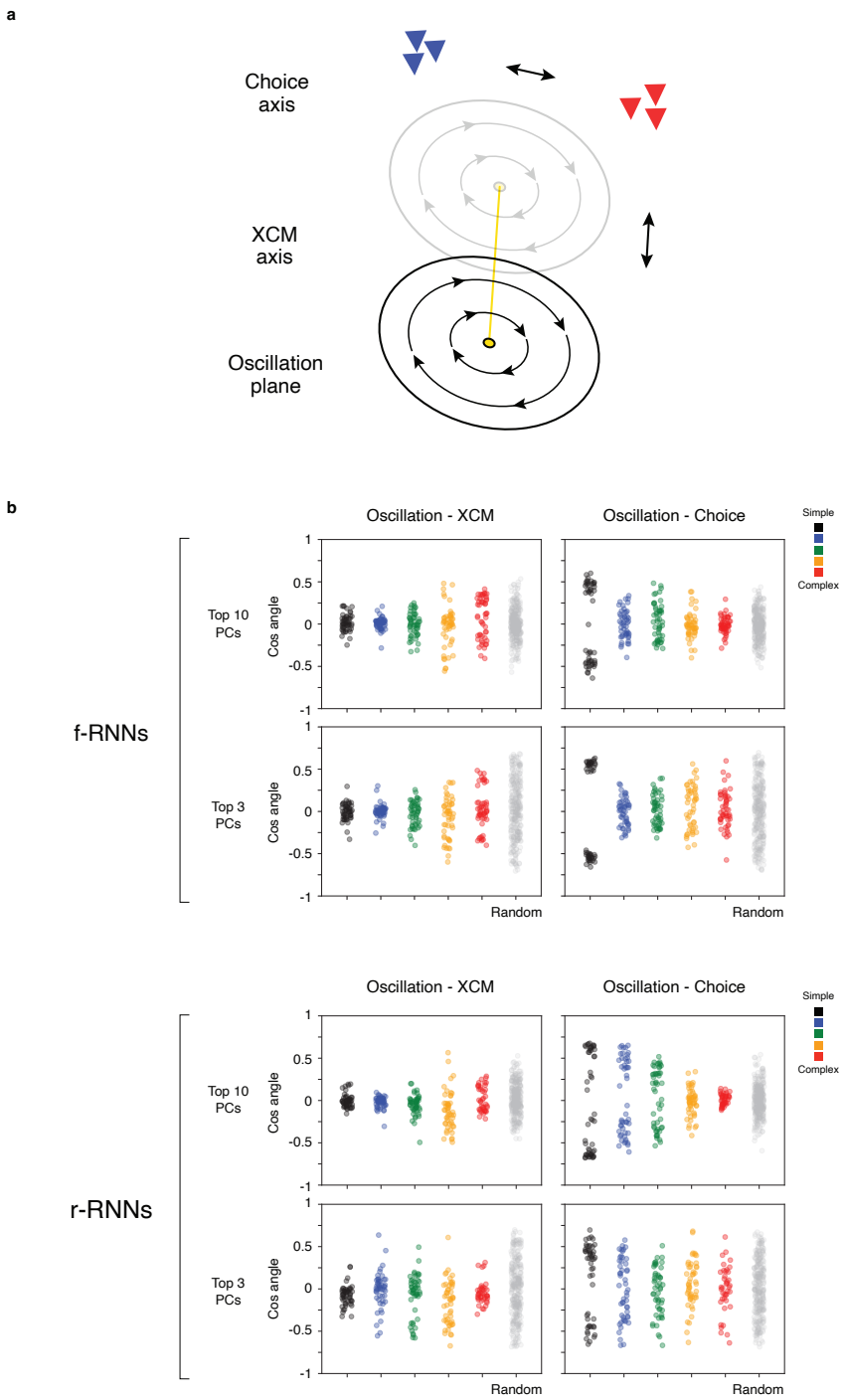

Fig. S8. Angular relationships of the oscillation associated with transitive comparison. **a**, Schematic of activity geometry in activity space (compare to **Figs. 5a** and **6a**). The oscillation plane refers to the oscillatory mode associated with transitive comparison. Note that simple-regime RNNs (**Figs. 5** and **6**) make two specific predictions: (1) the oscillation plane should be orthogonal to the XCM axis, and (2) the oscillation plane should be aligned (parallel or anti-parallel) to the choice axis. All measures were inferred from neural activity (see Methods). XCM: cross-condition mean. See Methods for further details. **b**, Quantification of activity geometry across RNNs. This quantification clarifies the predictions of simple-regime RNNs stated above. See also summary of predictions in Table 4.