# Spatiotemporal Program Learning in Human Adults, Children, and Monkeys

Tracey Mills[1], Nicole Coates[1], Alessandra A. Silva[2], Kaylee Ji[1], Stephen Ferrigno[2], Laura E. Schulz[1], Joshua B. Tenenbaum[1], and Samuel J. Cheyette[1]

[1]Department of Brain & Cognitive Sciences, Massachusetts Institute of Technology

[2]Department of Psychology, University of Wisconsin

## Abstract

People learn languages, music, games, mathematics, and a seemingly limitless assortment of other structures across domains. How do we learn such a large variety of richly structured representations efficiently? One possibility is that people "learn by programming," synthesizing data-generating algorithms to explain what they observe. We examine the nature and origins of structure learning mechanisms in human adults, human children, and nonhuman primates (rhesus macaques), using a highly unconstrained sequence prediction task. Human adults and older children (4-7 y.o.) learned many richly structured sequences, while monkeys and younger children (3 y.o.) succeeded mostly on simple, continuously-varying sequences (e.g. linear or approximately linear patterns). We test multiple learning models and find that adults and older children are best explained by an inference model that generates programs in a "Language of Thought" with motor and geometry primitives, while monkeys are best explained by local linear extrapolation strategies. Younger children exhibit variation in strategies but pattern more closely with monkeys than adults. By age 4, children show strong program-like inductive biases similar to adults and are best fit in aggregate by the LoT model.

# Introduction

A defining aspect of human cognition is its tendency to seek and impose structure on the world, even when the inputs are sparse and ambiguous. We guess the rules of new games after watching a single round, reproduce and elaborate on snippets of songs, and discern figures in clouds and stars [1]. This general capacity for inferring rich structures distinguishes humans from other learners, yet the mechanisms that enable such efficient structure learning in humans are still poorly understood. Although modern AI systems can acquire and manipulate complex symbol systems such as natural language, they are markedly less data-efficient than children [2] and often fail to generalize to new contexts in a human-like way even when explicitly trained to do so [3, 4].

A large body of empirical work has highlighted impressive statistical learning abilities in children and infants, demonstrating their sensitivity to distributional information in many domains. Infants are surprised by improbable events [5], track probabilities and co-occurrence statistics in sequences [6–9], and infer complex transitional regularities such as higher-order temporal structure in visual scenes [7], among other related capacities [10–18]. However, later in development and into adulthood, people learn structured concepts of many kinds that ostensibly go beyond what can be straightforwardly accounted for by simple statistical dependencies such as transition probabilities [19, 20]. This includes hierarchical social systems like kinship relations [21], symbolic numbers [22], musical rhythms [23], shapes and geometric figures [24–26], and written characters [27]. Even more challenging to explain, from a simple statistical learning perspective, is adults' ability to infer complex structure in *novel* domains on-the-fly, such as compositional functions [28–30] or logical concepts [31, 32], given only a handful of examples.

One possibility that may explain people's ability to learn structured concepts on-the-fly and increasingly complex symbolic systems over development is that we "learn by programming," synthesizing data-generating algorithms to explain observations [33, 34]. By treating learning as the search for compact, generative programs, this approach enables data-efficient inferences about latent structure. Program induction models have gained increasing traction as accounts of human concept learning in a number of domains, including logical rules, counting algorithms, and handwritten characters, among many others [e.g. 21, 22, 27, 31, 35–38]. While there is converging evidence that adults have "program-like" inductive biases favoring compositionality [e.g. 28], there has been little work rigorously testing program learning against other plausible learning mechanisms either in adults or throughout development.

There is some empirical work suggesting that inductive biases for hierarchical representations emerge early in human development. These biases arguably play a central role in uniquely

human intelligence [39, 40] and are a prerequisite for learning expressive programs. Infants exhibit a compositional bias in word learning that allows them to quickly learn combinatorial concepts such as quantity indicators [41], and by age 3 there is evidence that this bias extends beyond the domain of language and to more complex hierarchical structures, with children preferring center-embedded representations over equally plausible non-hierarchical representations for short visual sequences [42]. In more constrained settings, there is also evidence that children can reason about different kinds of underlying structures to explain data: 4- to 6-year-olds appeal to different abstract functional forms to explain causal relationships when matching linear, U-shaped, and cyclic functions across perceptually varying stimuli [43], and 5- to 6-year-olds can learn spatial sequences of up to 8 unique locations on the vertices of an octagon, with a bias for sequences with repeated rotations or symmetry which can be compactly written in a programming language with geometric primitives [36]. By age 9, children recognize hierarchical fractal sequences without feedback and with very little training [44].

These developmental results leave open whether the building blocks of program learning are continuous with mechanisms found in other animals or whether they represent a sharp discontinuity. Our evolutionary ancestors as distantly related as cuttlefish, bees, and birds seem to share certain common statistical and associative learning mechanisms [45, 46]. Nonhuman primates can infer and generalize abstract statistical representations such as the linear trend of a scatterplot [47] and learn transitional regularities in sequences [48]. However, apart from certain songbirds [49, 50], nonhuman animals typically show little evidence of spontaneous biases for compositional structure [24], their success with such concepts is often brittle [51], and imbuing these biases often requires extensive training [24, 39, 40, 42, 52, 53]. While monkeys can learn human-like biases for hierarchical concepts — such as a preference for center-embedded recursively structured sequences — [42] and learn simple programs like copying or reversing a short sequence depending on a cue [54], such biases are typically observed only after thousands of training examples in tightly controlled tasks. It is therefore unclear when nonhuman primates might spontaneously learn structured algorithms in less constrained settings or deploy such abilities to aid them in online inference.

To summarize, previous work has provided compelling but largely indirect evidence that inference over programs can help explain the efficiency of adult concept learning as well as the acquisition of symbol systems over development, that certain hierarchical and compositional biases emerge early in childhood, and that such biases are much weaker but can be instilled in nonhuman primates with sufficient training. It remains an open question whether humans and other primates begin with broadly similar inferential mechanisms that diverge over development as humans acquire stronger hierarchical and compositional biases or whether humans

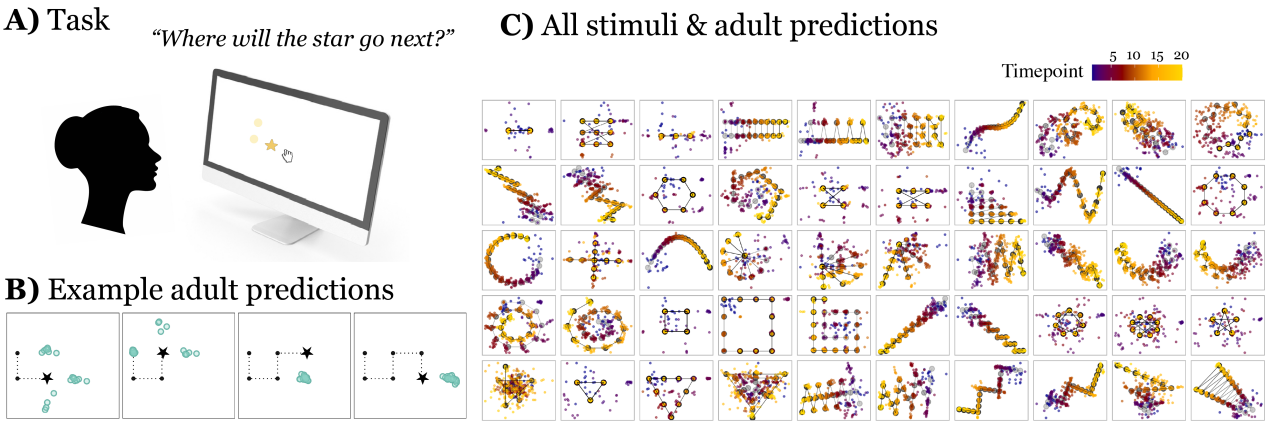


Figure 1: (A) Task as seen by children and adults. The large star is at the most recently revealed sequence location, with earlier locations indicated by smaller points. Monkeys saw an analogous display of red circles against a white background, with later circles brighter than earlier circles, and the most recently revealed circle larger than the rest. (B) An example of a sequence unfolding over from the third step (left) to the sixth step (right) and predictions made by adults. (C) All stimuli and adult predictions from Mills, Tenenbaum, & Cheyette (2023) [26]. The true patterns are shown underneath as black dots connected by gray lines. Human predictions are points, with earlier timesteps shown as cooler colors and later timepoints in warmer colors. Note that early in the sequences (cooler), predictions tend to be more dispersed and later in the sequence (warmer), predictions tend to be concentrated.

are endowed with distinct inductive capacities from the outset. To address these gaps, we introduce a paradigm that allows us to directly observe open-ended inferences about structured spatiotemporal sequences. This task offers a framework to adjudicate between competing accounts of learning — from simpler statistical learning mechanisms to inference over structured programs — and their applicability across development and species.

# A paradigm to test program learning

We present a program learning task in which participants predict the next location of a moving dot in a two-dimensional space based on its previously-observed trajectory (Fig. 1). This design builds on classic function learning paradigms [e.g. 28, 29, 36, 55–59] but extends them in several ways. First, rather than mapping one-dimensional inputs to outputs, our sequences unfold across two spatial dimensions and over time, supporting a far richer set of possible patterns as stimuli. Second, instead of focusing on continuously-varying parametric functions, the stimuli in this task include highly *structured* patterns (Fig. 1 C). Finally, where most function learning studies provide dozens or hundreds of examples, participants make predictions starting from

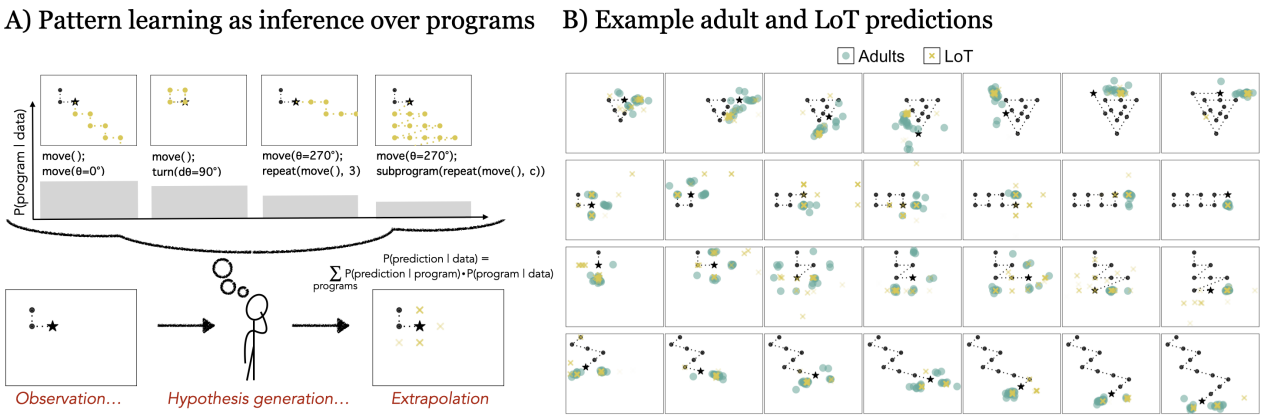


Figure 2: (A) Illustration of program learning as implemented by the LoT model. Starting at the bottom left, the learner observes the partially revealed pattern, then computes a distribution over generative programs conditioned on this observation, and finally runs the programs forward to extrapolate the pattern and predict the next point. Predictions are weighted by the posterior probability of their generative program. Programs are drawn from a grammar containing compositional functions and domain-specific motor and geometry primitives; see Methods and SI Fig. S2 for the full grammar. Programs are executed continuously to produce sequences of arbitrary length, with the execution count `c` incremented after each execution. (B) Example predictions made by human adults [26] (green) and the LoT model (yellow, transparency indicating log posterior probability) for selected patterns and timepoints. Each row shows a single pattern unfolding over successive timepoints. Both adults and the model exhibit signatures of multimodal, structured uncertainty.

the third timepoint in our task. This combination of sparse evidence and rich structure enables us to more directly probe the learning mechanisms and inductive biases that support online inference.

In previous work [26], we used this task to test human adults on the diverse set of patterns shown in Fig. 1C. Participants learned these patterns remarkably efficiently, often converging on the true underlying structure within a few datapoints. We compared adults to a set of Bayesian inference models, including several inspired by prominent accounts of human function learning [25, 28–30, 59, 60]. Each of these models performs probabilistic inference over hypotheses about the function generating the observed patterns, but vary in the degree of flexibility, compositionality, and expressiveness afforded by their hypothesis spaces. These include polynomial regressions, Gaussian Process (GP) models which select kernels and their parameters, and two models with compositional hypotheses: a Gaussian Process model which infers compositional kernel structures and their parameters [61], and an "LoT" program learning model that learns structured algorithms from general purpose operations such as repetition, concatenation, and recursion, as well as domain-specific motor and geometry primitives (Fig.

2A) — inspired by the simple Logo-like drawing language used by Sablé-Meyer et al. [25] to model human representations of geometric shapes.

Only the inference models with compositional hypothesis spaces were capable of learning the diverse set of patterns that adults learned, and adult predictions overall were best explained as inference over structured programs, as instantiated by the LoT model. This model exhibits key qualitative signatures of human learning, such as multimodal distributions of responses which indicate a bias for repeating motifs, regular angles (e.g. right-angles), and recursive structure. Illustrations of *structured uncertainty* present in both adult and LoT responses are shown in Fig. 2B.

Given the evidence that adults learn spatiotemporal patterns using Bayesian inference over expressive programs, we now aim to understand the developmental and evolutionary origins of this learning mechanism by comparing adults, children, and nonhuman primates on the exact same task. Because the task involves an intuitive prediction game that does not require natural language or complex instruction following, it is accessible to animals and children. At the same time, it allows us to test a breadth of concepts of varying complexity, such that we might distinguish between possible learning mechanisms, and task-specific or low-level strategies are unlikely to be confusable with more general abilities of interest.

This paradigm thus presents a window into how inductive biases for structure learning differ between species and across development. Here, we compare the ability of adults, preschool- and school-aged children, and rhesus macaques to learn a wide variety of structured patterns from sparse data, and evaluate a range of computational models to explain the learning mechanisms employed by each group.

# Experiment

Children (N = 110), adults (N = 20) and rhesus macaques (N = 2) each performed a sequential prediction task in which they repeatedly guessed the location of the next point in a 2D sequence. Participants watched the first three points of a sequence appear sequentially within a large rectangular region on a screen (see Fig. 1a). They then guessed where the next point would appear by tapping (children and monkeys) or clicking (adults) anywhere within the region. To prevent children and monkeys from being tempted to simply tap on the previously revealed point, guesses that fell directly on the previous point location were not registered. Otherwise, possible guess locations were only constrained by the large rectangular region. Participants received feedback indicating whether their guess was correct (based on whether it fell within an "acceptance distance," constant across sequences, of the true point location), before the

next true point appeared and they could make their next guess. The acceptance distance was selected to be large enough so that motor error did not impede success on the task, but small enough so that participants would be unlikely to be correct if they did not know the true location of the next point. Participants made 12 predictions per sequence, at timepoints 3-14. Each group was tested on 21 sequences drawn from the set shown in Fig. 1C, and children were also tested on three additional sequences selected to test for sensitivity to recursive structure (full stimulus set in SI Fig. S1).

# Results

## Humans

Consistent with our prior work [26], we found that human adults successfully learned the range of structures tested, with 79% of adult predictions counted as correct overall, and 95% correct on the final timepoint across patterns. Adult accuracy varied across patterns, from 97% correct on a line pattern (row 2, column 9 in Fig. 1C) to 45% correct on a curlicue pattern (row 1, column 9 in Fig. 1C).

Children were less accurate than adults, with 41% of predictions correct on average and 56% correct on the final timepoint. Like adults, they were most accurate on the line pattern (70%) and least accurate on the curlicue pattern (15%). Accuracy improved with age: 3-year-olds had an average accuracy of 16% overall and 24% on the final timepoint, while 7-year-olds had an average accuracy of 61% overall and 83% on the final timepoint. We fit a mixed-effects logistic regression model with fixed effects for age, prediction timepoint, and their interaction, and random intercepts and slopes for pattern types and participants to capture the fact that overall accuracy and learning speed might vary across participants and stimuli. Accuracy increased with prediction timepoint ($\beta = 0.49$, $p < .001$) and age ($\beta = 0.97$, $p < .001$), with a positive interaction ($\beta = 0.28$, $p < .001$) indicating that older children tended to both be more accurate and learn more quickly. There was a strong relationship between the relative accuracies of adults and 4-, 5-, 6-, and 7-year-olds across the different patterns (Fig. 3), suggesting that they found the same patterns relatively easy or difficult ($r = 0.77$, $r = 0.85$, $r = 0.93$, and $r = 0.93$, respectively, $p$s $< .001$). However, this relationship was not significant between adults and 3-year-olds ($r = 0.14$, $p = 0.54$).

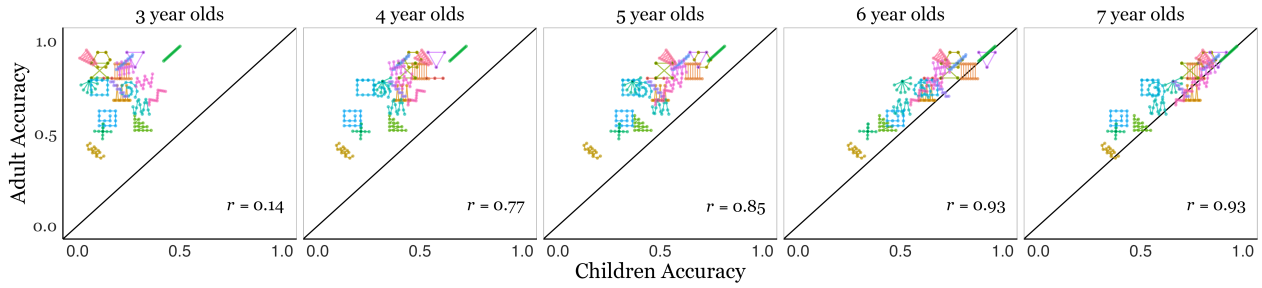


Figure 3: Correlations between children's (x-axis) and adults' (y-axis) performance on different patterns (points shown as icons of the stimulus). Children are faceted by age group, youngest (3-year-olds) to oldest (7-year-olds).

### Bayesian data analysis

One possibility is that developmental differences in performance might be attributed to differences in task difficulties, such as motor control, rather than differences in an ability to draw inferences about latent structure. To better understand the extent to which participants learned the underlying sequence-generating algorithms, rather than relying on alternative strategies, we fit a learning model to each group. The model accounts for three alternative prediction strategies that individuals might use: guessing somewhere near the next true point in the sequence, guessing somewhere around the previous point in the sequence, or guessing a random location on the screen.

We analyzed learning for participants in each group by computing posterior estimates for the probability of guessing around the true next point in the function, $\theta_{\text{true}}$, while estimating motor error. Specifically, we computed the probability of a participant guessing around the true next point in the sequence relative to these other two strategies in a logistic setup, such that

$$\theta_{\text{true}} = \text{logit}^{-1}\left(\alpha + \alpha_s + \beta_s \cdot t\right) \tag{1}$$

where $\alpha$ is an overall intercept, $\alpha_s$ is a sequence-specific contribution, and $\beta_s$ is the effect of timepoint $t$ in sequence $s$. These parameters are drawn from Normal(0,3). The likelihood of a given observed prediction $\hat{x}_t$, $\hat{y}_t$ is then given by

$$\hat{x}_t \sim \theta_{\text{true}} \cdot \mathcal{N}(x_t, \sigma_m^2) + \theta_P \cdot \mathcal{N}(x_{t-1}, \sigma_P^2) + \theta_R \cdot U(0, x_{\max})$$

$$\hat{y}_t \sim \theta_{\text{true}} \cdot \mathcal{N}(y_t, \sigma_m^2) + \theta_P \cdot \mathcal{N}(y_{t-1}, \sigma_P^2) + \theta_R \cdot U(0, y_{\max}),$$

where parameters $\theta_P$ and $\theta_R$ represent the probability of guessing around the previous point and guessing somewhere randomly on the screen, respectively; and $\sigma_m$ and $\sigma_P$ capture motor

noise when guessing around the true point and noise when guessing around the previous point respectively (see SI Methods for details).

Though estimated $\theta_{\text{true}}$ is related to accuracy, it can account for additional nuances in the data, such as predictions that might be counted as correct but are better explained as random draws from around the previous point, and allows groups with different motor error to be more fairly compared. To the extent that a participant's predictions are biased towards the next true point in a sequence on average, despite random noise on any one prediction, their estimated $\theta_{\text{true}}$ will be higher. To the extent that their predictions are better described as being distributed randomly around the previously revealed point, or uniformly at random within the screen, their estimated $\theta_{\text{true}}$ will be lower. We find that the observed developmental trend does not change qualitatively when considering $\theta_{\text{true}}$ rather than accuracy.

The posterior mean estimate of $\theta_{\text{true}}$, averaged across all patterns, was 0.77 for adults (0.94 at the final timepoint), 0.43 for children overall (0.56 at the final timepoint), and varied considerably between 3-year-olds (0.18 overall, 0.20 final) and 7-year-olds (0.63 overall, 0.85 final). As with accuracy, there was a strong positive relationship between relative estimates of $\theta_{\text{true}}$ between adults and 4, 5, 6, and 7-year-olds across the different patterns ($r = 0.81$, $r = 0.90$, $r = 0.92$, and $r = 0.94$, respectively, $p < .001$), but not with 3-year-olds ($r = 0.39$, $p = .082$).

## Rhesus Macaques

### Training

Before seeing the 21 test sequences used in the adult and children's experiments, the monkeys were trained to perform the sequence prediction task on a broad set of 20 different types of training sequences (see Methods and Fig. 4). Both monkeys demonstrated learning across the 20 different pattern types during training, with overall accuracy rates of 47% for Monkey 1 and 44% for Monkey 2 over 77,000 total trials. The correlation between the two monkeys' accuracy on training patterns was 0.86, indicating highly consistent performance (Fig. 4C). Pattern difficulty varied substantially, with linear and polynomial sequences having the highest accuracy rates of 73-81% for both monkeys. Both monkeys substantially improved overall throughout training, with positive trends on nearly all pattern types when comparing early training (first 25% of trials) to late training (last 25% of trials). Monkey 1 improved on all 20 pattern types with an average improvement of 23%, while Monkey 2 improved on 19 of 20 pattern types with an average improvement of 21%. But both monkeys struggled throughout with less continuous and more algorithmic patterns like radials, zigzags, polygons ($<$ 30%

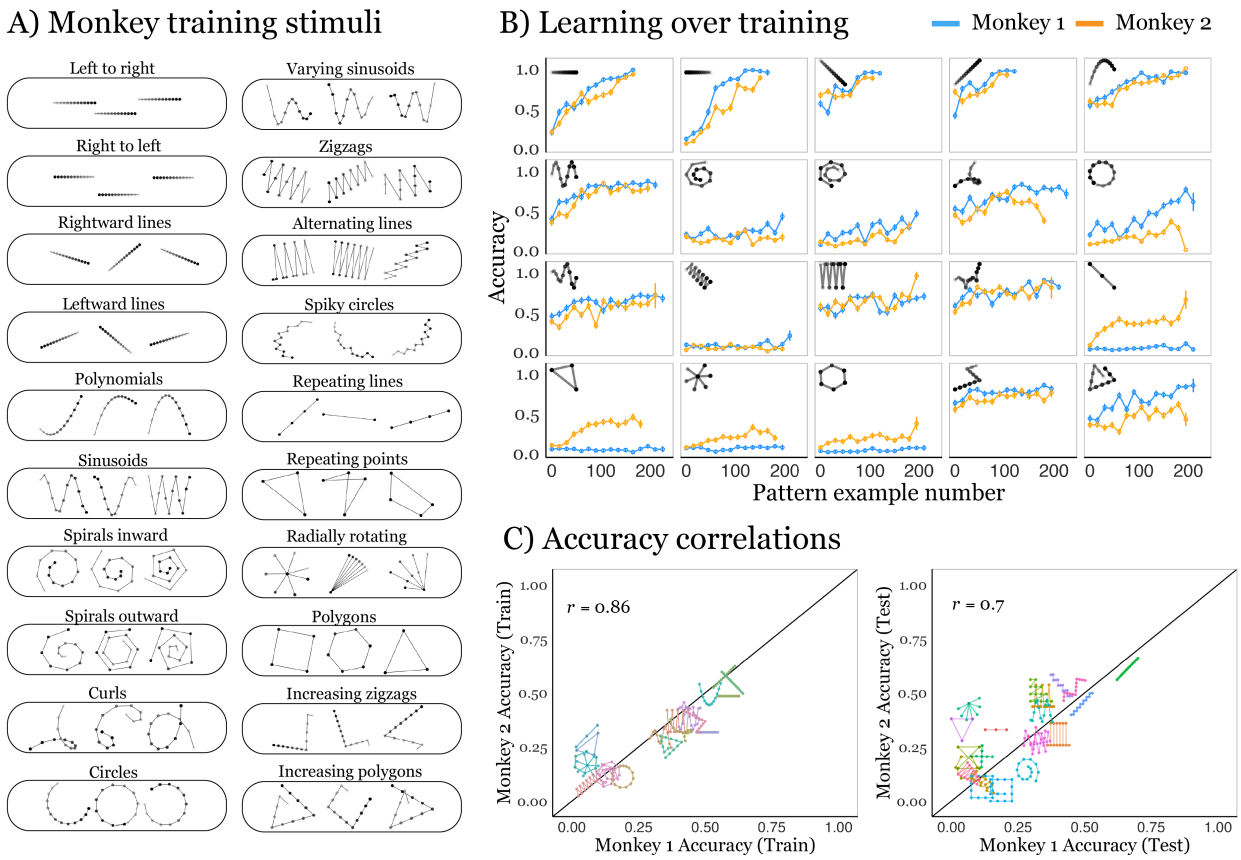


Figure 4: (A) The types of sequences used to train monkeys on the task, with examples of each stimulus type. Participants saw sequences in the order shown here (descending columns), i.e. "left-to-right" was the first pattern type shown and "increasing polygons" were the last. (B) Monkeys' accuracy (y-axis) as a function of exposure (x-axis) for each pattern type (facets); Monkey 1 is shown in blue and Monkey 2 is shown in orange. (C) Correlations between the monkeys' accuracy on different training pattern types (left) and on the test stimuli (right). Note that the test stimuli are those shown to human participants.

accuracy). Monkey 1 consistently outperformed Monkey 2 on most pattern types involving parametric functions, particularly on circular patterns (40% vs 17%) and spiral patterns; and Monkey 2 outperformed Monkey 1 on patterns involving the repetition of past points, including repeating lines, repeating points, and polygons.

To better characterize monkeys' learning trajectories across different pattern types, we fit a Bayesian mixture model to their responses throughout training analogous to Eq. 1. The model assumes that on each trial, monkeys either predict near the true next point with probability $\theta_s$, guess around the previous point, or guess randomly. The probability $\theta_s$ of responding accurately to pattern type $s$ increases with experience according to $\theta_s = \text{logit}(\alpha_s + \beta_s \cdot t_s) \cdot L_s$, where $t_s$ is the

number of examples seen, $\alpha_s$ and $\beta_s$ capture initial performance and learning rate respectively, and $L_s$ represents the asymptotic performance limit. Both monkeys showed positive learning rates ($\beta_s > 0$) for all pattern types, but final inferred accuracy $\theta_s$ varied dramatically across patterns, ranging from near zero (e.g., Monkey 1 on repeating points) to near perfect (both monkeys on left-to-right sequences; SI Fig. S3).

The primary determinant of monkeys' inferred accuracy at the end of training was how amenable each pattern was to linear extrapolation. We computed the proportion of points for each pattern type where a pure linear extrapolation strategy would succeed. The correlation between that measure — i.e., the (approximate) linearity of each pattern type — and monkeys' final inferred accuracy was $r = 0.89$ for Monkey 1 and $r = 0.77$ for Monkey 2. This indicates that despite exposure to diverse non-linear patterns — and even after the removal of all linear patterns from their training set — both monkeys continued to rely heavily on linear prediction strategies. However, both monkeys did outperform linear extrapolation on specific pattern types: Monkey 1 showed substantial advantages on non-linear but continuously-varying patterns like circles (88% vs. 32% expected from linear extrapolation) and spirals, while Monkey 2 significantly outperformed linear extrapolation on repetitive patterns (See SI Results). This indicates that while both monkeys largely shared a strategy of linear extrapolation, they deviated from it in idiosyncratic ways.

**Test**

The two monkeys had similar overall accuracy on the test patterns. Monkey 1 had an overall accuracy of 22%, and 34% on the final timepoint, while Monkey 2 had an overall accuracy of 26%, and 35% on the final timepoint. Accuracy varied substantially by pattern type. Both monkeys were correct on over half of their predictions on the line pattern, with 62% accuracy for Monkey 1 and 56% for Monkey 2. As in training, both monkeys' performance dropped sharply on highly non-linear patterns, with Monkey 1 performing worst on the triangle pattern (row 5, column 2 in Fig. 1C, 0% accuracy) and Monkey 2 performing worst on the square spiral pattern (row 4, column 5 in Fig. 1C, 0% accuracy).

There was a strong positive relationship between the relative accuracies of the two monkeys on the test patterns (Fig. 4C; $r = 0.70$, $p < .001$). The most notable difference between the monkeys was their performance on patterns involving the repetition of points. While Monkey 1 had near-zero accuracy on these patterns, Monkey 2 exhibited some learning across both train patterns (including repeating lines, repeating points, radials, and polygons) and test patterns (including the 3-point, hourglass, radial, and triangle patterns, shown in row 1, column 3; row

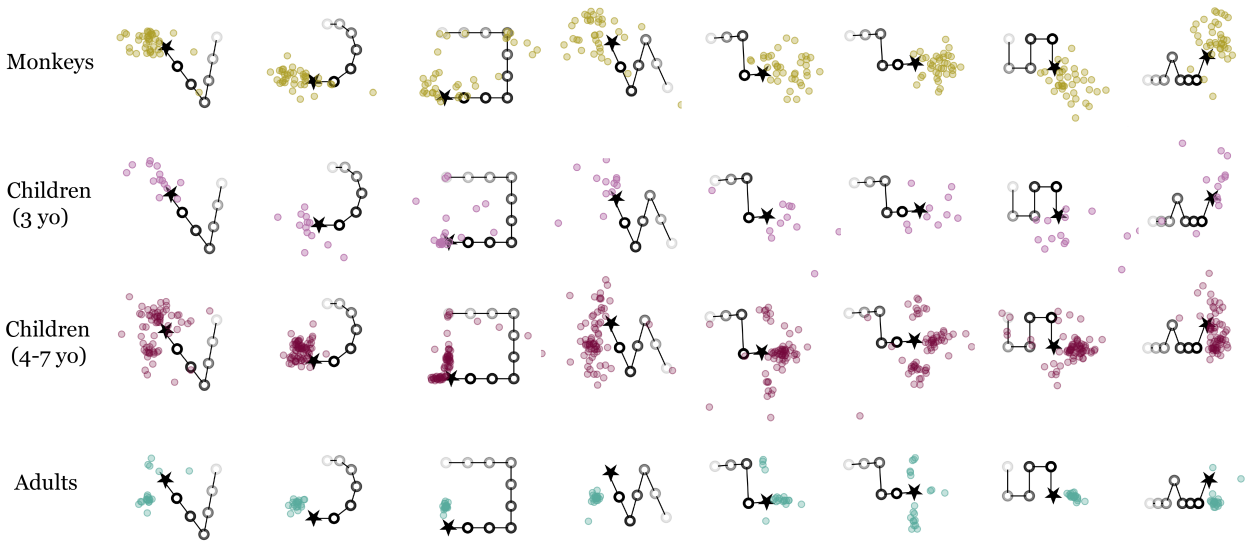


Figure 5: Predictions for how representative spatiotemporal patterns will continue, at a selected set of timepoints, generated by participants in four different population: monkeys, 3 year-old children, 4-7 year-old children, and adults. For adults and children, each dot represents the prediction of one participant. For the monkeys, dots aggregate the predictions of two monkeys across a number of trials on multiple test runs. Older children and adults show more structured and often multimodal predictions, whereas 3-year-olds' and monkeys' predictions tend to track the locally linear trend.

2, column 5; row 3, column 4; and row 5, column 2 of Fig. 1C respectively). When considering modeled $\theta_{\text{true}}$ as computed with Eq. 1, rather than raw accuracy, the relationship between the monkeys' relative performance across patterns remained strong ($r = 0.54$, $p = .011$), though estimated $\theta_{\text{true}}$ was overall higher for Monkey 1 (0.40 across all predictions, 0.42 on the final timepoint) than Monkey 2 (0.24 across all predictions, 0.27 on the final timepoint).

## Comparing humans and monkeys

Having tested human adults, children, and monkeys on the exact same patterns, we can now assess similarities and differences in performance between groups (Fig. 5). Even after accounting for individual differences in motor error, overall performance was strongest in adults (accuracy= 79%, $\theta_{\text{true}} = 0.77$), followed by children (accuracy= 41%, $\theta_{\text{true}} = 0.43$), then monkeys (accuracy= 24%, $\theta_{\text{true}} = 0.32$). Although there was a strong relationship between the relative accuracy of the two monkeys across the different test patterns, there was no evidence of such a relationship between monkeys and adults ($r = 0.17$, $p = .464$; rightmost facet of Fig. 6B). Between monkeys and children, this relationship was strongest with 3-year-olds ($r = 0.79$, $p < .001$), weaker with 4- and 5-year-olds ($r = 0.48$, $p = .028$ and $r = 0.44$, $p = .048$, respec-

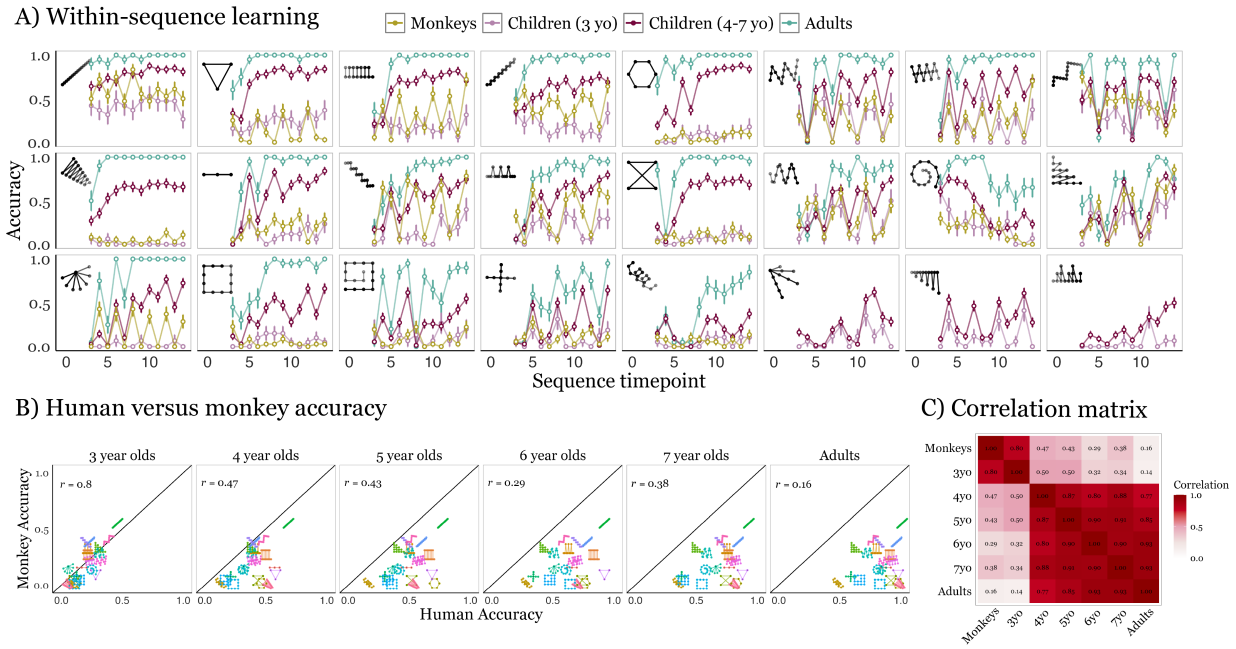


Figure 6: (A) Mean accuracy (y-axis) for each timepoint (x-axis) across all test sequences (facets). Adults are shown in green, younger children (3 yo) in light red, older children (4-7 yo) in dark red, and monkeys in gold. Ground truth sequences are shown as insets in each facet, as well as in SI Fig. S1. The three rightmost sequences in the bottom row were selected to test for children's sensitivity to recursive concepts, and were seen only by children. (B) Average accuracy across different pattern types for humans (x-axis) and monkeys (y-axis), split by age (youngest to oldest in facets). (C) A correlation heatmap for accuracy across different patterns between each population (monkeys, 3-7 yo, and adults).

tively), and not significant with 6- and 7-year-olds ($r = 0.29$, $p = 0.199$, and $r = 0.39$, $p = .081$ respectively); see Fig. 6B and C.

Similarly, when considering $\theta_{\text{true}}$ rather than raw accuracy, there was a positive relationship in relative performance across sequences between monkeys and 3-year-olds ($r = 0.46$, $p = .034$), but this relationship was not significant with older children or adults (SI Results). In sum, while 4- to 7-year-olds tended to find the same sequences as adults relatively easier or more difficult (accuracy: $0.77 \leq r \leq 0.93$; $\theta_{\text{true}}$: $0.81 \leq r \leq 0.94$;), 3-year-olds instead patterned closely with monkeys (Fig. 6C).

As shown in Fig. 6A, groups varied in the speed and extent to which accuracy increased over time within a given sequence. Using a logistic regression predicting accuracy based on sequence timepoint with random slopes and intercepts for individual participants and sequences (details in SI Results), we found significant effects of within-sequence learning that increase in strength with age starting at age 4 ($\beta = 0.37, 0.52, 0.78, 0.98,$ and $1.96$ for 4- through 7-year olds and

adults, respectively). However, there was no evidence of within-sequence learning in 3-year-olds ($\beta = 0.099$, $p = .34$) or in monkeys ($\beta = 0.074$, $p = .31$). We also analyzed how prediction accuracy varied with the next point's deviation from the local linear trajectory defined by the two most recent points. This analysis (details in SI Results) showed that monkeys and 3-year-olds relied heavily on linear extrapolation—accuracy fell sharply as the true path departed from a straight line—whereas older children and adults showed little overall linear bias and increasingly diverged from linear predictions as sequences unfolded, consistent with learning non-linear structure.

# Computational models

Our results highlight that while each group learned to do the task to some extent, they exhibited nuanced variations in performance including in overall accuracy, relative performance across sequences, and degree of learning within sequences. To explain the observed similarities and differences between groups, we evaluate a range of computational models that vary in their degree of representational flexibility and structure, and formalize alternative theories of how one might approach our open-ended structure learning task.

Each model performs Bayesian inference over a particular hypothesis space, and infers a distribution over possible underlying structures at each timepoint in a given sequence. Two of the models are drawn from classic work on human function learning [29, 30, 59, 60] and assume a fixed functional form or a structure chosen from a limited set: (i) a Bayesian polynomial-regression model and (ii) a non-compositional Gaussian Process (GP) model that infers a single kernel and its parameters. We also test two structure learning models: a compositional GP model that jointly infers both kernel structure and parameters [28, 61], and the "language-of-thought" (LoT) program learning model that best explained human adults' predictions in past work [26]. Finally, we include two models that linearly extrapolate based on the local moving average of a sequence. The linear model fits the angle and speed of a single repeated movement based on the three previous points in the sequence, essentially fitting a line through these points, and the linear + previous point model captures linear as well as simple periodic structure. It assumes that the next point will *either* be a linear extrapolation of the three previous points, or a previous location in the sequence. It fits a parameter, $p_{\text{periodic}}$, that determines the relative weights of periodic and linear predictions. Figure S7 shows examples of hypotheses learned by the Linear, GP, and LoT models while learning three different sequences.

Each model was implemented in Gen.jl [62] using Sequential Monte Carlo (SMC) with Markov Chain Monte Carlo (MCMC) rejuvenation steps. Specifically, all models run SMC

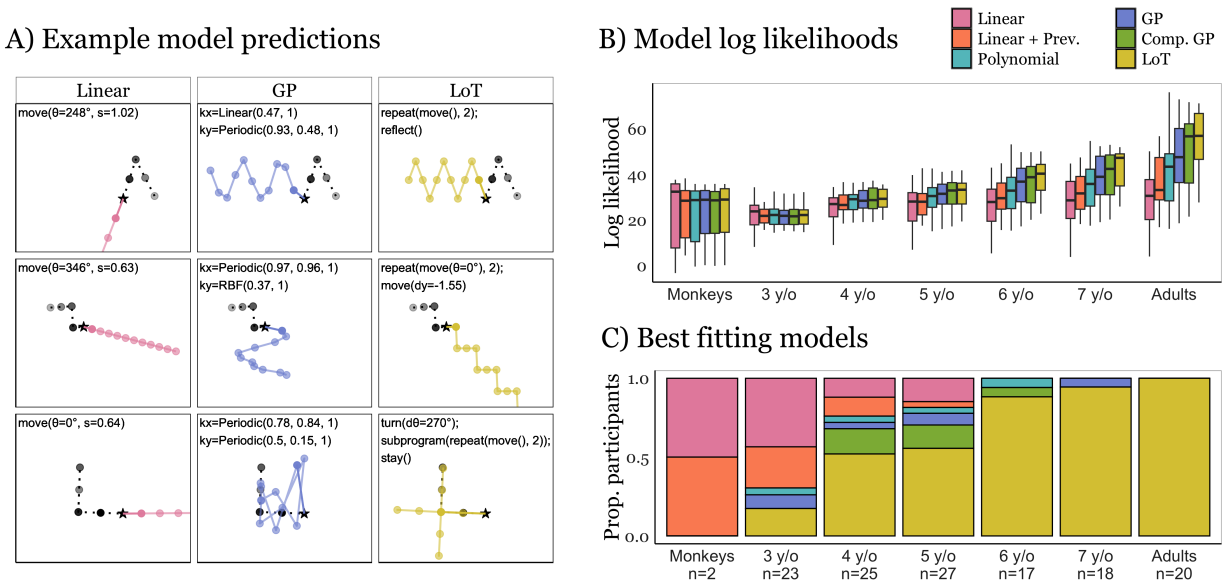


Figure 7: (A) Example hypotheses learned by the Linear, GP (Gaussian Process), and LoT models for selected patterns and timepoints. (B) Distribution of log likehoods (y-axis) of data by sequence in each group (x-axis), under each of the learning models. Log likelihoods are averaged across participants (and trials, for monkeys) within each sequence. (C) Proportion of participants best fit by each of the learning models (y-axis) by group (x-axis). A participant is best fit by a model if the likelihood of their data is highest under that model.

with 20 particles. At each timepoint, there are 100,000 rejuvenation steps on the inferred hypothesis (including noise parameters) for each particle. The rejuvenation steps use a generalization of Metropolis Hastings called Involutive MCMC that allows for custom-built kernels in the reversible jump MCMC framework. After rejuvenation, each resulting particle specifies a predicted location and noise estimate over this prediction for the next point in the sequence. Model predictions at each timepoint are marginalized over particle predictions, weighted by their posterior probability.

# Model-based results

We fit the predictions of adults, children, and monkeys under each learning model, including parameters to capture additional noise in the data: motor noise and variability stemming from perceptual imprecision, and inattentiveness or lack of effort on some trials. Specifically, we assumed participant responses were drawn from a model of the form,

$$\hat{x}_t \sim \theta_A \cdot \mathcal{N}\left(\mu_{x,\,t},\ \sigma^2_{x,\,t} + \sigma^2_m\right) + \theta_P \cdot \mathcal{N}\left(x_{t-1},\ \sigma^2_P\right) + \theta_R \cdot U(0, x_{\max}),$$

$$\hat{y}_t \sim \theta_A \cdot \mathcal{N}\left(\mu_{y,\,t},\ \sigma^2_{y,\,t} + \sigma^2_m\right) + \theta_P \cdot \mathcal{N}\left(y_{t-1},\ \sigma^2_P\right) + \theta_R \cdot U(0, y_{\max}),$$

where $\theta_A$ is the inferred accuracy controlling for lapses and motor noise (i.e. the proportion of trials on which participants are assumed to guess somewhere around the model's predicted points). $\theta_P$ and $\theta_R$ represent the proportion of trials on which participants guess somewhere around the previous, most recently revealed point and somewhere randomly on the screen, respectively. $\sigma_m$ is motor noise when guessing around the model's predicted next point and $\sigma_P$ is the noise when guessing around the previous point. $\mu_{x,\,t}$ and $\mu_{y,\,t}$ are the models' predictions for the $x$ and $y$ coordinates at time $t$, and $\sigma_{x,\,t}$ and $\sigma_{y,\,t}$ are the models' predicted standard deviations.

To estimate the extent to which the different learning models can explain the predictions of each group, we compare the likelihood of participants' predictions under each model. Fig. S7B shows the distribution of log likelihoods of participant data by sequence under each model, where the log likelihood for a particular sequence is computed as the average log likelihood across participants (and trials, for monkeys) who completed that sequence. Bootstrapped means and 95% CIs for these likelihood distributions are reported in SI Tables S1, S2 & S3. For adults and 4- through 7-year-olds, the model with the highest mean log likelihood was the LoT model. In contrast, and consistent with our empirical results, both monkeys and 3-year-olds were best fit on average by the local linear + previous point model.

Groups also varied in their best-fitting lapse and motor noise parameters. Under the best-fitting model for each participant, adults had an average inferred $\theta_P = 0.22$, $\theta_R = 0.00$ and $\sigma_m = 0.01$ (with a screen width of 1). Children in each age group tended to be noisier than adults, and under the best-fitting model for each participant had average inferred $\theta_P$ ranging from 0.30 (7-year-olds) to 0.43 (4-year-olds) , $\theta_R$ ranging from 0.03 (7-year-olds) to 0.29 (3-year-olds), and $\sigma_m$ ranging from 0.02 (7-year-olds) to 0.04 (3-year-olds); monkeys had average inferred $\theta_P = 0.28$, $\theta_R = 0.11$ and $\sigma_m = 0.03$ (details in SI Table S4).

Individuals within groups exhibited some variation in their learning biases (Fig. S7C). Every adult was best fit by the LoT model, suggesting that a highly structured, yet expressive, program learning model best explains patterns in adult learning biases and uncertainty. The majority of 4-through 7-year-olds were best fit by the LoT model as well, with 13/25 4-year-olds, 15/27 5-year-olds, 15/17 6-year-olds, and 17/18 7-year-olds best fit by this model. Thus while adult-like learning biases were reliably present in children as young as age 4, the uniformity of clear learning biases increased with age. A significant portion of 4- and 5-year-olds were best fit by other models, with at least one 4- and 5-year-old best fit by each of the five alternative

learning models.

While 4/23 3-year-olds were also best fit by the LoT model, the majority were best fit by either the local linear model or the local linear + previous point model (10/23 and 6/23 children respectively). Similarly, each monkey was best fit by one of the local linear models: Monkey 1 by the local linear model, and Monkey 2 by the local linear + previous point model. This is consistent with our empirical findings that while both monkeys perform best when sequences can be accurately predicted with local linear extrapolation, Monkey 2 also exhibits some learning of patterns that involve the repetition of points.

## Modeling monkeys with a generic sequence learning model

To establish a benchmark for what could be learned from the monkey training stimuli using minimal inductive biases, we implemented a generative pretrained transformer (GPT) model [63, 64] trained on identical trajectory data presented to the monkeys. This model uses the same basic attention-based sequential neural network architecture that underlies recent large neural language models such as ChatGPT that learn the structure of language only from observing sequences of text tokens, and allowed us to test whether the monkeys learned as much as they could in their training regime with similarly weak inductive biases. We also varied the model's context length from 1 to 13 previous positions to probe whether any suboptimal monkey performance could be attributed to limited capacity for tracking long-range temporal dependencies in the spatial sequences (i.e., memory limits). If monkeys are constrained by working memory or attention limitations that prevent them from integrating information across extended temporal windows, we would expect their performance to be best captured by transformer models with shorter context lengths.

The unlimited-context transformer model, which was trained on the same sequences as monkeys, outperformed both monkeys on all sequence types across the training set (54% difference in mean accuracy overall). The differences in absolute error and accuracy were smallest for approximately linear sequences (lines, polynomials) and greatest for non-linear and more algorithmic patterns (polygons, repeating points, zigzags). Performance on the test sequences — which the transformer was not trained on and so assess its ability to generalize — showed the same general trend with an accuracy difference of 34% overall.

One obvious possibility is that the performance differences are simply due to motor error and attentional lapses. However, an alternative possibility is that monkeys did not learn long-range dependencies needed to succeed in non-local, non-linear patterns. To test both possibilities jointly, we fit motor error and lapse terms (the same fitting procedure as the LoT, GP, and other

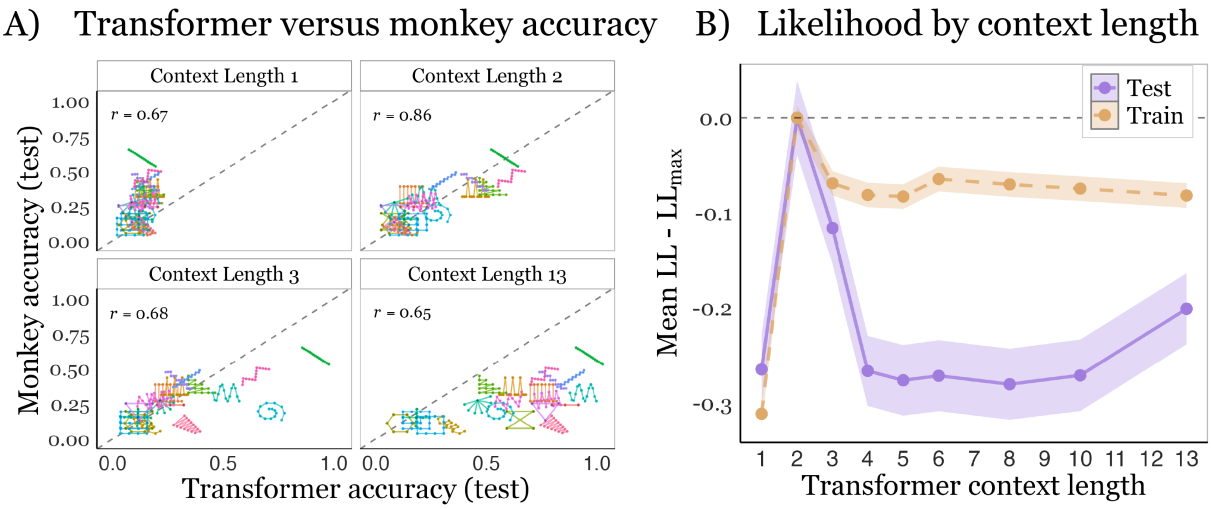


Figure 8: (A) Scatterplots showing the relationship between the transformer model's accuracy (x-axis) and monkeys' accuracy (y-axis) on the test sequences for context lengths 1, 2, 3, and 13 (full context). (B) The mean log likelihood relative to best-fitting (y-axis) of monkeys' predictions on the training (gold) and test (purple) sequences as a function of context length (x-axis).

models) to the transformer trained with each context window. Shortening the context length substantially improved the fit, indicating that motor error and inattention alone were insufficient to explain the performance gap between monkeys and the unlimited-context transformer model. In terms of overall performance, average likelihood, and correlation, the transformer trained with a context window of 2 was the best fit to the monkeys for both the training and test sequences (see Fig. 8). Notably, this model had very high correlations on mean accuracy for different pattern types of $r = 0.87$ and $r = 0.86$ for the training and test sequences respectively, and both the correlations and likelihood substantially diminished beyond context length 3. These results indicate that the patterns of monkey performance can be understood as statistical learning of only short-range dependencies (2-3 previous points) in the training sequences, which precluded learning global or highly non-linear structure. Consistent with this, the context-length 2 transformer provides the best fit to Monkey 1 relative to the other learning models we tested; almost half of 3-year-olds and a small proportion of older children are also best fit by limited-context transformer models (SI Fig. S7).

# Discussion

In this paper, we compared the performance of human adults, human preschool- and school-aged children, and two rhesus macaques on an unconstrained pattern prediction task. Both adults'

and older children's (4- through 7-year-olds) learning is overall best described as probabilistic inference over structured programs; performance differences are largely due to differences in noisiness (e.g. motor error or inattention). Statistical learning mechanisms, such as polynomial or Gaussian Process regression, fail to explain the breadth of patterns learned or capture biases for repetition, regularity, or compositionality observed in these groups. In contrast, younger children (primarily 3-year-olds) and the monkeys are best explained by simpler statistical learning models, primarily extrapolating patterns based on the moving average of previous points.

Successfully learning the diverse set of patterns in this task after observing only the first few points requires performing a series of computationally expensive tasks. Participants must first identify a relevant hypothesis space that admits the broad range of sequences they might see [19], such as programs constructed from repetition, concatenation, embedding, and motion primitives [25]. They must then search over the combinatorial space of such programs to generate candidate hypotheses and test them against their observations. Finally, they must translate their beliefs into an action plan to predict the next point in the sequence. Success therefore hinges on inductive biases favoring compositional structure, efficient hypothesis search and evaluation, and motor planning. The computational demands of this task make it all the more remarkable that many children at age four exhibit adult-like program learning abilities on this task from their very first try.

The fact that 3-year-olds predominantly used linear extrapolation admits at least two broad interpretations. Young children may possess program-induction capacities but fail to reveal them here because of limited executive control [65] or difficulties coordinating motor actions with abstract predictions. Between ages three and four, children typically acquire richer experience with tasks like tracing shapes and connecting dots which could enhance both attention to global geometric structure and motor planning [66]. Alternatively, 3-year-olds may not yet be able to either represent or perform inference over a sufficiently expressive "Language of Thought" to learn the patterns in this task. Program induction as a mechanism for predicting patterns on-the-fly may, in that case, be a distinctive capacity of mature human learning that only emerges gradually over childhood. Future work should aim to distinguish between these possibilities by using forced-choice tasks to reduce search demands and employing implicit measures such as surprise to separate inference from decision-making and motor planning.

Like younger children, and unlike older children and adults, the two rhesus macaques we tested largely relied on simpler local extrapolation strategies. One of the monkeys did learn to anticipate when a pattern would repeat by guessing a previously-observed point, demonstrating an ability to infer repeating structure even if not the exact algorithm. Importantly though, unlike children, the monkeys were trained on tens of thousands of examples. The monkeys'

persistent use of local extrapolation strategies may seem difficult to reconcile with past work demonstrating that nonhuman primates can learn structured concepts after extensive training. For instance, monkeys can learn to produce center-embedded sequences of the form "{[]}" [42] and to copy and reverse spatial sequences, e.g. learning on the input of ABC to produce ABC — CBA [54]. A common feature of those tasks, however, is that the monkeys did not have to synthesize novel algorithms *on-the-fly*. Instead, they learned one or two algorithms and generalized them to novel inputs. Our task requires not simply applying a learned algorithm, but learning a *new* algorithm based on sparse data on each trial.

It is quite plausible that the monkeys were unable to learn structures that they could theoretically represent — if taught variants of a single or small set of patterns — due to memory and processing constraints that make it challenging to perform online inference over longer-range spatiotemporal dependencies. In support of this hypothesis, we found that a transformer model with general learning mechanisms but highly constrained memory (context window of 2-3) learned biases that largely recapitulated those of the monkeys. This is consistent with other work showing that monkeys can learn short-range dependencies but struggle with longer-range dependencies relative to children [46, 67–69], and aligns with a proposal from Cantlon and Piantadosi (2024) that differences in human and nonhuman primate cognition are primarily driven by differences in overall information processing capacity [70]. However, our results also suggest a path towards unifying this account with another perspective on the human cognitive "singularity," based on evidence that humans alone use "languages of thought" with discrete symbols and recursive, compositional rules [40]. In particular, although both young children and nonhuman primates may fail to flexibly learn novel algorithms on-the-fly in unfamiliar and open-ended domains, an extended childhood and greater overall information-processing capacity may enable us to acquire these abilities.

An important direction for future work will be to broaden and generalize our findings by testing a larger sample of monkeys as well as other animals. Songbirds, and corvids in particular, have shown a propensity for learning compositional structure that may exceed most primates, and offer a particularly interesting test case for nonhuman program learning [49, 50, 71]. Another important direction is to better understand how people are able to search for programs efficiently. The algorithms underlying human program learning remain mysterious given the computational difficulty of navigating large spaces of combinatorial hypotheses. One promising avenue for understanding how program induction might become increasingly efficient with experience is "library learning" techniques, in which learners add increasingly abstract program fragments to their repertoire of available primitives [72, 73]. Future developmental work might probe the roles of language and abstraction learning [74–76] in efficient program

search through longitudinal or curriculum-based studies. Finally, we hope that our paradigm and results will inspire neuroscientific work aimed at identifying how the brain encodes and performs probabilistic inference over programs or other structured representations.

# Methods

## Participants

### Adults

N = 20 adults were recruited from Prolific ($M_{age}$ = 45.9) and completed the same experiment as children, but indicated guesses by clicking on their computer screen rather than tapping on a tablet. Each adult completed all 21 sequences in a random order. They took 21 minutes to complete the study on average and were paid $3.75.

### Children

We recruited 110 3- to 7-year old children (after exclusion) using the Children Helping Science online platform. Parents or legal guardians gave verbal consent for their child to participate after being given a written description of the study. To ensure that each child could be tested on as many of the patterns as possible while remaining engaged, we broke the study into multiple sessions (over distinct days) to limit the attentional burden for any one day. Children who completed the first session and passed inclusion criteria were invited to return for subsequent sessions, with regular reminder emails after each one. Children who completed at least 8 test sequences were eligible for inclusion.

To make the game engaging, sequences were indicated by a star with a happy face that moved across the screen, leaving smaller stars at previous locations (see Fig. 1). Before playing, children were told they would be playing a game with a star that makes patterns in the sky. They were told that they would first have to watch where the star goes, and then guess where it would go next. On each sequence, the star would first make two movements, revealing the first three points. A sound would then play indicating that it was time for the child to make their first guess by tapping on the screen. A small pink circle would appear where the child tapped. If their guess was correct, a pleasant chime would play and the star would spin. Otherwise, a melancholy swooping sound would play. The star would then move to its next location and the child would be allowed to guess again. Each child first completed a practice sequence which was a straight line. If children did not get at least 3 of 9 predictions correct on this initial sequence they completed the same sequence again. To ensure that our analyses only included children who understood the task, we excluded children who did not get at least 3 of 9 predictions correct on the initial sequence within two attempts. This resulted in the exclusion of 31/54 3-year-olds, 17/42 4-year-olds, 9/36 5-year-olds, 4/21 6-year-olds, and 1/19 7-year-olds.

After the practice sequence in the first session, children were then guided to complete 8 randomly selected sequences in each session, so that all 24 sequences would be completed after 3 sessions, with the option to stop early or complete more sequences within a session if they wished. Children completed

8.04 sequences and took 9.82 minutes per session on average, and completed 2.36 sessions and 19.01 sequences in total on average. Parents were compensated with $5 for each of the first two sessions and $10 after completing all the sequences.

### Rhesus Macaques

We additionally tested two adult rhesus macaques on this task. At the time of testing, Monkey 1 was 3 years and 5 months old, and Monkey 2 was 3 years and 6 months old. Animals were kept on a diet of monkey chow, fruits and vegetables, and *ad libitum* water. All procedures, care, and housing were in accordance with regulations of the Harlow Primate Lab, veterinary staff, and an Institutional Animal Care and Use Committee (IACUC) protocol. Monkeys performed the task on a tablet fixed outside of their cages. To reduce confusion, monkeys did not see the happy star display, but a simpler, analogous display in which sequence locations were indicated by red circles on a white background, with a larger, brighter circle at the most recently revealed location.

The monkeys were trained to perform the sequence prediction task on a broad set of training sequences over the course of 6 months (before which they were experimentally naive). They received food rewards for correct predictions. The training set was gradually expanded to include more complex classes of functions, from horizontal lines, to all lines, to other polynomial curves, sine waves, spirals, zigzags, and finally to repeated short sequences of points (e.g. triangles). Both monkeys' performance improved throughout training.

After 10 weeks of training, the monkeys began performing 5-10 test sequences per day, along with 40-50 training sequences in a random order over the course of 5 weeks. To counteract the linearity bias observed in testing, all patterns that could be successfully predicted using linear extrapolation were removed from the training set, and the monkeys were trained for 6 more weeks until they had each performed 150 trials of each remaining sequence type. Finally the test sequences were reintroduced over the course of 5 weeks as in the initial testing phase.

## Computational Models

All models were implemented in Julia using the probabilistic programming library Gen.jl for inference [62]. In Gen.jl, the assumptions of the problem-domain are encoded in a probabilistic generative function, which defines a data-generating process involving random choices. In each of our models, the generative function includes random choices determining the synthesized programs, parameters, and data (i.e. a sequence of points), and thus specifies a joint probability distribution over these. During inference (implemented with Sequential Monte Carlo with Markov Chain Monte Carlo rejuvenation as described in the main text) we then approximately sample from this distribution, conditioned on data. Each model had 20 particles for SMC and used 100,000 MCMC rejuvenation steps after each timepoint. Each model re-sampled noise, hyperpriors, and real-valued parameters, and the program synthesis models also re-sampled program trees as described in the main text.

## Fixed-structure models

### Bayesian Polynomial Ridge Regression Model

In classical Ridge regression, the goal is to minimize the objective function $\|y - \beta X\|_2^2 + \alpha \cdot \|\beta\|_2^2$, where $y$ is some observed data, $\beta$ is a set of weights, $X$ are features (i.e., $x$, $y$, and $t$), and $\alpha$ is a regularization term setting the penalty on large $\beta$ coefficients. Bayesian Ridge regression allows the hyperparameter on the weight regularization term, and the amount of uncertainty in the weights, to be inferred from the data. This then takes the form $p(y \mid \lambda,\ \alpha, X) = \mathcal{N}(y \mid \beta \cdot X,\ \alpha)$. $\alpha$ and $\lambda$ represent the precision of the noise and the precision of the weights, respectively. These parameters are assumed to be drawn from a Gamma distribution, with shape and rate (inverse scale) hyperparameters set to 1, and fit jointly with the weight coefficients to the data. All $x$, $y$, and $t$ values were standardized across the entire sequence.

### Non-Compositional GP Model

Our GP model had four base kernels: Constant, Linear, Radial Basis Function, and Periodic. In the SI Methods, we provide the mathematical formulations for each of these kernels and their typical use cases. The GP model assumes that there are two kernels that independently model the $x$ and $y$ dimensions; the kernel types, their input dimension ($x$, $y$, or $t$), and their associated parameters are both inferred. We used a uniform prior over the kernel type and a uniform prior on its input dimension. There was also a gamma prior on noise added to the covariance matrix. To compute the likelihood of the data given a kernel and its parameters, the sampled kernels are used to compute covariance matrices between two dimensions, e.g. $K_{X,Y}$. The diagonals of these matrices determine variance at a particular point, and the sampled noise value is added to the diagonal. Output values are then sampled from multivariate normal distributions parameterized by $\vec{\mu} = 0$ and the covariance matrices.

### Linear model

The linear model infers the angle and magnitude of a vector specifying the movement from any one point to the next in a sequence, conditioning only on the previous three points (two movements) in the sequence. The angle is drawn from a uniform distribution between -4 and 4 (with a value of 1 corresponding to 90 degrees) and the magnitude is drawn from an exponential distribution with a rate parameter of 0.5. Observed points are assumed to be drawn from normal distributions with separate precision parameters for the $x$ and $y$ dimensions. These precision parameters are each drawn from gamma distributions, parameterized by shape and scale parameters themselves drawn from gamma distributions (with shape = 8, scale = 8 and shape = 2, scale = 10) respectively.

### Linear + previous point model

This model combines two hypotheses about how the next point is generated: a periodic hypothesis and a linear hypothesis. It infers a mixture weighting $p_{periodic}$, drawn from a beta distribution with $\alpha = \beta = 1$, which determines the relative contribution of the two hypotheses. Under the periodic

hypothesis, the next point is assumed to be drawn from a uniformly weighted mixture of normals, each centered at one of the previous unique points in the sequence, and with separate standard deviations for the $x$ and $y$ dimensions drawn from exponential distribution with a rate parameter of 3. Under the linear hypothesis, the next point is predicted from the average vector of the last two movements (3 points) in the sequence. The true next point is assumed to be drawn from a normal distribution centered at this point, with separate standard deviations for the $x$ and $y$ dimensions again drawn from exponential distributions with a rate parameter of 3.

# Program synthesis models

## LoT model

The LoT grammar draws inspiration from [24], which defined a simple Logo-like drawing language to model human representations of geometric shapes. That language includes a mixture of control operations (e.g. repetition), motor commands (e.g. turn), and numeric expressions, which in combination efficiently generates myriad canonical and more complex geometric shapes, though is also limited in certain ways (e.g., cannot draw a right triangle). Our grammar, which is described in more detail in the SI Methods, extends theirs to include access to inner-state variables and a wider range of motor commands and algebraic operators, making it highly expressive. Programs drawn from this grammar define algorithms for generating unboundedly-long sequences of $x$ and $y$ positions over time.

**function** GenerativeFunction($T$)
  $P \sim PCFG$
  $\theta \sim \text{Uniform}(-4, 4)$
  $s \sim \text{Exp}(0.5)$
  $\vec{\mu_x}, \vec{\mu_y} = \text{RunProgram}(P, \theta, s, T)$
  $\alpha_x \sim \text{Gamma}(10, 10)$
  $\beta_x \sim \text{Gamma}(10, 10)$
  $\eta_x \sim \text{Gamma}(\alpha_x, \beta_x)$
  $\alpha_y \sim \text{Gamma}(10, 10)$
  $\beta_y \sim \text{Gamma}(10, 10)$
  $\eta_y \sim \text{Gamma}(\alpha_y, \beta_y)$
  **for** $t \leftarrow 1$ to $T$ **do**
    $x_t \sim \mathcal{N}(\mu_{x,t}, (1/\sqrt{\eta}_x))$
    $y_t \sim \mathcal{N}(\mu_{y,t}, (1/\sqrt{\eta}_y)$

Here, the generative function specifies a joint probability distribution over the program, initial values of the angle and speed internal state variables, noise parameters, and the $x$ and $y$ location of each point in the sequence. The input to the function is $T$, the number of points in the sequence so far. In the generative function, the initial program $P$ is sampled from the PCFG. We also sample initial values for the internal state variables $\theta$ and $s$ which determine the starting speed and heading for $P$. The RunProgram function returns the sequence of $x$ and $y$ locations ($\vec{\mu_x}$ and $\vec{\mu_y}$ ) specified by $P$ and these parameters.

In RunProgram, $P$ is evaluated as follows. The internal state variables angle ($\boldsymbol{\theta}$), where a value

of 1 corresponds to 90 degrees, speed ($\boldsymbol{s}$), $x$ and $y$ positions ($\boldsymbol{x}$, $\boldsymbol{y}$), timepoint ($\boldsymbol{t}$), and number of program executions ($\boldsymbol{c}$), are continuously tracked and updated. At the beginning of each timepoint, $\boldsymbol{x}$ and $\boldsymbol{y}$ are set to the true x and y position of the point observed at the previous timepoint, so that $P$ determines the movement from the previous true point to the next point at each timepoint. Specifically, each call in $P$ either specifies the control structure of the program, updates the internal state, or specifies the next point in the sequence with a `stay` or `move` call.

When `stay` is called, the current $\boldsymbol{x}$ and $\boldsymbol{y}$ values are added to $\vec{\mu_x}$ and $\vec{\mu_y}$ respectively. When `move` is called, $\boldsymbol{x}$ and $\boldsymbol{y}$ are updated according to the current speed and angle of movement, and these updated values are added to $\vec{\mu_x}$ and $\vec{\mu_y}$ respectively.

$P$ is wrapped in an outer `continue` call which repeatedly executes the program, and increments the internal state execution count variable $\boldsymbol{c}$, until $\vec{\mu_x}$ and $\vec{\mu_y}$ are of length $T$. Programs without a call to `move` or `stay` will not produce output, and so $\vec{\mu_x}$ and $\vec{\mu_x}$ are set to $[x_0]_T$ and $[y_0]_T$ respectively.

### GP structure learning model

**function** GenerativeFunction(input)
$n \leftarrow$ length(input)
$k^{(x)} \sim$ PCFG
$\sigma_x \sim \text{Gamma}(0.1, 0.2)$
$K^{(x)} \leftarrow \text{ComputeCovarianceMatrix}(k^{(x)}, \text{input})$
$K^{(x)} \leftarrow K^{(x)} + \sigma_x^2 I_n$
$k^{(y)} \sim$ PCFG
$\sigma_y \sim \text{Gamma}(0.1, 0.2)$
$K^{(y)} \leftarrow \text{ComputeCovarianceMatrix}(k^{(y)}, \text{input})$
$K^{(y)} \leftarrow K^{(y)} + \sigma_y^2 I_n$
$\mathbf{x} \sim \mathcal{N}(\mathbf{0}, K^{(x)})$
$\mathbf{y} \sim \mathcal{N}(\mathbf{0}, K^{(y)})$

Here, the generative function specifies a joint probability distribution over the kernel functions, noise parameters, and output values (point locations) for both the x and y dimensions. The input is a 3-dimensional vector of previous timepoints, and x and y locations of points at those timepoints. For each output dimension, the kernel is sampled from the PCFG as described in the main text. The ComputeCovarianceMatrix call takes the sampled kernel function (which specifies the covariance between a pair of input points), and the input, and computes a matrix of the covariance between every pair of input points. The diagonal of this matrix is the variance at a particular point, and the sampled noise value is added to the diagonal. Output values are then sampled from a multivariate normal distribution parameterized by $\vec{\mu} = 0$ and the computed covariance matrix.

## Transformer model architecture and training

We implemented a GPT-style causal transformer model based on the nanoGPT architecture [64] to serve as a computational benchmark for the monkey spatial sequence learning task. The model consisted of 6 transformer layers, each containing 8 attention heads with 64-dimensional embeddings.

The architecture employed causal self-attention with layer normalization and residual connections, following standard transformer design principles. Multi-layer perceptrons within each block used a $4\times$ expansion factor (64 to 256 hidden units) with GELU activation functions. Dropout regularization was applied at a rate of 0.1 across attention weights and residual connections.

Input sequences of 2D spatial coordinates were processed through a linear embedding layer ($2 \rightarrow 64$ dimensions) followed by hyperbolic tangent activation. Learned positional embeddings were added to encode temporal order within the sequence. The model output consisted of four values per time step: predicted next $x$ and $y$ coordinates, and corresponding uncertainty estimates (standard deviations) for each spatial dimension. The output layer applied a linear transformation ($64 \rightarrow 4$ dimensions) followed by sigmoid activation to ensure positive uncertainty values.

The transformer was trained using Gaussian Negative Log Likelihood Loss, which jointly optimizes coordinate predictions and uncertainty estimates by treating the model's spatial predictions as parameters of a bivariate Gaussian distribution. Training employed the Adam optimizer with a batch size of 128, L2 regularization of 0.0001, and 200,000 training steps. The learning rate was decreased exponentially during training from an initial value of 0.0005 to a final value of 0.00007.

We trained models with a large dataset of 100,000 spatial sequences (5,000 per function type) generated using the same Python script that created the monkey training stimuli. To investigate the role of temporal dependencies in spatial sequence learning, we systematically varied the context length from 1 to 13 previous positions, training separate models for each context window under both data paradigms. All models were implemented in PyTorch and final model performance was evaluated on the same test trajectories used in the behavioral experiments.

# References


1. Kemp, C., Hamacher, D. W., Little, D. R. & Cropper, S. J. Perceptual grouping explains similarities in constellations across cultures. *Psychological Science* **33,** 354–363 (2022).

2. Frank, M. C. Bridging the data gap between children and large language models. *Trends in Cognitive Sciences* **27,** 990–992 (2023).

3. Schröder, S., Morgenroth, T., Kuhl, U., Vaquet, V. & Paaßen, B. Large Language Models Do Not Simulate Human Psychology. *arXiv preprint arXiv:2508.06950* (2025).

4. Xie, H. & Zhu, J.-Q. Centaur May Have Learned a Shortcut that Explains Away Psychological Tasks. *PsyArXiv* (2025).

5. Xu, F. & Garcia, V. Intuitive statistics by 8-month-old infants. *Proceedings of the National Academy of Sciences* **105,** 5012–5015 (2008).

6. Kirkham, N. Z., Slemmer, J. A. & Johnson, S. P. Visual statistical learning in infancy: Evidence for a domain general learning mechanism. *Cognition* **83,** B35–B42 (2002).

7. Fiser, J. & Aslin, R. N. Statistical learning of new visual feature combinations by infants. *Proceedings of the National Academy of Sciences* **99,** 15822–15826 (2002).

8. Slone, L. K. & Johnson, S. P. When learning goes beyond statistics: Infants represent visual sequences in terms of chunks. *Cognition* **178,** 92–102 (2018).

9. Fló, A., Benjamin, L., Palu, M. & Dehaene-Lambertz, G. Sleeping neonates track transitional probabilities in speech but only retain the first syllable of words. *Scientific reports* **12,** 4391 (2022).

10. Stahl, A. E., Romberg, A. R., Roseberry, S., Golinkoff, R. M. & Hirsh-Pasek, K. Infants segment continuous events using transitional probabilities. *Child development* **85,** 1821–1826 (2014).

11. Saffran, J. R., Johnson, E. K., Aslin, R. N. & Newport, E. L. Statistical learning of tone sequences by human infants and adults. *Cognition* **70,** 27–52 (1999).

12. Pelucchi, B., Hay, J. F. & Saffran, J. R. Learning in reverse: Eight-month-old infants track backward transitional probabilities. *Cognition* **113,** 244–247 (2009).

13. Ferguson, B., Franconeri, S. L. & Waxman, S. R. Very young infants learn abstract rules in the visual modality. *PloS one* **13,** e0190185 (2018).

14. Dawson, C. & Gerken, L. From domain-generality to domain-sensitivity: 4-month-olds learn an abstract repetition rule in music that 7-month-olds do not. *Cognition* **111,** 378–382 (2009).

15. Santolin, C., Zacharaki, K., Toro, J. M. & Sebastian-Galles, N. Abstract processing of syllabic structures in early infancy. *Cognition* **244,** 105663 (2024).

16. Marcus, G. F., Vijayan, S., Bandi Rao, S. & Vishton, P. M. Rule learning by seven-month-old infants. *Science* **283,** 77–80 (1999).

17. Johnson, S. P. *et al.* Abstract rule learning for visual sequences in 8-and 11-month-olds. *Infancy* **14,** 2–18 (2009).

18. Koulaguina, E. & Shi, R. Abstract rule learning in 11-and 14-month-old infants. *Journal of psycholinguistic research* **42,** 71–80 (2013).

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

20. 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).

21. Mollica, F. & Piantadosi, S. T. Logical word learning: The case of kinship. *Psychonomic Bulletin & Review,* 1–34 (2019).

22. Piantadosi, S. T., Tenenbaum, J. B. & Goodman, N. D. Bootstrapping in a language of thought: A formal model of numerical concept learning. *Cognition* **123,** 199–217 (2012).

23. Tillmann, B. Music and language perception: expectations, structural integration, and cognitive sequencing. *Topics in cognitive science* **4,** 568–584 (2012).

24. Sablé-Meyer, M. *et al.* Sensitivity to geometric shape regularity in humans and baboons: A putative signature of human singularity. *Proceedings of the National Academy of Sciences* **118,** e2023123118 (2021).

25. Sablé-Meyer, M., Ellis, K., Tenenbaum, J. & Dehaene, S. A language of thought for the mental representation of geometric shapes. *Cognitive Psychology* **139,** 101527 (2022).

26. Mills, T., Tenenbaum, J. B. & Cheyette, S. J. *Human spatiotemporal pattern learning as probabilistic program synthesis* in *Thirty-seventh Conference on Neural Information Processing Systems* (2023).

27. Lake, B. M., Salakhutdinov, R. & Tenenbaum, J. B. Human-level concept learning through probabilistic program induction. *Science* **350,** 1332–1338 (2015).

28. Schulz, E., Tenenbaum, J. B., Duvenaud, D., Speekenbrink, M. & Gershman, S. J. Compositional inductive biases in function learning. *Cognitive psychology* **99,** 44–79 (2017).

29. DeLosh, E. L., Busemeyer, J. R. & McDaniel, M. A. Extrapolation: the sine qua non for abstraction in function learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition* **23,** 968 (1997).

30. Griffiths, T. L., Lucas, C., Williams, J. & Kalish, M. Modeling human function learning with Gaussian processes. *Advances in neural information processing systems* **21** (2008).

31. Piantadosi, S. T., Tenenbaum, J. B. & Goodman, N. D. The logical primitives of thought: Empirical foundations for compositional cognitive models. *Psychological review* **123,** 392 (2016).

32. Feldman, J. Minimization of Boolean complexity in human concept learning. *Nature* **407,** 630–633 (2000).

33. Piantadosi, S. T. & Jacobs, R. A. Four problems solved by the probabilistic language of thought. *Current Directions in Psychological Science* **25,** 54–59 (2016).

34. Rule, J. S. *et al.* Symbolic metaprogram search improves learning efficiency and explains rule learning in humans. *Nature Communications* **15,** 6847 (2024).

35. Goodman, N. D., Tenenbaum, J. B., Feldman, J. & Griffiths, T. L. A rational analysis of rule-based concept learning. *Cognitive science* **32,** 108–154 (2008).

36. Amalric, M. *et al.* The language of geometry: Fast comprehension of geometrical primitives and rules in human adults and preschoolers. *PLoS computational biology* **13,** e1005273 (2017).

37. Rule, J. S., Tenenbaum, J. B. & Piantadosi, S. T. The child as hacker. *Trends in cognitive sciences* **24,** 900–915 (2020).

38. Rothe, A., Lake, B. M. & Gureckis, T. Question asking as program generation. *Advances in neural information processing systems* **30** (2017).

39. Hauser, M. D., Chomsky, N. & Fitch, W. T. The faculty of language: what is it, who has it, and how did it evolve? *Science* **298,** 1569–1579 (2002).

40. Dehaene, S., Al Roumi, F., Lakretz, Y., Planton, S. & Sablé-Meyer, M. Symbols and mental programs: a hypothesis about human singularity. *Trends in Cognitive Sciences* (2022).

41. Pomiechowska, B., Bródy, G., Téglás, E. & Kovács, Á. M. Early-emerging combinatorial thought: Human infants flexibly combine kind and quantity concepts. *Proceedings of the National Academy of Sciences* **121,** e2315149121 (2024).

42. Ferrigno, S., Cheyette, S. J., Piantadosi, S. T. & Cantlon, J. F. Recursive sequence generation in monkeys, children, US adults, and native Amazonians. *Science Advances* **6,** eaaz1002 (2020).

43. Coates, N., Siegel, M., Tenenbaum, J. & Schulz, L. *Representations of Abstract Relations in Early Childhood* in *Proceedings of the Annual Meeting of the Cognitive Science Society* **45** (2023).

44. Martins, M. D., Laaha, S., Freiberger, E. M., Choi, S. & Fitch, W. T. How children perceive fractals: Hierarchical self-similarity and cognitive development. *Cognition* **133,** 10–24 (2014).

45. Cole, P. D. & Adamo, S. A. Cuttlefish (Sepia officinalis: Cephalopoda) hunting behavior and associative learning. *Animal Cognition* **8,** 27–30 (2005).

46. Santolin, C. & Saffran, J. R. Constraints on statistical learning across species. *Trends in Cognitive Sciences* **22,** 52–63 (2018).

47. Ciccione, L., Dighiero-Brecht, T., Claidière, N., Fagot, J. & Dehaene, S. Can non-human primates extract the linear trend from a noisy scatterplot? *iScience* **28** (2025).

48. Saffran, J. *et al.* Grammatical pattern learning by human infants and cotton-top tamarin monkeys. *Cognition* **107,** 479–500 (2008).

49. Liao, D. A., Brecht, K. F., Johnston, M. & Nieder, A. Recursive sequence generation in crows. *Science Advances* **8,** eabq3356 (2022).

50. Schmidbauer, P., Hahn, M. & Nieder, A. Crows recognize geometric regularity. *Science Advances* **11,** eadt3718 (2025).

51. Englund, M., Whitham, W., Conway, C. M., Beran, M. J. & Washburn, D. A. Nonhuman primates learn adjacent dependencies but fail to learn nonadjacent dependencies in a statistical learning task with a salient cue. *Learning & Behavior* **50,** 242–253 (2022).

52. Taraban, R. & Bandara, A. Beyond recursion: critique of Hauser, Chomsky, and Fitch. *East European Journal of Psycholinguistics* **4,** 58–66 (2017).

53. Tian, L. Y. *et al.* Neural representation of action symbols in primate frontal cortex. *bioRxiv* (2025).

54. Jiang, X. *et al.* Production of supra-regular spatial sequences by macaque monkeys. *Current Biology* **28,** 1851–1859 (2018).

55. Lucas, C. G., Griffiths, T. L., Williams, J. J. & Kalish, M. L. A rational model of function learning. *Psychonomic bulletin & review* **22,** 1193–1215 (2015).

56. Wu, C. M., Schulz, E. & Gershman, S. J. Generalization as diffusion: human function learning on graphs. *BioRxiv,* 538934 (2019).

57. Li, M. Y., Callaway, F., Thompson, W. D., Adams, R. P. & Griffiths, T. L. Learning to Learn Functions. *Cognitive Science* **47,** e13262 (2023).

58. Kalish, M. L., Griffiths, T. L. & Lewandowsky, S. Iterated learning: Intergenerational knowledge transmission reveals inductive biases. *Psychonomic Bulletin and Review* **14,** 288 (2007).

59. Carroll, J. D. Functional learning: The learning of continuous functional mappings relating stimulus and response continua. *ETS Research Bulletin Series* **1963,** i–144 (1963).

60. Koh, K. & Meyer, D. E. Function learning: induction of continuous stimulus-response relations. *Journal of Experimental Psychology: Learning, Memory, and Cognition* **17,** 811 (1991).

61. Saad, F. A., Patton, B. J., Hoffmann, M. D., Saurous, R. A. & Mansinghka, V. K. *Sequential Monte Carlo Learning for Time Series Structure Discovery* in *International Conference on Machine Learning* (2023).

62. Cusumano-Towner, M. F., Saad, F. A., Lew, A. K. & Mansinghka, V. K. *Gen: a general-purpose probabilistic programming system with programmable inference* in *Proceedings of the 40th acm sigplan conference on programming language design and implementation* (2019), 221–236.

63. Vaswani, A. *et al.* Attention is all you need. *Advances in neural information processing systems* **30** (2017).

64. Karpathy, A. *nanoGPT: minimal, educational implementation of GPT in Nano-Python* `https://github.com/karpathy/nanoGPT`. 2022.

65. Zelazo, P. D. The Dimensional Change Card Sort (DCCS): A method of assessing executive function in children. *Nature protocols* **1,** 297–301 (2006).

66. Vinter, A., Puspitawati, I. & Witt, A. Children's spatial analysis of hierarchical patterns: Construction and perception. *Developmental Psychology* **46,** 1621 (2010).

67. Wilson, B., Smith, K. & Petkov, C. I. Mixed-complexity artificial grammar learning in humans and macaque monkeys: evaluating learning strategies. *European Journal of Neuroscience* **41,** 568–578 (2015).

68. Wilson, B. *et al.* Auditory artificial grammar learning in macaque and marmoset monkeys. *Journal of Neuroscience* **33,** 18825–18835 (2013).

69. Fitch, W. T. & Hauser, M. D. Computational constraints on syntactic processing in a nonhuman primate. *Science* **303,** 377–380 (2004).

70. Cantlon, J. F. & Piantadosi, S. T. Uniquely human intelligence arose from expanded information capacity. *Nature Reviews Psychology* **3,** 275–293 (2024).

71. Taylor, A. H., Hunt, G. R., Holzhaider, J. C. & Gray, R. D. Spontaneous metatool use by New Caledonian crows. *Current Biology* **17,** 1504–1507 (2007).

72. Ellis, K., Ritchie, D., Solar-Lezama, A. & Tenenbaum, J. B. Learning to infer graphics programs from hand-drawn images. *arXiv preprint arXiv:1707.09627* (2017).

73. Ellis, K. *et al.* Dreamcoder: growing generalizable, interpretable knowledge with wake–sleep bayesian program learning. *Philosophical Transactions of the Royal Society A* **381,** 20220050 (2023).

74. Wong, L. *et al.* From word models to world models: Translating from natural language to the probabilistic language of thought. *arXiv preprint arXiv:2306.12672* (2023).

75. Carey, S. *The origin of concepts* (Oxford University Press, 2009).

76. Carey, S. Conceptual change in childhood. *American Psychologist* **41,** 67–18 (1985).

# Supplementary Methods

## Test Stimuli

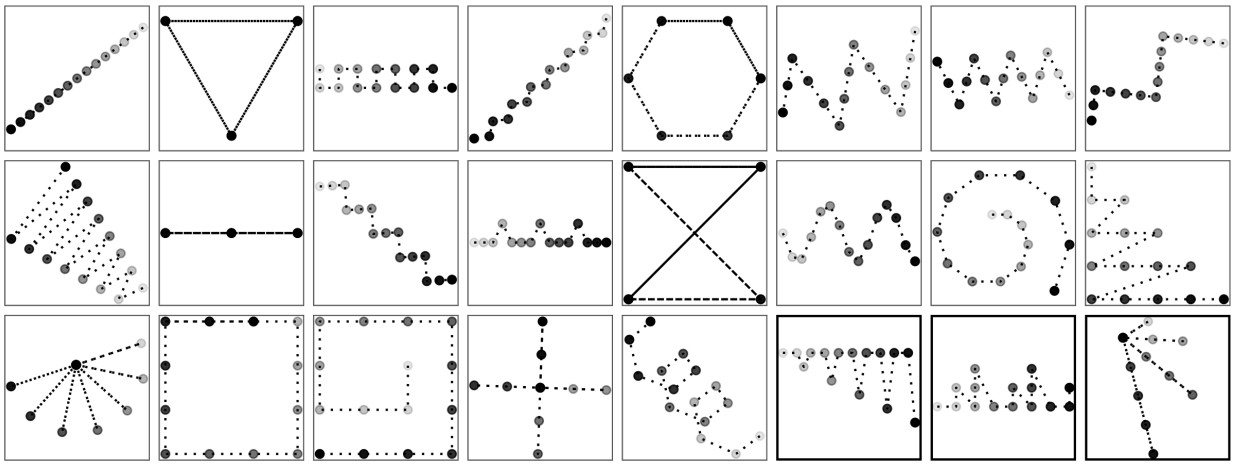

Figure S1: All test stimuli used in the experiment. The three rightmost cells in the bottom row, with bold borders, include stimuli that were selected specifically to test for a sensitivity to recursive structure, and were seen only by children.

## Bayesian data analysis of test sequences

As a way of measuring accuracy in the test sequences that accounts for individual motor error and inattention, we modeled participants' responses as a mixture of three components: (1) predicting near the true next point, (2) predicting near the previous point, and (3) guessing uniformly at random within the display bounds. For sequence type $s$ after $t$ steps within the sequence, the probability of a "true-next" response is

$$\theta_{\text{true}} \;=\; \text{logit}^{-1}(\alpha \;+\; \alpha_s \;+\; \beta_s\, t)\,,$$

where $\alpha$ is a global intercept, $\alpha_s$ is a sequence-type intercept (difficulty), $\beta_s$ is a non-negative within-sequence slope (learning with timepoint), and $t$ is the (standardized) timepoint.
The likelihood for each observed prediction $(x_t, y_t)$ (coordinates normalized to the unit-width

display) is

$$\begin{aligned}\log p(\hat{x}_t, \hat{y}_t) \;=\; \log\Big[&\theta_{\text{true}} \cdot \mathcal{N}\big(\hat{x} \mid x_t^{\text{true}}, \sigma^2_{\text{true}}\big)\, \mathcal{N}\big(\hat{y}_t \mid y_t^{\text{true}}, \sigma_m^2\big) \\ &+ \big(1-\theta_{\text{true}}\big)\,\pi \;\cdot\; \mathcal{N}\big(\hat{x}_t \mid x_{t-1}^{\text{true}}, \sigma_P^2\big)\, \mathcal{N}\big(\hat{y}_t \mid y_{t-1}^{\text{true}}, \sigma_P^2\big) \\ &+ \big(1-\theta_{\text{true}}\big)\,(1-\pi) \;\cdot\; \mathcal{U}\big(\hat{x}_t \mid 0, U_x\big)\, \mathcal{U}\big(\hat{y}_t \mid 0, U_y\big)\Big],\end{aligned}$$

where $(x_t^{\text{true}}, y_t^{\text{true}})$ are the coordinates of the true next point at time $t$ and $(x_{t-1}^{\text{true}}, y_{t-1}^{\text{true}})$ are the coordinates of the current point (previous target). The parameters $\sigma_m$ and $\sigma_P$ capture motor noise when responding around the true point and noise when responding around previous locations, respectively. The mixing proportion $\pi$ gives the share of non-true responses attributable to guessing around the previous point; the remainder $(1-\pi)$ corresponds to uniform random guessing within the display bounds, with $U_x = 1$ and $U_y$ set by the screen aspect ratio.

We used weakly informative priors:

$$\alpha \sim \mathcal{N}(0,3), \quad \alpha_s \sim \mathcal{N}(0,3), \quad \beta_s \sim \mathcal{N}(0,3) \text{ (truncated to } [0,\infty)),$$

$$\sigma_m \sim \text{Exponential}(1), \quad \sigma_P \sim \text{Exponential}(1), \quad \pi \sim \text{Beta}(1,1).$$

The model was implemented in Stan (RStan) and fit via Hamiltonian Monte Carlo with the No-U-Turn Sampler (4 chains, 2,000 iterations per chain). Convergence was assessed via $\widehat{R}$ and effective sample sizes.

## Computational model details

## Gaussian Process Model Kernels

### Constant Kernel

$$K(\mathbf{x}, \mathbf{x}') = c^2, \tag{2}$$

where $c$ is a constant.

### Linear Kernel

The Linear kernel is a simple kernel that allows the GP model to predict a linearly varying function. This kernel is useful when the underlying relationship between the inputs and outputs is assumed to be linear, with constant variance. Given input vectors $\mathbf{x}$ and $\mathbf{x}'$, the Linear kernel takes the form,

$$K(\mathbf{x}, \mathbf{x}') = \sigma^2 + \mathbf{x}^T \mathbf{x}', \tag{3}$$

where $\sigma^2$ is a constant offset.

### Radial Basis Function Kernel

The Radial Basis Function (RBF) kernel is one of the most commonly used kernels in GP regression. The RBF kernel characterizes the similarity between input vectors $\mathbf{x}$ and $\mathbf{x}'$ based on the Euclidean distance between them. Mathematically, this takes the form,

$$K(\mathbf{x}, \mathbf{x}') = \sigma^2 \exp\left(-\frac{(\mathbf{x} - \mathbf{x}')^T (\mathbf{x} - \mathbf{x}')}{2l^2}\right), \tag{4}$$

where $\sigma^2$ represents the overall variance of the data and $l$ is the length-scale hyperparameter controlling how quickly the similarity between two data points decays as their distance increases.

### Periodic Kernel

Lastly, the Periodic kernel is the product of an exponential and a sine-squared term. This kernel is often used for data that exhibits cyclical behavior, like weather patterns or seasonal trends. The Periodic kernel is defined as:

$$K(\mathbf{x}, \mathbf{x}') = \sigma^2 \exp\left(-\frac{2\sin^2(\pi \frac{\mathbf{x} - \mathbf{x}'}{p})}{l^2}\right), \tag{5}$$

where $p$ is the period of the function and $l$ is the length-scale hyperparameter.

The GP model assumes that there are two independent kernels for the $x$ and $y$ dimensions. We used a uniform prior over the kernel type and a uniform prior on its input dimension ($x$, $y$, or $t$). There was also a uniform prior on noise added to the covariance matrix. To compute the likelihood of the data given a kernel and its parameters, the sampled kernels are used to compute covariance matrices between two dimensions, e.g. $K_{X,Y}$. The diagonals of these matrices determine variance at a particular point, and the sampled noise value is added to the diagonal. Output values are then sampled from multivariate normal distributions parameterized by $\vec{\mu} = 0$ and the covariance matrices.

## LoT model primitives

```
state:= {x, y, θ, s, c} // current x and y location,
                        // angle and speed of movement,
                        // and program execution count
program:=
   /* compositional primitives */
   program; program     // run both programs
   repeat(program, n=param)   // run the program n times
   continue(program)  // run the program continuously
   subprogram(program)  // run the program, then reset the state

   /* motor and geometry primitives */
   move(θ=param, s=param,
        dx=param, dy=param)     // move and add current loc. to sequence
   stay( )     // add current loc. to sequence
   turn(dθ=param)     // change angle of movement
   reflect( )     // negate angle of movement
   accelerate(ds=param)     // change speed of movement
   change_x(dx=param)     // change x location
   change_y(dy=param)     // change y location

param:=
   c     // program execution count
   real     // real valued number
   integer     // integer valued number
   param [+, -, *, /, %] param     // mathematical operations
```

Figure S2: The LoT model grammar. Optional arguments show in gray. The model samples a program and an initial state, and executes the programs by wrapping it in a `continue` call, incrementing the program execution count `c` after each execution.

# Supplementary Results

## Monkey training

To account for monkeys' accuracy and learning trajectories for different types of patterns over the course of training, we constructed a model to infer their probability of guessing around the true next point in each sequence. This data analysis model is very similar to the data analysis model used for each population in the test sequences, with terms to account for learning pattern types across training. Specifically, on each trial, the monkeys' response is modeled as arising from a mixture of three components: (1) predicting near the true next point, (2) predicting near the previous point, and (3) guessing randomly on the screen. The probability of responding accurately to a given sequence type $s$ after having seen $t_s$ examples of that pattern is given by,

$$\theta_s = \text{logit}(\alpha_s + \beta_s \cdot t_s) \cdot L_s,$$

where $\alpha_s$ is an intercept, $\beta_s$ captures the effect of accumulated exposure to sequence type $s$ during training, and $L_s$ represents the asymptote of accuracy. The likelihood for each observed

prediction $(x_i, y_i)$ is given by:

$$\log p(x_i, y_i) = \log \Big[ \theta_s \cdot \mathcal{N}(x_i \mid \mu_i^{\text{true}}, \sigma^2) \cdot \mathcal{N}(y_i \mid \nu_i^{\text{true}}, (\delta\sigma)^2) + (1 - \theta_{s_i}) \cdot \pi_{s_i} \cdot \mathcal{N}(x_i \mid \mu_i^{\text{prev}}, \sigma^2) \cdot \mathcal{N}(y_i \mid \nu_i^{\text{prev}}, (\delta\sigma)^2) + (1 - \theta_{s_i}) \cdot (1 - \pi_{s_i}) \cdot \mathcal{U}(x_i \mid 0, 1) \cdot \mathcal{U}(y_i \mid 0, 1) \Big],$$

where $\mu_i^{\text{true}}, \nu_i^{\text{true}}$ denote the true next location on trial $i$, $\mu_i^{\text{prev}}, \nu_i^{\text{prev}}$ denote the previous location, $\sigma$ represents motor noise, $\delta$ scales the $y$-axis by the screen aspect ratio, and $\pi_s$ captures the proportion of non-learning responses attributable to repeating the previous location for sequence type $s$.

We place weakly informative priors:

$$\alpha_s \sim \mathcal{N}(0, 3), \quad \beta_s \sim \mathcal{N}(0, 1), \quad L_s \sim \text{Beta}(1, 1), \quad \pi_s \sim \text{Beta}(1, 1), \quad \sigma \sim \text{Exponential}(1),$$

for each sequence type $s$. The model was implemented in RStan and fit using Hamiltonian Monte Carlo with the No-U-Turn Sampler.

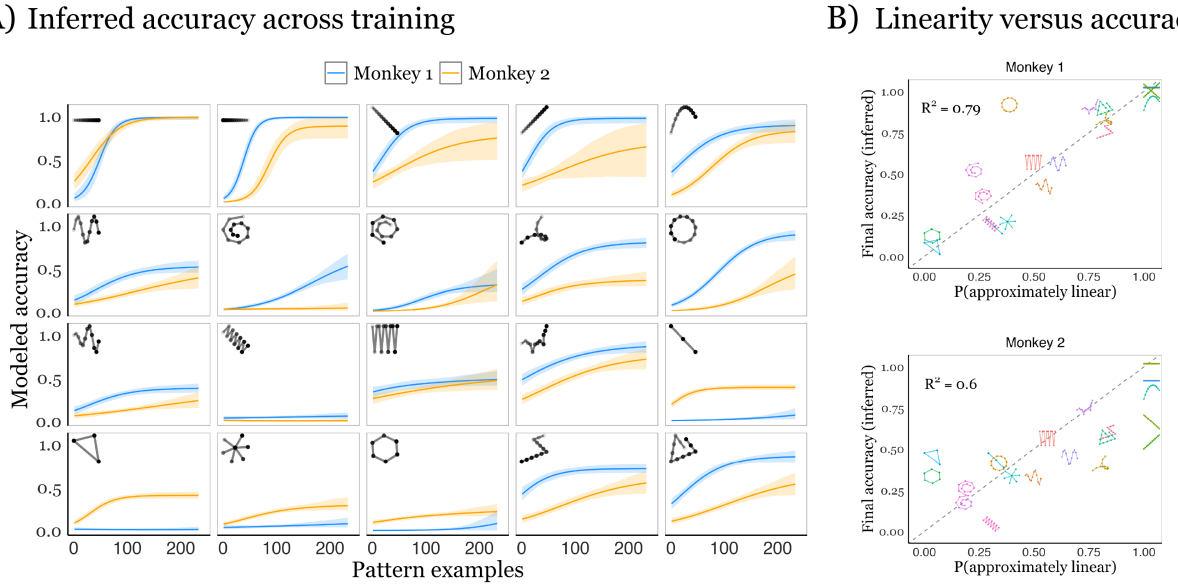


Figure S3: (A) Monkeys' inferred posterior mean probability of correctly guessing the next point in the sequence as a function of exposure to each pattern type across training. Error bars are 95% credible intervals. (B) The probability of responding correctly using a linear extrapolation strategy (x-axis) versus MAP inferred accuracy at the end of training (y-axis) for Monkey 1 (top) and Monkey 2 (bottom).

Fig S3A shows the mean and 95% credible interval for $\theta_s$ as a function of experience with

each pattern type over the course of training. Both monkeys had positive average slopes ($\beta_s$) for all pattern types, meaning they exhibited at least some degree of learning. However, there were large differences in the final inferred $\theta_s$, ranging from approximately 0 (e.g. Monkey 1 on repeating point patterns) to approximately 1 (e.g., for both monkeys on left-to-right patterns). Most of the variation in monkeys' average success on each pattern type can be explained by how often linear extrapolation would be expected to succeed. Fig S3B shows the probability the next point in a sequence was approximately linear versus monkeys final inferred $\theta_s$ at the end of training. The correlation between the (approximate) linearity of a pattern and the inferred accuracy at the end of training for Monkey 1 was $r = 0.89$ and $r = 0.77$ for Monkey 2. This indicates that, despite withholding patterns that were mostly linear for the second half of training, monkeys largely used linear extrapolation anyway.

We tested whether monkeys developed strategies that exceeded the performance of pure linear extrapolation by comparing their modeled end-of-training performance against the expected success rate using linear extrapolation for each pattern type. There was evidence that both monkeys outperformed linear extrapolation on a small subset of patterns. Monkey 1 had an inferred final accuracy of 88% for circular/ellipsoid patterns, which far exceeds the probability of success using linear extrapolation (32%), and was similarly substantially better than expected from linear extrapolation for inward-spirals (48% versus 19%) and spiky circles (87% versus 72%). Monkey 1 also had small (¡10%), but significant (> 99% credible interval), numerical advantages as well for outward spirals (32% versus 23%), polygons (8% versus 0%) and increasing polygons (86% versus 79%).

Monkey 2 showed a more limited but still notable ability to exceed linear extrapolation performance. The clearest advantages were observed for repeating patterns, where Monkey 2 achieved 41% accuracy on repeating point patterns compared to 0% expected from linear extrapolation, and 39% accuracy on repeating line patterns versus 29% from linear strategies. Monkey 2 also outperformed linear extrapolation on polygon patterns (29% versus 0%) and showed modest but significant advantages on outward spirals (22% versus 16%) and circles (37% versus 30%). However, Monkey 2 exhibited substantially worse performance than Monkey 1 on several pattern types where linear strategies would be optimal, including simple lines (62% versus 100% linear expectation), polynomials (80% versus 100%), and curvilinear patterns (37% versus 77% for curly patterns). This suggests that while Monkey 2 developed some capacity for recognizing repetitive and geometric patterns beyond linear prediction, this came at the cost of reduced proficiency with linear extrapolation strategies, resulting in suboptimal performance on inherently linear sequences.

We analyzed the discrepancy between each monkey's theoretical maximum performance ($L_s$)

and their actual end-of-training accuracy ($\theta_s$) to identify which, if any, patterns had significant potential for further learning. In general, both monkeys had reached their theoretical maximum for most pattern types, with absolute learning gaps averaging only 8% for Monkey 1 and 10% for Monkey 2. However, there was wide uncertainty in learning gaps for patterns where there was little evidence of meaningful learning in training (i.e., where performance barely or never improved). For instance, Monkey 1 had highly uncertain learning potential on repeating points (37.4% [0.3%, 96.0%]), polygons (31.4% [-5.9%, 84.9%]), and repeating lines (22.0% [-10.7%, 80.7%]) — in each case, accuracy was near-zero. Monkey 2 had the most uncertain learning potential on zigzag patterns (30.4% [0.3%, 94.6%]), outward spirals (30.4% [-7.3%, 74.0%]), and circles (23.8% [-9.6%, 61.1%]). This highlights the inherent difficulty of discerning whether learning a certain type of pattern is just *extremely slow* or if that pattern type is actually *unlearnable* for the monkeys.

## Learning within and across trials

To quantify within-sequence learning within each participant group, we fit a mixed-effects logistic regression to each group separately, predicting accuracy based on sequence timepoint with random slopes and intercepts for individual participants and sequences. Most groups exhibited within-sequence learning, with accuracy increasing with timepoint for 4-year-olds ($\beta = 0.37$, $p = 0.002$), 5-year-olds ($\beta = 0.52$, $p < .001$), 6-year-olds ($\beta = 0.78$, $p < .001$), 7-year-olds ($\beta = 0.98$, $p < .001$) and adults ($\beta = 1.96$, $p < .001$). However, this effect was not significant for 3-year-olds ($\beta = 0.099$, $p = .34$), or for monkeys ($\beta = 0.074$, $p = .31$).

To test whether children exhibited meta-learning over the course of the task as they completed more sequences, we fit a mixed-effects logistic regression model with fixed effects for age, prediction timepoint, sequence order, and their pairwise interactions, as well as random intercepts and slopes on timepoint for pattern types and random intercepts and slopes on timepoint and sequence order for participants. In addition to the effects of prediction timepoint ($\beta = 0.50$, $p < .001$), age ($\beta = 0.97$, $p < .001$), and their interaction ($\beta = 0.28$, $p < .001$), children also improved slightly across sequences ($\beta = 0.12$, $p < .001$). A small positive interaction between sequence order and timepoint ($\beta = 0.044$, $p = .00780$) suggests that in later sequences, within-sequence learning increases slightly. Adults also exhibit a moderate degree of improvement over the course of the task. In a mixed-effects logistic regression model with fixed effects for prediction timepoint and sequence order, their interaction, random intercepts and slopes on timepoint for pattern types, and random intercepts and slopes on timepoint and sequence order for participants, there is a small positive effect of sequence order on accuracy

($\beta = 0.27$, $p < .001$), along with a large positive effect of timepoint ($\beta = 2.00$, $p < .001$) and positive interaction ($\beta = 0.19$, $p < .001$).

## Linearity & accuracy

As an additional test of the degree to which each population relied on a linear extrapolation strategy, we computed the Euclidean distance of each predicted point from the linear trajectory defined by the previous two points. We used this as a predictor of accuracy in a mixed-effects logistic regression with timepoint and their interaction as covariates and random intercepts and slopes of these predictors with the distance from the previous point. In monkeys, accuracy decreased significantly as deviation from linearity increased ($\beta = -1.89$, $p < .001$), with a small positive interaction with time ($\beta = 0.08$, $p = .024$), suggesting a reliance on linear extrapolation that only slightly weakened as the patterns unfolded. Three-year-old children showed a similarly robust effect of linear distance on accuracy ($\beta = -1.07$, $p < .001$), and a marginal interaction with time ($\beta = 0.21$, $p = .073$). In contrast, 4- and 5-year-olds did not exhibit significant main effects of linearity ($p = .76$ and $p = .14$, respectively), but did show significant positive interactions with time ($\beta = 0.19$, $p = .022$; $\beta = 0.28$, $p = .001$). Six- and 7-year-olds showed slight (non-significant for 7-year-olds) main effects of linearity ($\beta = -0.49$, $p = .037$; $\beta = -0.38$, $p = .19$) but strong interactions with time ($\beta = 0.35$, $p < .001$; $\beta = 0.41$, $p < .001$), consistent with early linear extrapolation and subsequent learning of non-linear structure. Adults showed no reliable main effect of linearity ($p = .13$), but there was a strong interaction with time ($\beta = 0.66$, $p < .001$). Together, these results suggest that monkeys and younger children tended to rely on linear extrapolation and only slightly deviated from linear predictions as sequences unfolded; on the other hand, older children and adults either did not use linear extrapolation or deviated from linear predictions as the sequences progressed, indicative of inferring more complex (non-linear) functions to make predictions.

## Inferred accuracy

Between monkeys and humans of each age group, there was a positive relationship in average inferred accuracy across sequences between monkeys and 3-year-olds ($r = 0.46$, $p = .034$), but this relationship was not significant with 4-year-olds ($r = 0.18$, $p = .44$), 5-year-olds ($r = 0.32$, $p = .16$), 6-year-olds ($r = 0.33$, $p = .15$), 7-year-olds ($r = 0.38$, $p = .093$), or adults ($r = 0.17$, $p = .46$). Pairwise correlations in accuracy and inferred accuracy between groups are shown in Fig. S4 and Fig. S5.

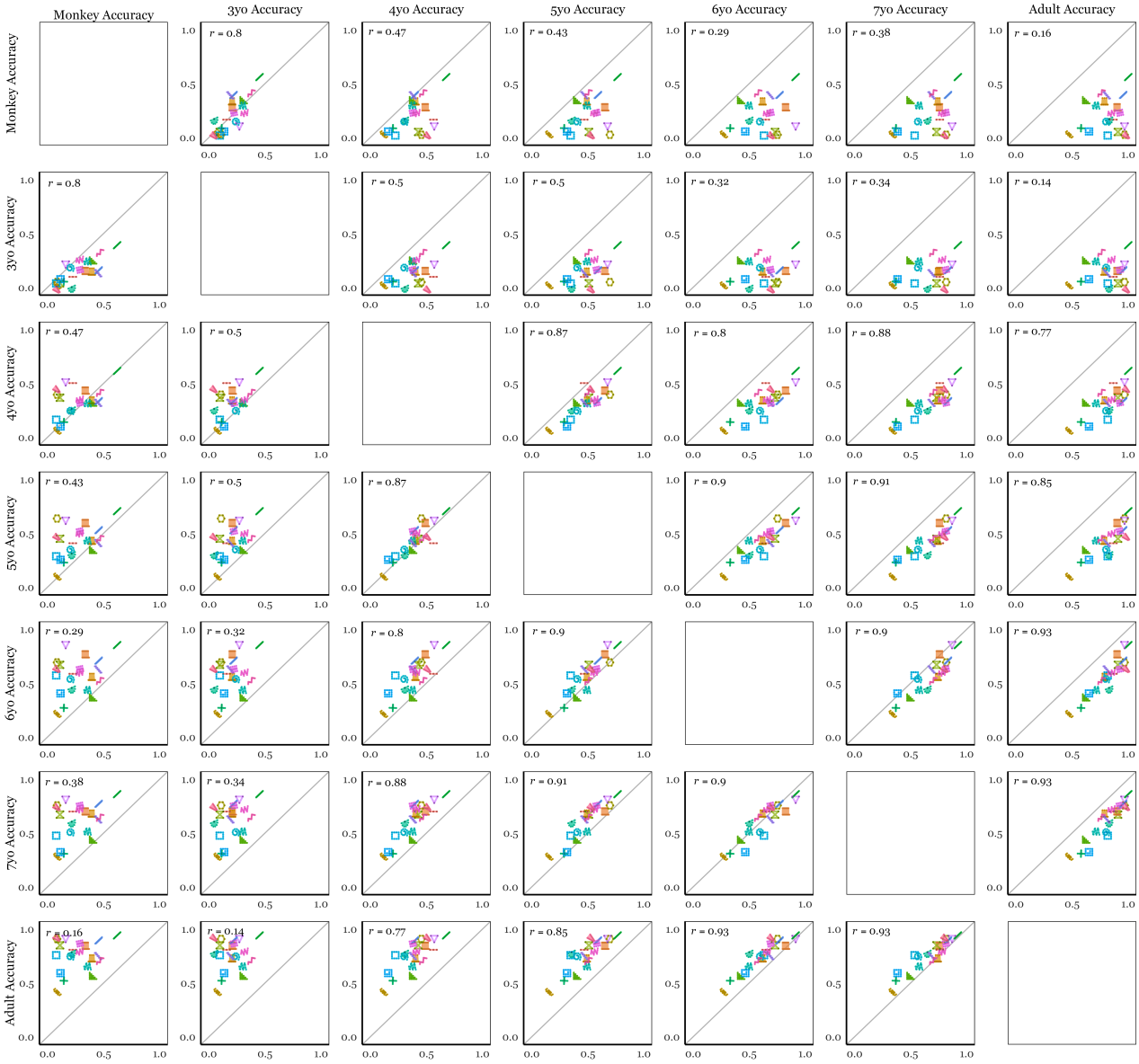


Figure S4: Pairwise correlations for raw accuracy on each test pattern between each population.

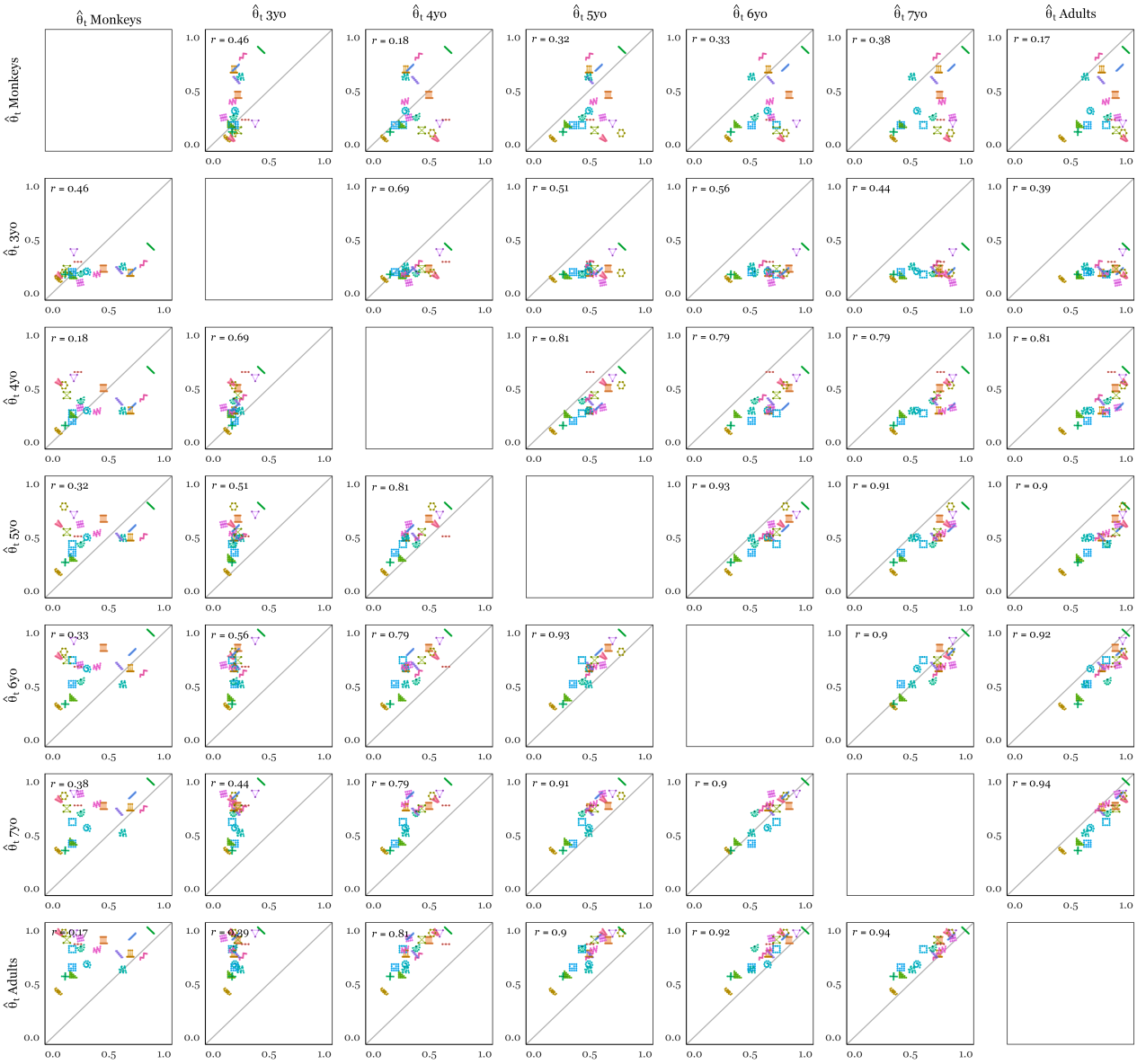


Figure S5: Pairwise correlations for inferred accuracy $\hat{\theta}_{\text{true}}$ on each test pattern across populations.

## Model likelihoods

| Model | $\sigma_m$ | $\theta_P$ | $\sigma_P$ | $\theta_R$ | $\text{LL}_{\text{lower}}$ | $\text{LL}_{\text{upper}}$ | $\text{LL}_{\text{mean}}$ |
|---|---|---|---|---|---|---|---|
| LoT | 0.01 | 0.22 | 0.08 | 0.00 | 34.30 | 35.90 | 35.11 |
| Comp. GP | 0.01 | 0.27 | 0.07 | 0.00 | 30.62 | 32.09 | 31.36 |
| GP | 0.01 | 0.27 | 0.08 | 0.00 | 29.20 | 30.63 | 29.91 |
| Polynomial | 0.00 | 0.12 | 0.01 | 0.01 | 26.00 | 27.22 | 26.62 |
| Linear + Prev. | 0.01 | 0.17 | 0.05 | 0.00 | 22.93 | 24.02 | 23.47 |
| Linear | 0.02 | 0.66 | 0.10 | 0.00 | 17.36 | 18.19 | 17.78 |

Table S1: Adults: Mean model parameters and LL bootstrapped means and 95% upper and lower bounds. LLs averaged across participants within each sequence, and then across sequences.

| Model | $\sigma_m$ | $\theta_P$ | $\sigma_P$ | $\theta_R$ | $\text{LL}_{\text{lower}}$ | $\text{LL}_{\text{upper}}$ | $\text{LL}_{\text{mean}}$ |
|---|---|---|---|---|---|---|---|
| LoT | 0.04 | 0.60 | 0.06 | 0.16 | 14.70 | 15.31 | 14.99 |
| Comp. GP | 0.04 | 0.61 | 0.06 | 0.15 | 14.59 | 15.18 | 14.87 |
| GP | 0.04 | 0.61 | 0.06 | 0.15 | 14.61 | 15.20 | 14.89 |
| Polynomial | 0.03 | 0.63 | 0.06 | 0.13 | 14.41 | 15.00 | 14.69 |
| Linear + Prev. | 0.03 | 0.41 | 0.06 | 0.09 | 15.34 | 15.95 | 15.63 |
| Linear | 0.04 | 0.33 | 0.06 | 0.18 | 14.93 | 15.55 | 15.22 |

Table S2: Monkeys: Mean model parameters and LL bootstrapped means and 95% upper and lower bounds. LLs averaged across trials and participants within each sequence, and then across sequences.

| Model | Age (y/o) | $\sigma_m$ | $\theta_P$ | $\sigma_P$ | $\theta_R$ | $\text{LL}_{\text{lower}}$ | $\text{LL}_{\text{upper}}$ | $\text{LL}_{\text{mean}}$ |
|---|---|---|---|---|---|---|---|---|
| LoT | 3 | 0.04 | 0.43 | 0.38 | 0.34 | 13.33 | 14.17 | 13.74 |
| Comp. GP | 3 | 0.06 | 0.45 | 0.35 | 0.35 | 13.25 | 14.08 | 13.66 |
| GP | 3 | 0.06 | 0.45 | 0.38 | 0.36 | 13.24 | 14.07 | 13.66 |
| Polynomial | 3 | 0.05 | 0.49 | 0.30 | 0.30 | 13.29 | 14.13 | 13.69 |
| Linear + Prev. | 3 | 0.04 | 0.43 | 0.21 | 0.28 | 13.59 | 14.44 | 14.00 |
| Linear | 3 | 0.05 | 0.33 | 0.38 | 0.33 | 13.54 | 14.41 | 13.97 |
| LoT | 4 | 0.03 | 0.48 | 0.13 | 0.15 | 17.93 | 18.85 | 18.39 |
| Comp. GP | 4 | 0.03 | 0.49 | 0.13 | 0.16 | 17.71 | 18.63 | 18.17 |
| GP | 4 | 0.04 | 0.48 | 0.13 | 0.14 | 17.57 | 18.45 | 18.00 |
| Polynomial | 4 | 0.02 | 0.47 | 0.12 | 0.13 | 17.15 | 18.02 | 17.58 |
| Linear + Prev. | 4 | 0.02 | 0.44 | 0.05 | 0.11 | 16.56 | 17.37 | 16.95 |
| Linear | 4 | 0.04 | 0.51 | 0.15 | 0.13 | 15.12 | 15.88 | 15.49 |
| LoT | 5 | 0.02 | 0.48 | 0.07 | 0.07 | 20.07 | 21.03 | 20.53 |
| Comp. GP | 5 | 0.02 | 0.49 | 0.07 | 0.07 | 19.45 | 20.40 | 19.92 |
| GP | 5 | 0.02 | 0.50 | 0.07 | 0.06 | 19.14 | 20.06 | 19.58 |
| Polynomial | 5 | 0.01 | 0.47 | 0.05 | 0.05 | 18.47 | 19.38 | 18.91 |
| Linear + Prev. | 5 | 0.01 | 0.44 | 0.06 | 0.05 | 17.19 | 18.00 | 17.59 |
| Linear | 5 | 0.03 | 0.54 | 0.09 | 0.06 | 15.59 | 16.37 | 15.98 |
| LoT | 6 | 0.02 | 0.35 | 0.07 | 0.04 | 23.49 | 24.74 | 24.14 |
| Comp. GP | 6 | 0.02 | 0.36 | 0.07 | 0.04 | 22.32 | 23.51 | 22.92 |
| GP | 6 | 0.02 | 0.36 | 0.07 | 0.04 | 21.80 | 22.98 | 22.39 |
| Polynomial | 6 | 0.01 | 0.32 | 0.05 | 0.03 | 20.34 | 21.42 | 20.89 |
| Linear + Prev. | 6 | 0.01 | 0.28 | 0.06 | 0.03 | 18.44 | 19.44 | 18.94 |
| Linear | 6 | 0.03 | 0.57 | 0.09 | 0.04 | 15.90 | 16.78 | 16.35 |
| LoT | 7 | 0.02 | 0.30 | 0.07 | 0.03 | 25.84 | 27.16 | 26.50 |
| Comp. GP | 7 | 0.02 | 0.31 | 0.07 | 0.03 | 24.45 | 25.70 | 25.08 |
| GP | 7 | 0.02 | 0.30 | 0.07 | 0.03 | 23.90 | 25.15 | 24.51 |
| Polynomial | 7 | 0.01 | 0.27 | 0.04 | 0.02 | 21.67 | 22.76 | 22.22 |
| Linear + Prev. | 7 | 0.01 | 0.24 | 0.05 | 0.02 | 19.99 | 21.04 | 20.51 |
| Linear | 7 | 0.02 | 0.59 | 0.09 | 0.03 | 16.65 | 17.54 | 17.10 |

Table S3: Children: Mean model parameters and LL bootstrapped means and 95% upper and lower bounds, by age group. LLs averaged across participants within each sequence, and then across sequences.

| | $\sigma_m$ | $\theta_P$ | $\sigma_P$ | $\theta_R$ |
|---|---|---|---|---|
| Monkeys | 0.03 | 0.28 | 0.06 | 0.11 |
| 3 y/o | 0.04 | 0.37 | 0.30 | 0.29 |
| 4 y/o | 0.03 | 0.43 | 0.05 | 0.13 |
| 5 y/o | 0.02 | 0.41 | 0.07 | 0.06 |
| 6 y/o | 0.02 | 0.34 | 0.07 | 0.04 |
| 7 y/o | 0.02 | 0.30 | 0.07 | 0.03 |
| Adults | 0.01 | 0.22 | 0.08 | 0.00 |

Table S4: Mean model parameters for the best fitting model for each participant, separated by group.

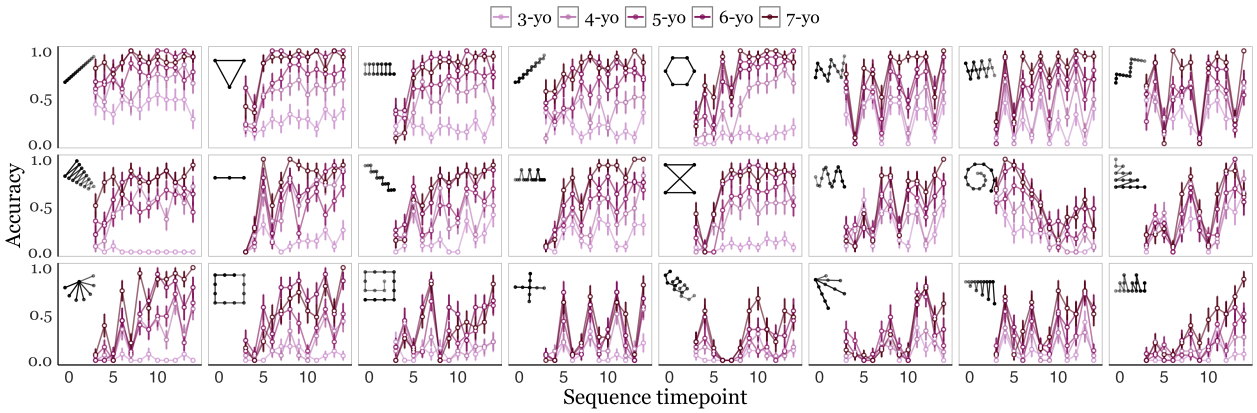


Figure S6: Average accuracy across different sequences (facets) for children split by age (youngest in pink to oldest in dark red).

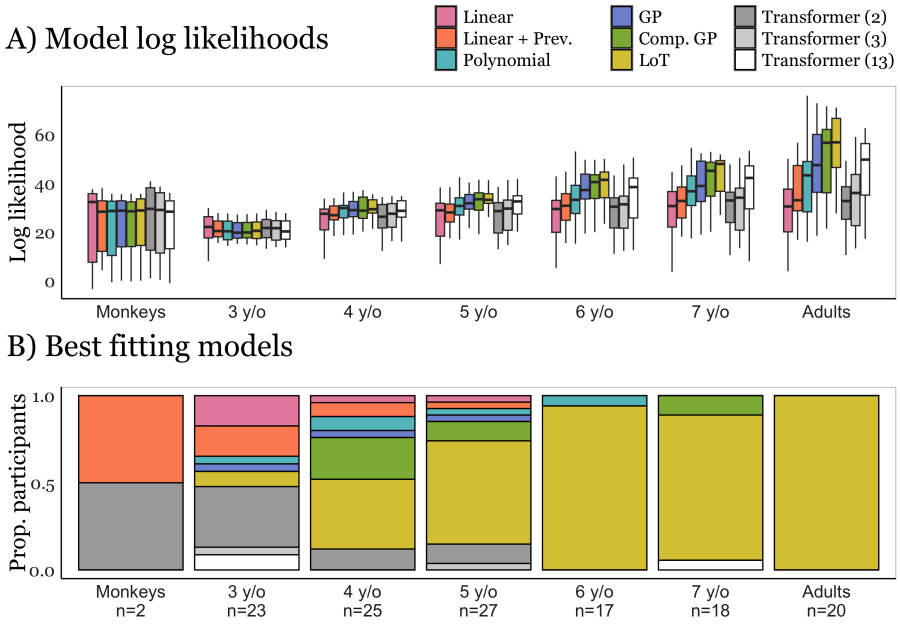


Figure S7: Comparison of learning models including transformer models with context lengths 2, 3, and 13 (full model). Likelihoods are computed based on the 21 test patterns shown to adults, children, and monkeys. (A) Distribution of log likehoods (y-axis) of data by sequence in each group (x-axis). Log likelihoods are averaged across participants (and trials, for monkeys) within each sequence. (C) Proportion of participants best fit by each of the learning models (y-axis) by group (x-axis). A participant is best fit by a model if the likelihood of their data is highest under that model.