Geometric Concept Acquisition in a Dueling Deep Q-Network

Alex Kuefler (akuefler@stanford.edu)
Symbolic Systems Program

Mykel J. Kochenderfer (mykel@stanford.edu)
Department of Aeronautics and Astronautics

James L. McClelland (jlmcc@stanford.edu)
Department of Psychology
Stanford University, Stanford, CA 94305 USA

Abstract

Explaining how intelligent systems come to embody knowledge of deductive concepts through inductive learning is a fundamental challenge of both cognitive science and artificial intelligence. We address this challenge by exploring how a deep reinforcement learning agent, occupying a setting similar to those encountered by early-stage mathematical concept learners, comes to represent ideas such as rotation and translation. We first train a Dueling Deep Q-Network on a shape sorting task requiring implicit knowledge of geometric properties, then we query this network with classification and preference selection tasks. We demonstrate that scalar reinforcement provides sufficient signal to learn representations of shape categories. After training, the model shows a preference for more symmetric shapes, which it can sort more quickly than less symmetric shapes, supporting the view symmetry preferences may be acquired from goal-directed experience.

Introduction

Mathematical concepts are formally definable and may be deduced from of axioms. In contrast, most human mental representations, such as visual categories and language concepts, resist precise definitions and only come to be known after considerable inductive experience. Similarly, deep neural networks have achieved human-level or state-of-the-art performance on tasks such as object recognition (He, Zhang, Ren, & Sun, 2016), game playing (Mnih et al., 2015; Silver et al., 2016), and speech generation (van den Oord et al., 2016) by learning distributed (rather than symbolic) representations through inductive (rather than deductive) training. The empirical success of human learners and artificial neural networks contrasts sharply with the description of mathematical concepts as abstract, formal, and universal ideals.

A growing literature argues that the developmental details of embodied agents are not mere nuisance variables associated with the learning and deployment of mathematical concepts, but are necessary tools in facilitating cognition. For instance, some work suggests that the use of hand gestures aids learning by grounding the meaning of abstract principles such as continuity and magnitude in the willful motions of the body (Goldin-Meadow, Cook, & Mitchell, 2009; Marghetis & Núnez, 2013; Marghetis, Núñez, & Bergen, 2014). The tuning of low-level responses of the visual system has also been associated with algebraic expertise. Marghetis, Landy, and Goldstone (2016) argue that a process of "regimented perception", implemented by object-based attention, lets viewers parse algebraic notation in such a way that makes salient its hierarchical structure, thus offloading much of the work of calculation from high-level cognition to perception. Taken together, this work suggests that for all its formality, mathematics is still an evolutionarily recent phenomenon, and in its comprehension, one marshals bodily and neural resources adapted for other purposes.

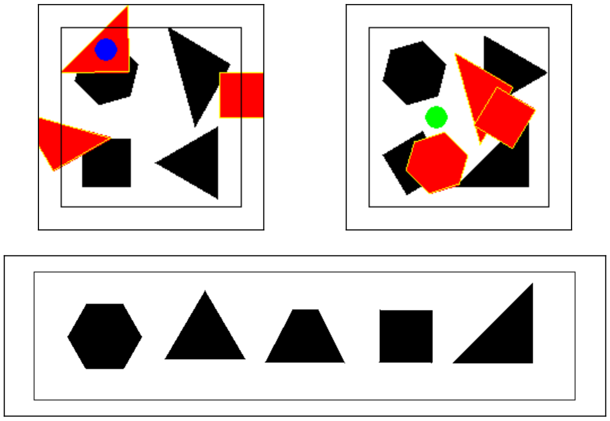

Figure 1: The top panels show example frames from the shape sorting environment. The blue cursor indicates grabbing, and the green cursor is not grabbing. The bottom panel shows examples of each shape in isolation: Hexagon (Hex.), Equilateral Triangle (E. Tri.), Trapezoid (Trap.), Square, and Right Triangle (R. Tri.).

This account dovetails nicely with the Parallel Distributed Processing approach (Rogers & McClelland, 2014), or more recently, advances in deep learning. Just as embodied cognition challenges the primacy of symbolic representations in mental processes, deep learning has been used to overcome a number of problems once thought solvable only through the manipulation of compositional tokens or formal logic. For example, the "Differentiable Neural Computer" of Graves et al. (2016) can answer queries involving textual and hierarchical reasoning, relying solely on a learned, neural memory device. Furthermore, these models learn "end-to-end", adjusting connection strengths between stimulus-facing neurons based on error signals propagated down from higher level decisions (Mnih et al., 2015), much as Marghetis et al. (2016) argue that pursuing algebraic skills re-tunes the visual system. Despite these connections to the embodied approach in mathematical cognition, we are not aware of major work exploring how deep neural networks capture mathematical, and in particular geometric intuitions through a goal-driven, learning process of perception and actuation.

This paper provides a proof of concept and an exploratory analysis. We demonstrate that a domain-agnostic learning algorithm is able to represent geometric concepts that are only implicitly coded in its training task's structure. In particular, we develop a simulated shape sorting toy similar to those enjoyed by young children. Such a learning environment tasks an agent to carry out sequences of actions that require knowledge about some geometric properties (whether implicit or explicit) to maximize reward. We train an agent to expertise in this task, then evaluate its learned behavior and representation of shape categories.

Our primary contribution has been to show that reinforcement learning (RL) is sufficient to train a convolutional neural network agent to perform a shape sorting task expertly. This training leads to the development of shape-specific representations at the top convolutional layer, which feeds into network layers that compute value. We also found that our model exhibited a preference for objects with higher orders of symmetry, supporting the view that experience, rather than an innate symmetry bias, may be the basis for similar similarity preferences in both human infant and primate studies (Bornstein, Ferdinandsen, Gross, 1981; McMahon Olson, 2007). We also make available the source code for the pixels-to-actions shape-sorting task described in this paper.

Approach

Our goal is to demonstrate that deep learning may provide a computational paradigm for building on psychological theory and generating new hypotheses about geometric concept acquisition. Blocks worlds have previously been used to study and model intuitive physics (Hamrick, Battaglia, & Tenenbaum, 2011; Zhang, Wu, Zhang, Freeman, & Tenenbaum, 2016). Although such environments feature a finite set of discrete entities adhering to rules of interaction, their broad properties and questions of investigation tend to differ. For instance, these environments simulate physical properties, like velocity. In contrast, we seek to understand abstract properties, such as shape and symmetry, and we thus introduce an environment with few physical constraints. As such, the experimental setup can be decomposed into an environment and learning agent, which we represent with a neural network.

Environment

During the initial training stage of our neural network, the model interacts with a simulated shape sorting toy (see Figure 1), which may be interpreted as a finite horizon Markov Decision Process (MDP) with deterministic transitions and high-dimensional states (Kochenderfer & Reynolds, 2015). Environment interactions are divided into trials, which consist of at most 500 timesteps. At timestep t, the environment emits an image $s_t \in \mathbb{R}^{84\times84}$ depicting some combination of two types of objects: *blocks* and *holes*. Every object in s_t is characterized by an orientation and a position vector, which remain fixed for holes, but are subject to change for blocks. Each object is also characterized by a convex, 2D shape drawn from the set $\mathcal{X}$, which includes squares, trapezoids, equilateral triangles, right triangles, and hexagons. Once a shape is chosen for an object, it is held constant throughout the course of a trial. The image s_t also includes a cursor used by an agent to manipulate the position and orientation of blocks.

The initial frame s_0 includes three blocks whose shape assignments $\mathcal{X}_b$ are drawn uniformly with replacement from $\mathcal{X}$ with randomized positions and orientations. Four holes with random orientations are also given shape assignments $\mathcal{X}_h$ drawn without replacement from $\mathcal{X}$. A constraint $\mathcal{X}_b \subset \mathcal{X}_h$ ensures that no block will be generated without a corresponding hole. The positions for holes are also randomized at the beginning of each trial, but only drawn from four possible, equidistant locations. This second constraint ensures that holes never overlap. In contrast, blocks may overlap (sometimes completely) but are manipulable, and can be disentangled by an agent.

Given s_t, an agent responds with action a_t in $\mathcal{A}$, which includes: up, down, left, right movements, toggle grab, rotate clockwise or counterclockwise, or wait. Rotations are 30° and a single cardinal movement covers 10% of the height or width of s_t. If the grab is active, blocks will "stick" to the cursor, changing position and orientation with the cursor. If the grab is inactive, movement and rotation actions do not influence the blocks.

The environment uses a reward function that assigns a small reward when the cursor grabs a block $r_1 = 0.001$, a small penalty $-r_1$ when the cursor contacts the border of the screen, a large reward $r_2 = 1.0$ when the cursor fits a block to a corresponding hole or completes a trial. A fit occurs when the cursor "releases" a block over a hole, and the block's vertex set is contained by the hole's vertex set. If a fit occurs, the block disappears and will not return for the rest of the trial.

Agent

We desire a model of a learning agent that is (1) *Deep*, or capable of expressing multiple, hierarchical representations that could feasibly embody geometric invariants, given raw pixels, (2) *Psychologically plausible*, or sufficiently similar to animal decision making to suggest research directions for cognitive science, and (3) *Powerful* enough to solve the nontrivial MDP described in the previous section. The Deep Q-Network (DQN) (Mnih et al., 2015) is an attractive option that can accommodate these considerations.

DQN has attained state-of-the-art results on similar tasks that include discrete action spaces and high-dimensional state spaces. Sharing many architectural properties with convolutional neural networks, it learns a succession of hidden rep-

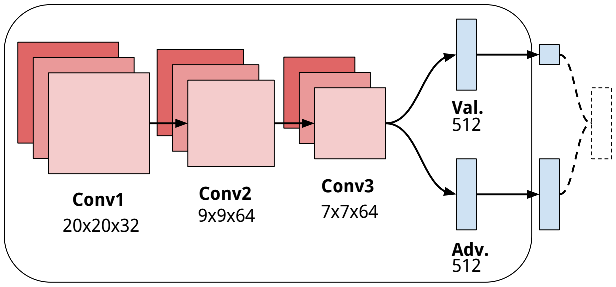

Figure 2: The Dueling Deep-Q Network architecture and the dimensionality of each layer. Boxed layers are involved in representation, output layers are involved in actuation. Dotted lines denote scalar-vector broadcasting that merge value and advantage streams. We use the same filter sizes and strides as Wang et al. (2016)

resentations that can be visualized and interpreted as an abstraction hierarchy (Zeiler & Fergus, 2014). Furthermore, DQN features some desiderata as a model of animal decision making. A prioritized replay pool has been compared to hippocampal learning mechanisms (McClelland, McNaughton, & O'Reilly, 1995), and the architecture is trained using temporal difference (TD) learning, which has been shown to underpin some forms of animal learning (Shah, 2012).

TD learning is here accomplished by minimizing the adaptive loss function

$$L_i(\theta_i) = \mathbb{E}_{(s,a,r,s')\sim D}\left[(y_i - Q(s,a;\theta_i))^2\right] \tag{1}$$

with target value

$$y_i = r + \gamma \underset{a'}{max} Q(s', a'; \theta_i{}^{-}) \tag{2}$$

State-action-reward sequences (s,a,r,s') observed during environment interactions are drawn from a replay buffer D and used as training samples. $Q(s,a;\theta_i)$ represents the sum of discounted future rewards if action a is taken from state s and is estimated at epoch i by a DQN parameterized by θ_i. Updating the model using estimates from a target network parameterized by $\theta_i{}^{-}$ has been shown to improve the stability of training (Mnih et al., 2015; Wang et al., 2016). A policy can then be induced from the DQN by selecting actions maximizing $Q(s,a;\theta_i)$ with probability ε and otherwise selecting exploratory, random actions.

In this work, $Q(s,a;\theta_i)$ is represented by a Dueling Deep Q-Network (DDQN) which is subject to the same TD learning paradigm as DQN, but features a different architecture (Wang et al., 2016), shown in Figure 2.[1] DDQN follows its convolutional layers with two disjoint, fully-connected streams that represent the scalar value of a state $V(s)$ and the advantage $A(s,a) = Q(s,a) - V(s)$ of a state-action pair separately. These representations are merged with the broadcasting rule

$$\begin{aligned} Q(s,a;\theta,\alpha,\beta) = \\ V(s;\theta,\beta) + A(s,a;\theta,\alpha) - \frac{1}{|\mathcal{A}|}\sum_{a'} A(s,a';\theta,\alpha) \end{aligned} \tag{3}$$

where θ, α, and β parameterize the convolutional, advantage, and value sub-networks respectively. DDQN has been shown to improve the state-of-the-art beyond the performance of DQN and has some favorable properties as a neurobiological model, as it extends to deep neural networks the advantage learning paradigm, which has been shown to correlate with striatal neural activity during instrumental learning tasks (O'Doherty et al., 2004).

Learned Behavior

We trained our agent to complete the shape sorting task over the course of one week on a single GeForce GTX 980 graphics processing unit. It completed approximately 480,000 trials consisting of at most 500 timesteps each. After training, the agent completed two experimental tasks consisting of 50,000 trials. Although the agent trained using ε-greedy exploration with $\varepsilon = 0.1$, testing tasks were performed using a pure greedy policy. On both experimental tasks, we found that the agent adopted the strategy of performing translation actions early in the trial, followed by rotation actions once the block was in place over the correct hole.

Single Block Performance. One block per trial was drawn from $\mathcal{X}_b$ and initialized at a random position with four holes from which to choose. The agent's cursor was initialized at the center of the screen. Each trial ended when the agent fit the block to the appropriate hole, attempted to fit the block to an incorrect hole, or exceeded 500 timesteps. Incorrect fits and time outs together accounted for less than 4% of the total number of trials, with the vast majority of failed trials resulting from time outs.

Table 1 demonstrates the agent's efficiency at the task on the trials it successfully completed, in addition to the estimated value computed by the network upon first grabbing a block. Although the agent performed nearly optimally on all shapes, we found that the network assigned higher estimated value for shapes with greater symmetry (which also corresponds the minimum number of steps needed to fit the block to a hole). However, although the right triangle and trapezoid share the same symmetry order, the trapezoid is assigned slightly higher value.

Shape Preference. Observing that our model estimated higher value for some shapes over others, we tested to see whether the agent demonstrated preferences in a two-alternative forced choice. In this experiment, two blocks belonging to different shape categories were generated and placed equidistantly to corresponding holes. Trials terminated when the agent selected a "winner" by releasing a held

[1] Our implementation adapts source code from https://github.com/devsisters/DQN-tensorflow

Table 1: Agent's performance on single-block trials, including value estimated by layer *Val.*, and average number of steps needed to complete the trial against average number of steps actually taken by the agent, per each shape category.

Shape	Value	Min. Steps	Act. Steps	Ratio
Hex.	0.70	10.74 ±(3.7)	11.83 ±(4.0)	0.91
Square	0.67	11.37 ±(3.8)	12.57 ±(4.1)	0.91
E. Tri.	0.64	11.76 ±(3.9)	12.90 ±(4.1)	0.91
Trap.	0.57	13.73 ±(4.2)	15.06 ±(4.7)	0.91
R. Tri.	0.55	13.66 ±(4.2)	15.38 ±(4.8)	0.89

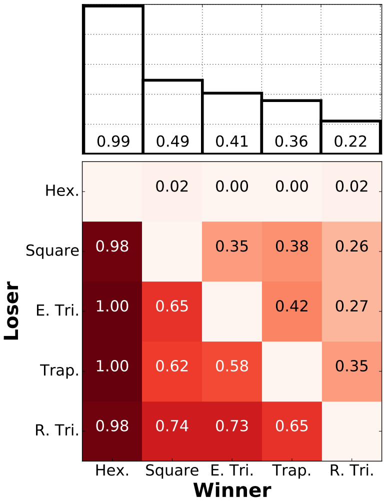

Figure 3: Each cell displays the fraction of choices between two shapes in which the shape on the x-axis was chosen. Marginal probabilities of choosing a given shape are shown above.

block over a corresponding hole. We observed a slight bias such that the policy selected the block appearing on the right hand side of the screen 58.65% of the time, but controlled for this effect by randomizing block position each trial.

The results shown in Figure 3 accord with the findings from the single block experiment. Blocks appear to be preferred on the basis of the number of steps needed to achieve a fit, which is in turn determined by their symmetry order. However, the trapezoid is again preferred to the right triangle, despite the fact that both blocks are equally symmetric. We explore a possible explanation for this result in the next section.

Learned Representations

To gain insight on the agent's elicited behavior, we treat the network as a feature extractor and use classification techniques to explore how it internally represents shape categories in different layers. Within the context of computational neuroscience, linear classifiers have been used to decode information about categorical stimuli from neural responses (Naselaris, Kay, Nishimoto, & Gallant, 2011). We adopt a similar approach. Intuitively, because neural networks are universal function approximators (Hornik, Stinchcombe, & White, 1989), the activation vectors of a well-tuned network corresponding to different categories should be discriminable up to a linear transformation. As such, we assess the classification accuracy of a Support Vector Machine (SVM) trained on encodings from different layers of the DDQN.

Our SVM implementation comes from the open source library, *scikit-learn* (Buitinck et al., 2013) and makes use of a linear kernel $K(x,x') = x^{\intercal}x'$. Multiway classification is achieved using a "one-versus-all" scheme, such that for n classes, n separate binary classifiers are trained to discriminate its corresponding class from examples belonging to other categories. The final classification is made by the model that makes its predictions with the largest margin.

Dataset

One might argue that shape representations in the network depend heavily on scene context. For example, when a scene contains multiple blocks, it may not be useful to encode any information about shape identity until the cursor has taken hold of a single block, as only then must it make a decision contingent on the identity of the shape. To test this hypothesis, we also repeat the discrimination experiment for conditions involving an absent cursor, and conditions in which the cursor is visibly grabbing the block.

We generated our training and validation image sets by enumerating all the possible positions and orientations for a single block, subject to the environment's translation and rotation step sizes, and excluding duplicate shape orientations resulting from symmetry. Each combination was used to produce a set of 100 unique examples by randomly permuting the background holes. The resulting data set consisted of 81,000 images, which we shuffled and partitioned into training and validation sets using a 25-75 split. The data sets including a cursor were generated by the same process.

Each frame s_i was replicated four times, producing an 84 × 84 × 4 tensor, which was then encoded as an activation vector ${z_i}^j$ at the jth layer of the DDQN. Because the size of the layers differ greatly, we use principal component analysis (PCA) to enforce that all activation vectors have 300 dimensions. To establish a classification baseline, we repeat the same analysis on encodings produced by an untrained, randomly initialized neural network with the same architecture. We do not standardize encodings prior to classification or dimensionality reduction, as all input variables are ReLU-gated activations and are thus measured on the same scale.

Results

Results shown in Figure 4 support the view that shape information is scene dependent, albeit slightly. At the level of *Conv3* and beyond, classification accuracy was consistently greater by about 3% when the cursor was grabbing the block. Figure 5 visualizes the activation vectors in both conditions, and suggests that the categories become better separated during a grab. Interestingly, these results contradict the view

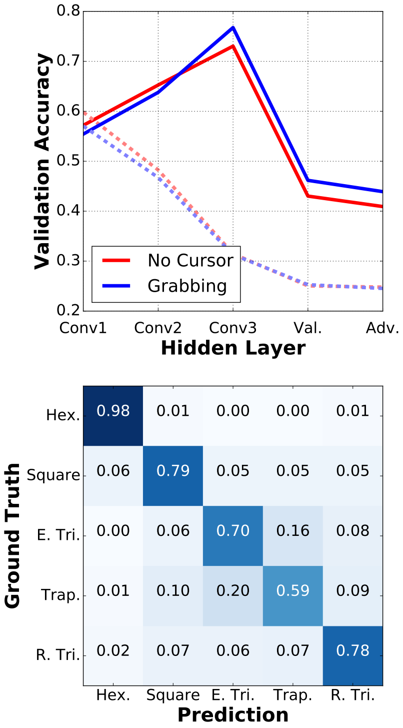

Figure 4: Average validation accuracy of the SVM. The top plot shows accuracy per encoding layer, with baseline accuracy on encodings produced by a DDQN with random weights. The bottom plot shows classification confusion on the “Grabbing” condition at *Conv3* in the trained network.

that shape information plays a major role in the directions of greatest variance in higher layers. Whereas the classifiers achieve above 70% accuracy on frames encoded by *Conv3*, upstream encodings from *Adv.* and *Val.*, were discriminable only about 45% of the time, dropping even below *Conv1* and *Conv2*.

The most significant misclassifications are shown to be between the equilateral triangle and trapezoid, whereas the most easily discriminable shapes were the hexagon and square. These misclassifications may explain the preference for trapezoid blocks over right triangles, despite their similar orders of symmetry, as the network seems to “confuse” trapezoids with the easier triangle shape, whereas the right triangle is easy to classify as a difficult block.

Conclusions

Learning mechanisms and computational principles underlying mathematical cognition are not well understood. However, deep neural networks provide opportunities for exploring this direction of inquiry. We hypothesized that reinforcement learning, which incorporates active probing of an environment, serves as a sufficient training signal for learning many geometric properties embodied implicitly in an interactive shape sorting task.

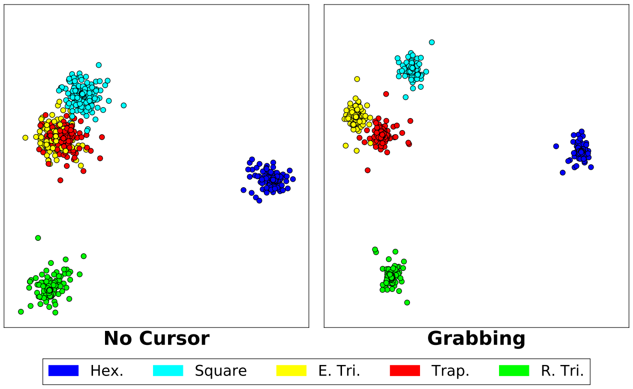

Figure 5: A sample of 2,500 encodings extracted from *Conv3*. Linear Discriminant Analysis was used to project the 300-dimensional vectors onto a subspace in which the classes are well-separated. The separation is more clear when the cursor is grabbing the block.

On a representational level, we showed that shape identity can be recovered from the network’s hidden layers using linear classifiers and that this information is more strongly encoded in later convolutional layers than in the final hidden layers needed to valuate states and possible actions. Recent work suggests an analogy between the hierarchical structure of convolutional neural networks and the hierarchical structure of the visual system (Yamins et al., 2014). If this analogy is to be taken seriously, we should predict that despite their simplicity, geometric forms may find representation in later visual areas when tied to one’s goals, as when playing with a shape sorter or interpreting mathematical diagrams on an exam.

On a behavioral level, we found also that a preference for symmetric blocks emerged as a consequence of their ease of fitting. Past work indicates that a preference for symmetric shapes exists among both monkeys (McMahon & Olson, 2007) and human beings, but emerges only later in development (Bornstein, Ferdinandsen, & Gross, 1981). Bornstein et al. (1981) in particular suggest that this preference, which favors vertical over horizonal symmetry, arises not from the information redundancy present in such figures, but from their adaptive value. Symmetrical figures tend to be animate, and can thus act as adversaries or allies in an organism’s pursuit of goals. In our domain as well, we found that a block’s degree of symmetry influences a learning agent’s discounted sum of expected rewards. This result further supports the adaptive view of symmetry preference over the redundancy view, and implies a number of testable predictions for future work.

Follow up studies should investigate whether children trained to play with shape sorters prefer different blocks on the basis of their symmetry properties, and if so, if this preference can be modulated by increasing the stakes of the task, whether by providing greater rewards or less time to react. Further simulation work should also explore the relationship between symmetry and visual similarity. Despite both being completely asymmetric, we found that trapezoids were strongly preferred to right triangles, as they are visually closer to equilateral triangles. An experiment in which "adversarial" shapes attempt to look easier to fit than they really are may demonstrate how the constraints of perception (imposed by the visual similarity between blocks) and the constraints of actuation (imposed by the reward signal, or task) must be mutually satisfied in embodied, geometric concept acquisition.

The code associated with this paper can be found at `https://github.com/akuefler/shape-sorting`.

Acknowledgements

We would like to thank Steven Hansen for useful discussions.

References

Bornstein, M. H., Ferdinandsen, K., & Gross, C. G. (1981). Perception of symmetry in infancy. *Developmental Psychology*, *17*(1), 82.

Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., … Varoquaux, G. (2013). API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning* (pp. 108–122).

Goldin-Meadow, S., Cook, S. W., & Mitchell, Z. A. (2009). Gesturing gives children new ideas about math. *Psychological Science*, *20*(3), 267–272.

Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwińska, A., … Hassabis, D. (2016). Hybrid computing using a neural network with dynamic external memory. *Nature*, *538*(7626), 471–476.

Hamrick, J., Battaglia, P., & Tenenbaum, J. B. (2011). Internal physics models guide probabilistic judgments about object dynamics. In *Proceedings of the 33rd Annual Conference of the Cognitive Science Society* (pp. 1545–1550).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770–778).

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366.

Kochenderfer, M. J., & Reynolds, H. J. D. (2015). *Decision making under uncertainty: Theory and application*. MIT press.

Marghetis, T., Landy, D., & Goldstone, R. L. (2016). Mastering algebra retrains the visual system to perceive hierarchical structure in equations. *Cognitive Research: Principles and Implications*, *1*(1), 25.

Marghetis, T., & Núnez, R. (2013). The motion behind the symbols: A vital role for dynamism in the conceptualization of limits and continuity in expert mathematics. *Topics in Cognitive Science*, *5*(2), 299–316.

Marghetis, T., Núñez, R., & Bergen, B. K. (2014). Doing arithmetic by hand: Hand movements during exact arithmetic reveal systematic, dynamic spatial processing. *The Quarterly Journal of Experimental Psychology*, *67*(8), 1579–1596.

McClelland, J. L., McNaughton, B. L., & O'Reilly, R. C. (1995). Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, *102*(3), 419.

McMahon, D. B., & Olson, C. R. (2007). Repetition suppression in monkey inferotemporal cortex: relation to behavioral priming. *Journal of Neurophysiology*, *97*(5), 3532–3543.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., … Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529–533.

Naselaris, T., Kay, K. N., Nishimoto, S., & Gallant, J. L. (2011). Encoding and decoding in fMRI. *Neuroimage*, *56*(2), 400–410.

O'Doherty, J., Dayan, P., Schultz, J., Deichmann, R., Friston, K., & Dolan, R. J. (2004). Dissociable roles of ventral and dorsal striatum in instrumental conditioning. *Science*, *304*(5669), 452–454.

Rogers, T. T., & McClelland, J. L. (2014). Parallel distributed processing at 25: Further explorations in the microstructure of cognition. *Cognitive Science*, *38*(6), 1024–1077.

Shah, A. (2012). Psychological and neuroscientific connections with reinforcement learning. In M. Wiering & M. van Otterlo (Eds.), *Reinforcement learning: State-of-the-art* (pp. 507–537). Springer.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., … Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, *529*(7587), 484–489.

van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., … Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. *CoRR abs/1609.03499*.

Wang, Z., Schaul, T., Hessel, M., van Hasselt, H., Lactot, M., & de Freitas, N. (2016). Dueling network architectures for deep reinforcement learning. In *International Conference on Machine Learning (ICML)* (p. 1995-2003).

Yamins, D. L., Hong, H., Cadieu, C. F., Solomon, E. A., Seibert, D., & DiCarlo, J. J. (2014). Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, *111*(23), 8619–8624.

Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. In *European Conference on Computer Vision* (pp. 818–833).

Zhang, R., Wu, J., Zhang, C., Freeman, W. T., & Tenenbaum, J. B. (2016). A comparative evaluation of approximate probabilistic simulation and deep neural networks as accounts of human physical scene understanding. *arXiv preprint arXiv:1605.01138*.