UC Merced

Proceedings of the Annual Meeting of the Cognitive Science Society

Title

Learning, Development, and Nativism: Connectionist Implications

Permalink

Journal

Authors

Sirois, Sylvain
Shultz, Thomas R.

Publication Date

Peer reviewed

Learning, Development, and Nativism: Connectionist Implications

Sylvain Sirois (sirois@psych.mcgill.ca)
Department of Psychology, McGill University
1205 Penfield Avenue
Montréal, Qc H3A 1B1 CANADA

Thomas R. Shultz (shultz@psych.mcgill.ca)
Department of Psychology, McGill University
1205 Penfield Avenue
Montréal, Qc H3A 1B1 CANADA

Abstract

Feedforward neural network models of cognitive development are reviewed within the framework of a functional distinction between learning and development. This analysis suggests that static architecture networks implement a learning theory, whereas generative architecture networks combine learning and development. Both types of networks are then evaluated in terms of genetic costs. Within a levels-of-innateness framework, generative architectures are viewed as more plausible than static ones. Static architecture networks appear to implement a form of nativistic elicitation.

Introduction

Feedforward neural networks process information through brain-inspired principles: excitatory and inhibitory stimulation, activation summation, activation threshold, unit activation, and massively parallel and distributed processing. Although much simpler than neural tissue found in most (if not all) animals, these networks can process complex information and provide an alternative framework to rule-based symbolic approaches to the study of cognition. The fact that such networks can learn also provides researchers with powerful tools for the study of human learning and development (Elman, Bates, Johnson, Karmiloff-Smith, Parisi, & Plunkett, 1996; Shultz & Mareschal, 1997). This paper is an expansion on work by Quartz (1993), who studied assumptions of PDP models within the framework of Valiant's probably approximately correct (PAC) model of learning. The first section proposes a formal distinction between learning and development, which serves to evaluate the theoretical implications of developmental work using different neural network architectures. The second section evaluates underlying assumptions of different types of neural network algorithms within the levels-of-innateness framework outlined by Elman and his colleagues (1996). Each of the two arguments that can be formulated to support static neural networks as models of human cognition are inconsistent with the neurological evidence. The discussion stresses that static networks implement a form of nativistic elicitation, as suggested by their inability to escape Fodor's paradox. Overall, static networks do not seem to be good candidates for modeling cognitive development.

A Distinction Between Learning and Development

In order to evaluate the possible contribution of neural network modeling to our understanding of human cognitive development, it is important to distinguish it from learning. As Carey (1985) suggested, attempts to differentiate between learning and development often confound two distinctions: whether knowledge acquisition requires restructuring, and whether changes are domain-general or domain-specific. This paper focuses on the former.

How knowledge acquisition may or may not require restructuring is a question that was not directly addressed by Elman and his colleagues in their landmark book on the connectionist perspective of cognitive development (1996). Although they do provide justification for the study of development above and beyond learning, they do not commit to a formal distinction between these two processes. Because of the substantial impact their book has on the study of cognitive development, a clarification appears timely as it bears on the theoretical implications of different types of neural network research.

In this paper, the following functional distinction is proposed. Learning is defined as a change *within* an existing processing structure in order to adapt to information from the environment. This broad description is compatible with general statements about learning, such as found in nativistic accounts (e.g., Fodor, 1980) and developmental models (e.g., Carey 1985; Piaget, 1982). In contrast, development is defined as change *of* an existing structure to enable more complex and adaptive cognitive activity. This general statement highlights the key idea underlying most theories of development; that is, a qualitative change in the structure supporting cognition. Such a general definition of a developmental mechanism can be found explicitly or implicitly in Piaget's (1982) abstraction, Karmiloff-Smith's (1992) representational redescription, and Carey's (1985) conceptual change, for example. These functional definitions of learning and development allow a distinction between the two processes, removing overlap between them and constraining their individual contribution to cognitive change. Learning is viewed as parameter adjustment within a given structure; development as change of structure within

which learning (as well as other cognitive processes) takes place.

The outlined distinction between learning and development is also useful in the evaluation of neural network models of cognitive development. A simple example that illustrates this point is the XOR Boolean operator. Implemented as a function, XOR takes two arguments that can be either true or false and returns *true* if one and only one argument is true, otherwise it returns *false*. Figure 1 presents two different networks that solve the XOR problem. Given that the network on the left is a typical static feedforward network and the one on the right is a generative cascade-correlation network, how they achieve these indistinguishable solutions at the output level from initially random weights requires two different stories.

At the onset of training, the static network of Figure 1 has the representational power to solve the problem. Within the multidimensional weight space determined by its topology, there is a region that will produce the correct output. Only quantitative changes are required to move the network from its initially random position in weight space to the region that solves the problem. Because the learning algorithm capitalizes on the nonlinear properties of hidden and output units, these gradual weight changes will not be linear (i.e., the delta value for a weight is a function of the receiving unit's nonlinear activation, which changes from epoch to epoch). Although these nonlinear changes qualify as learning, they do not qualify as development, because only parameters of the current structure are changed.

The learning history in the cascade-correlation network from Figure 1 would be different. The initial two-layer architecture of the network does not allow it to properly solve the XOR problem. There is no region in its weight space that will produce the correct output. Therefore error reduction in output training will stagnate above a

satisfactory level (typically, the network will reduce all weights to near-zero values in order to minimize error on all patterns). This stagnation in error reduction spurs the recruitment of a hidden unit that is then used for further output training. This new unit increases the dimensionality of the weight space, in which a solution region now exists. For this network, both learning and development combine to achieve a solution to the XOR problem. Learning (i.e., parameter adjustment) within the initial structure is unsuccessful, prompting a modification of the architecture. This qualifies as development because the structure within which learning takes place is changed. Further learning within the new structure finally solves the problem.

We are not arguing that a generic XOR function is a human developmental problem. However, this basic example highlights how static and generative networks tell a different story about learning and development for nonlinear problems. Namely, that the former network type is a learning model, and that the latter is a developmental model (that also incorporates learning).

Elman and his colleagues (1996), by not providing a clear distinction between learning and development, make an equivocal statement about connectionism as a model of development. Even though they make a good case that there is development and not just learning in human cognitive change, the simulations they report consist of learning models that capture developmental data. A good example is their discussion of a balance-scale model by McClelland (1989).

The balance-scale task consists of a beam with a series of unit-spaced pegs on both of its sides, centered on a fulcrum. On a given trial, a number of unit weights are placed on one peg on each side. The participant is required to predict which side of the beam (if any) would go down, provided that the balance would be free to move. Robust and replicable developmental effects have been observed in children of various ages performing this task (Siegler, 1981).

Initially, younger children perform at chance level (stage 0). As they grow older, they begin to use weight information in their predictions (stage 1). They predict that one side will go down if it has more weight. At the next stage, they begin to use distance information, only if there is equal weight on both arms of the scale (stage 2). Stage 3 children use both types of information yet fail to integrate them, so when weight and distance conflict, they perform at chance level. This is associated with a U-shaped effect on conflict problems in which the side with larger weight would go down. Stage 2 children make a correct prediction on these problems whereas older, stage 3 children perform at chance level. Finally, stage 4 is the level where performance is correct on all problem types. At stage 4, children's answers appear to follow the torque-rule solution of the problem, which states that if the products of weight and distance on each side of the beam are different, the beam will not balance but tip to the side with the larger product.

The balance-scale task provides modelers with a robust target of stage-wise developmental data. Mathematical models based on catastrophe theory maintain that such a
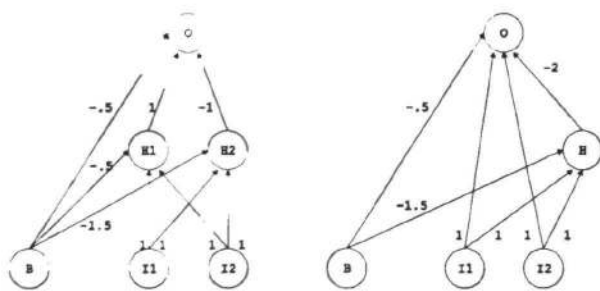

Figure 1: Two different solutions of the XOR problem. The network on the left represents a static network. All the weights and units depicted are there at the onset of training. The solution of the XOR problem required only gradual changes of the initially random weights. The cascade-correlation network on the right differed at the onset of training, with only bias, input, and output units. The recruitment of a hidden unit being necessary at the end of the first output training phase, the network increased its representational power in order to solve the problem. Whereas the output of both networks will be indistinguishable on all four instances of the problem after training, the static network learned and the generative network both learned and developed.

developmental profile is better described as discontinuous (van der Maas & Molenaar, 1992). A neural network that claims to be an adequate model of human development should replicate this ordered progression through the 5 stages of performance. According to Elman and colleagues (1996), this is exactly what was accomplished by McClelland's (1989) model.

McClelland (1989) used static backpropagation networks to model the balance-scale task. His results show that the networks progress through all stages, without reliably settling into stage 4 performance. At the end of training, network behavior is between stage 3 and stage 4 levels of performance. Because not all human adults spontaneously reach stage 4 performance, the results were considered satisfactory (McClelland, 1989).

Irrespective of the fact that some humans do reach stage 4, the networks were viewed as apt developmental models, exhibiting competence acquisition that follows qualitatively distinct stages. However, Raijmakers, van Koten, & Molenaar (1996) suggest that it is the evaluation method and not network learning that is responsible for this stage-wise progression. In McClelland's simulations, network performance was assessed at every training epoch. The performance categories were mutually exclusive, so at any point in training, network performance could be associated with only one stage. Because of the binary decision involved in determining whether a network is at one stage or not, Raijmakers and colleagues (1996) suggest that stage-like discontinuous progression in these networks is an artifact of the evaluation procedure that was applied to the continuous, gradual learning taking place in the networks.

Elman and colleagues did not discuss a cascade-correlation model of the balance-scale task (Shultz, Mareschal, & Schmidt, 1994). This model progresses through all stages in an orderly fashion, and performs at stage 4 by the end of training. Moreover, there are genuine discontinuities in network learning, as it needs to alter its topology to solve the task.

Although it is possible that the discontinuities observed in child performance are due to measurement intervals too broad to assess continuous change taking place at a smaller intervals (Siegler, 1998), this appears unlikely (van der Maas & Molenaar, 1992). If this were the case, the static network might be a better model. What is relevant for this paper, though, is that a learning model may be construed as capturing developmental phenomena if learning and development are not formally distinguished. The ability of a network to mimic developmental data does not, in itself, make it a developmental model. Static backprop models only implement learning, even when one observes nonlinear changes during learning. As we argue in the next section, they also offer a different view of innateness than generative networks.

Elman (1993) reports an interesting simulation using the simple recurrent network architecture (SRN). SRN networks are similar to static feedforward networks, with the addition of a bank of context units. These units take as activation values the activations of the hidden units at one time step, and are fed back to the hidden units at the following time step. Such context units provide the network with a working memory, essential for sequential problems such as language. Elman (1993) found that he could improve the performance of SRN networks on complex problems that they failed to learn by enabling the context units to deal with progressively longer strings. Elman (1993) actually implemented a sort of sequential generative architecture. However, the rest of the static architecture of the SRN is problematic from a nativistic perspective, as is the case for all static models.

Levels of Innateness

One important contribution of the Elman and colleagues' (1996) book is their review of innateness. They identify three levels at which concepts could have an innate basis: representations, architectures, and timing. The focus is not as much towards identifying what is or is not innate, but rather towards defining how things could be innate.

Hardwired concepts or knowledge would be at the representational level. This level of innateness implies that specific synapses in the brain must be designed in order to represent concepts before experience could have shaped such connections. Spelke's (1994) suggestion of innately specified core theories in infants would fall in this category of innateness, for example. So would a language acquisition device for a universal grammar (Chomsky, 1975). For nature to implement such information prior to any experience, a large amount of precisely designed connections need to be made.

Elman and his colleagues reject such a level of cognitive innateness based on two observations: human DNA cannot encode such a large amount of precise information, and the human cortex exhibits a significant equipotentiality that is incompatible with pre-specified representations. Both observations also have implications for the evaluation of neural network models.

Human DNA is found on twenty-three pairs of chromosomes. It is estimated that it can carry up to 10^9 bits of information on base pairs (Elman et al., 1996). This is not enough data to specify the specific location and configuration of each cell in the body (i.e., mosaic development), so nature must rely on heuristics to generate a viable being from minimal information (i.e., regulatory development). Elman and colleagues (1996) provide an extensive review of known mechanisms through which cells organize themselves functionally and spatially through an interaction between their DNA and the environment.

Their conclusion is that the different processes through which nature makes use of minimal DNA to build humans argues against representational innateness. The information in genes operates at a more abstract level than is required for innate representations, so other constraints are suggested to account for species specific stereotypical behavior.

The plasticity or equipotentiality of the cortex also argues against representational innateness (Elman et al., 1996). Compelling evidence from studies of brain rewiring in small mammals is reported to highlight the plasticity of neural tissue, even in species for which behavior is typically considered more rigid (i.e., innately specified). First, redirecting the visual input of mice to the auditory cortex

and vice-versa has the resulting effect that the auditory cortex will process visual signals and the visual cortex will process auditory signals. Moreover, the auditory cortex will develop receptive fields and ocular dominance columns as would normally take place in the visual cortex. Second, transplanting cortical columns from one cortical area to another has a similar effect on the transplanted tissue. Rather than processing the type of information it would have initially, it will process information from its new location and develop to be indistinguishable from its neighbor columns. Both these effects argue against innate constraints on representation, because the cortex will learn to process whatever it is fed and will be strongly influenced by its neighbors.

One justification that is invoked to sustain the idea of innate knowledge in humans is that it would be unreasonable to assume it in all animals but humans (e.g., Karmiloff-Smith, 1992). Quartz and Sejnowski (1997) reviewed the literature on brain development, and concluded that plasticity is most often found in species that are phylogenetically recent and proximal to humans. They suggest that it may be more appropriate to speak of human evolution as moving towards maximal plasticity rather than towards hyperspecialization. Like Elman and colleagues (1996), they suggest that plasticity may be the more adaptive solution, that it is more compact than innately specified knowledge, and that it is sustained by comparative data (Quartz & Sejnowski, 1997).

The next level of innateness is architectural. According to Elman and his colleagues (1996), this is the level where innate constraints on the brain can have a plausible effect on knowledge. Architectural constraints themselves can be divided into three levels. Unit level architectural constraints deal with the specific properties of neurons (e.g., neuron types, response characteristics, type of transmitter). Local architectural constraints, as the name implies, deal with the local organization of neural tissue (e.g., layers, density, degree and nature of connectivity). Finally, global architectural constraints concern the global organization of the local areas. According to the view of the brain as a network of networks, global constraints specify how networks are interconnected.

In these architectural forms of innateness, knowledge is not innate, but the overall structure of the brain constrains how, where, and what information will be processed. This embodies species specific aspects of cognition, without requiring representational innateness. The almost universal specific localization of many important cognitive processes (e.g., Broca's area) is guaranteed by specifying what input is sent to different areas. For Elman et al. (1996), architectural constraints are not only a reasonable way through which genes may constrain cognition: connectionist models implement this level of innateness.

The final level at which something cognitive can be innate is with chronotropic constraints. These affect the timing of maturational events, from cell division in neurogenesis to waves of synaptic growth and pruning, as well as the temporal development of different cortical areas (Elman et al., 1996).

Whereas architectural constraints deal with *what* and *where* information is processed, chronotopic constraints add a *when* dimension to the equation. The order in which information can be processed and integrated over development will have an important impact on the nature of cognitive processes (Elman et al., 1996).

This revised interpretation of innateness leaves one important question unanswered, though: What is implied by different neural network algorithms with respect to innate specification? Elman et al. (1996) took the opposite perspective: What do innate specifications imply? Their answer is that only architectural and chronotopic constraints are reasonable forms of innateness, and consequently so are neural networks with unspecified weights. However, we argue that different architectures have different genetic costs and theoretical implications.

Neural Networks and Innateness

Quartz (1993) has shown that a static network cannot learn what is beyond its representational power (defined by the number of weights, the activation functions of units, and the topology of the network). However, too powerful a network may correctly produce output without having abstracted any relevant information through training. What is implied for the human brain by these results? The exact topology of a network is crucial for its learning behavior. Therefore an important issue, in order to consider static architecture networks as models of human cognition, is sustaining that the brain would have the appropriate topology beforehand as an experience-independent given.

Using static networks as models of human cognitive development would require one of the following two assumptions. The first assumption would be that the large number of neurons in the brain are highly interconnected in such a way that for anything humans need to (and can) learn, the probability of an appropriately connected network existing in advance is extremely high. We call this the *probable-network* assumption. The second assumption is that the brain is provided a priori knowledge of what will have to be learned, and has the appropriately connected networks before any experience, which would be a form of representational innateness (Quartz, 1993). We refer to this as the *knowledgeable-network* assumption, because networks are specified from a priori knowledge about tasks. Let us consider each of these assumptions.

The Probable-Network

There is indeed a very large number of neurons in the brain, estimated at about 10^{11}, and with an average of 10^3 connections per neuron, it is safe to say that it is a powerful computer (Churchland, 1989). But like a supercomputer with an operating system that would allow only single digit arithmetic, such power is useless if not properly wired. According to the probable-network assumption, a great deal of computational power in the human brain is wasted. Only those neurons organized in the appropriate topology for a given task will be of use, the others discarded. Elman and colleagues (1996) refer to the observation that there is an initial proliferation of neurons in young children, followed by sub-

stantial pruning. The problems with this approach are three-fold.

First, the simulations reported by Elman and his colleagues should not be understood as making a one-to-one correspondence between neurons in the brain and units in the networks. Rather, units in neural network models should be viewed as analogous to groups of neurons in the brain (columns, modules, regions...), and weights as pathways. This implies that the unit cost of a given network is greater at the level that implements it, namely neurons. In order for the brain to make sure that there is somewhere a network appropriate for the task at hand, a huge number of neurons must go to waste. Such a costly solution does not appear adaptive (Quartz & Sejnowski, 1997).

Second, given that learning and development occur across the lifespan and that most of the neural pruning takes place in childhood (Elman et al., 1996), the implication is that in order to learn a new task in later life, the brain will have kept the appropriate network from childhood. This would be at best odd, because a) pruning was understood as an experience-driven process by which useless neurons are removed whereas their usefulness has not yet been evaluated, and b) the brain would require some mysterious access to the solutions of yet to be encountered problems to keep suitable solutions for future learning.

Finally, support for the probable-network assumption would go against recent findings in neuroscience research (Quartz & Sejnowski, 1997). It is suggested that the overproduction/pruning model of development is overstated in the literature, and that flexibility in the brain provided by synaptogenesis, the generation of new synapses, has a greater role in cognitive change. Moreover, this flexibility available through synaptogenesis is more often found in species that are close to humans phylogenetically.

For these reasons, static neural networks do not appear a tenable approach to modeling human cognition, unless one commits to the second assumption, that networks are already properly connected.

The Knowledgeable-Network

The problem with this suggestion is that it implies more than architectural innateness, unlike what is suggested by Elman and colleagues (1996). They argue that representations need not be innate because a network with initially random connections will find its way to the appropriate representation through weight adjustment. Representational innateness would imply that the weights, and not just the topology, would be pre-specified. This suggestion masks the fact that in order to generate the appropriate static topology, the brain would still require some a priori representation of the problem it will come to learn, because network topology determines what can be learned, and how. If an appropriate topology is not available beforehand, the organism could fail to learn, or learn inappropriately. Given that the appropriate topology is defined as a function of the problem, networks with initially random weights still imply representational innateness (if only a relaxed version). In which case the networks implement elicitation of knowledge through parameter adjustment. Elicitation is a nativistic synonym of learning, where only parameter values need to be derived through experience, because the organism was provided with the required parameters. So the suggestion the authors raise about the implausibility of genetically specified representations, presented earlier, should be taken a step further and would argue against genetically specified topologies for each problem humans will come to learn.

Conclusion

Overall, static neural networks do not fare well in light of the reasonable objections to some forms of innateness highlighted by Elman and his colleagues (1996). Either they imply a costly and mysterious use of neurons that clashes with data from developmental neuroscience (the probable-network assumption), or they require a disguised form of representational innateness for their implementation (the knowledgeable-network assumption). This is not the case with generative networks such as cascade-correlation. These are cost efficient, because they will develop the architecture required as they learn. Because of this generative property, there is no requirement, through either overproduction of neurons or innately specified architectural maps, for an appropriate topology to be present prior to learning. And generative networks are more consistent with developmental observations in neuroscience (Quartz & Sejnowski, 1997).

As a final note, generative networks can escape the nativistic paradox of development formulated by Fodor (1980), whereas static networks cannot (Mareschal & Shultz, 1996; Shultz & Mareschal, 1997). Fodor's paradox states that a system with a given level of logical power will be unable to generate a logical system at a higher level (Fodor, 1980). This implies that it is impossible to represent something for which one does not already have the representational power. Formulated in the context of computational approaches to learning, this is a strong argument, and it has been used to argue against development and in favor of nativistic ideas.

Cascade-correlation escapes Fodor's paradox through its principled recruitment of additional units during the learning process (Mareschal & Shultz, 1996). Recall the XOR example from the first section. Initially, with its two-layer topology, a cascade-correlation network cannot represent a logical operator of the XOR level. Only linear functions like OR and AND could be learned. XOR, a combination of AND and OR, can only be represented in the network through recruiting an additional, hidden unit. The algorithm does just that when error reduction stagnates, and the new unit is trained to track the network's residual error. The solution is not given to the network, it develops and learns one. Static networks, at the onset of training, must have an appropriate topology in order to succeed. To some extent, the solution is thus built in because networks are powerful enough to represent it. No matter what the initial weight values are, the number and arrangement of these weights determines what the networks can and cannot learn (Quartz, 1993). As such, they fail to escape Fodor's paradox. Moreover, given the implications at the implementational level discussed in the previous section, the learning model implied by static networks may not only

fail to realize development, but may very well succeed at implementing nativism as elicitation. Elicitation implies that mere exposure to stimuli will produce the predetermined behavior through parameter tweaking. A priori topologies do exactly that by constraining the representations of a network.

The ideas presented in this paper are not meant to be definitive. The purpose is to raise awareness to a common confound between learning and development, as well as to the biological implications associated with different neural network models. Because of the immense potential of neural network tools for the study of human development, it would be of great disservice to ignore the basic questions that pertain to theoretical assumptions and implications associated with these models.

Acknowledgments

References

Carey, S. (1985). *Conceptual Change in Childhood*. Cambridge, MA: MIT Press.

Chomsky, N. (1975). *Reflections on Language*. New-York, NY: Parthenon Press.

Churchland, P. M. (1989). Cognitive activity in artificial neural networks. In D.N. Osherson & E.E. Smith (Eds.), *Thinking: An Invitation to Cognitive Science*, Vol. 3 (pp.199-227). Cambridge, MA: MIT Press.

Elman, J. L. (1993). Learning and development in neural networks: The importance of starting small. *Cognition, 48*, 71-99.

Elman, J. L., Bates, E. A., Johnson, M. H., Karmiloff-Smith, A., Parisi, D., & Plunkett, K. (1996). *Rethinking Innateness: A Connectionist Perspective on Development*. Cambridge, MA: MIT Press.

Fodor, J. (1980). Fixation of belief and concept acquisition. In Piatelli-Palmarini (Ed.), *Language and Learning: The Debate Between Chomsky and Piaget* (pp. 143-149). Cambridge, MA: Harvard Press.

Karmiloff-Smith, A. (1992). *Beyond Modularity: A Developmental Perspective on Cognitive Science*. Cambridge, MA: MIT Press / Bradford Books.

Mareschal, D., & Shultz, T. R. (1996). Generative connectionist networks and constructivist cognitive development. *Cognitive Development, 11*, 571-603.

McClelland, J. L. (1989). Parallel distributed processing: Implications for cognition and development. In R.G.M. Morris (Ed.), *Parallel Distributed Processing: Implications for Psychology and Neurobiology* (pp. 8-45). Oxford: Oxford University Press.

Piaget, J. (1982). La psychogenèse des connaissances et sa signification épistémologique. In M. Piattelli-Palmarini (Ed.), *Théories du Language, Théories de l'Apprentissage* (pp. 53-64). Paris: Éditions du Seuil.

Quartz, S. R. (1993) Neural networks, nativism, and the plausibility of constructivism. *Cognition, 48*, 223-242.

Quartz, S. R., & Sejnowski, T. (1997). The neural basis of cognitive development: A constructivist manifesto. *Behavioural and Brain Sciences, 20*, 537-596.

Raijmakers, M. E. J., van Koten, S., & Molenaar, P. C. M. (1996). On the validity of simulating stagewise development by means of PDP networks: Application of catastrophe analysis and an experimental test of rule-like network performance. *Cognitive Science, 20*, 101-136.

Shultz, T. R., & Mareschal, D. (1997). Rethinking innateness, learning, and constructivism: connectionist perspectives on development. *Cognitive Development, 12*, 563-586.

Shultz, T. R., Mareschal, D., & Schmidt, W. C. (1994). Modeling cognitive development on balance scale phenomena. *Machine Learning, 16*, 57-86.

Siegler, R. S. (1981). Developmental sequences between and within concepts. *Monographs of the Society for Research in Child Development, 46* (Whole No. 189).

Spelke, E. S. (1994). Initial knowledge: Six suggestions. *Cognition, 50*, 431-445.

van der Maas, H., & Molenaar, P. (1992). Stagewise cognitive development: An application of catastrophe theory. *Psychological Review, 99*, 395-417.