International Journal of Academic Information Systems Research (IJAISR)
ISSN: 2643-9026
Vol. 4, Issue 9, September – 2020, Pages: 74-81

Modeling Cognitive Development of the Balance Scale Task Using ANN

Yara Essam Al-Atrash, Ahmed Tariq Wishah, Tariq Hosni Abul-Omreen, Samy S. Abu-Naser

Department of Information Technology,
Faculty of Engineering and Information Technology,
Al-Azhar University, Gaza, Palestine

Abstract. *In this paper we describe a Artificial Neural Network model of children's development on the balance scale task. In balance scale experiments, the child is asked to predict the outcome of placing certain numbers of equal weights at various distances to the left or right of a fulcrum. Both stage progressions and information salience effects have been found with children on this task. Artificial Neural Network provided better fits to these human data than did previous models, whether rule-based or connectionist. The ANN network model was used to generate a variety of novel predictions for psychological research. The dataset was collected from UCI Machine learning repository. The model was trained and validated using Just Neural Network Tool. The accuracy rate was 93%.*

Keywords: Artificial Neural Network**, JNN**, balance scale, ANN

INTRODUCTION

One of the most challenging areas facing Cognitive Science is that of cognitive development. Although the child enters the world in a nearly helpless state, within a decade he can manipulate objects, reason abstractly, and communicate with others, and he has acquired a host of other skills too numerous to mention. If we ever hope to understand the nature (and nurture) of the human information processing system, the developmental processes that lead to these diverse abilities must be understood. For a long period, research on cognitive development was almost entirely experimental, but in recent years efforts have been made to explain developmental trends in computational terms.

Psychological researchers typically present the child with a rigid balance beam in which differing numbers of equal weights are placed on pegs at various distances to the left or right of a fulcrum. The child's task is to predict which side of the scale will drop when supporting blocks are removed. Typically, all of the weights on one side of the fulcrum are placed on a single peg.

Authors in [1] have used the six different types of balance scale problems to assess the rules that children might be using on this task. So-called *balance* problems have equal numbers of weights placed at equal distances from the fulcrum so that the scale balances. For *weight* problems, the side with more weights goes down since the distances from the fulcrum are equal. In *distance* problems, the side with the weights placed a greater distance from the fulcrum goes down since the two sides have equal weights. The three types of *conflict* problems have more weight on one side but more distance on the other side. The side that actually goes down is the side with greater weight for *conflict weight* problems, and the side with greater distance for *conflict-distance* problems. The scale balances in *conflict-balance* problems.

Author in [2] has found that children's performance on the balance scale progresses through four distinct stages, each of which can be characterized by a symbolic rule: (1) use weight information alone to determine if the scale will balance, (2) emphasize weight information, but also use distance information in the event that the weights to the left and right of the fulcrum are equal, (3) consider both weight and distance information for simple problems, but get confused when weight conflicts with distance, (4) multiply distance by weight for each side and compare the products. The author has noted that each of these rules makes specific predictions about the kinds of problems that children using the rule will solve. These predictions are given by the predicted percentages correct. This orderly stage progression constitutes the first major psychological regularity in the balance scale literature.

The other major balance scale phenomenon is the torque difference effect [3]. The torque on each side of the fulcrum is defined as the product of weight and distance for that side. The torque difference for the problem is the absolute difference between the torques on the two sides of the fulcrum. The psychological result is that the larger the torque difference, the easier the problem is for children to solve. This could be regarded as an effect of information salience; the more perceptually salient the critical information, the easier the problem is to solve.

An artificial neural network (ANN) is the piece of a computing system designed to simulate the way the human brain analyzes and processes information. It is the foundation of artificial intelligence (AI) and solves problems that would prove impossible or difficult by human or statistical standards. ANNs have self-learning capabilities that enable them to produce better results as more data becomes available [4].

Artificial neural networks are built like the human brain, with neuron nodes interconnected like a web. The human brain has hundreds of billions of cells called neurons. Each neuron is made up of a cell body that is responsible for processing information by carrying information towards (inputs) and away (outputs) from the brain.
An ANN has hundreds or thousands of artificial neurons called processing units, which are interconnected by nodes. These processing units are made up of input and output units. The input units receive various forms and structures of information based on an internal weighting system, and the neural network attempts to learn about the information presented to produce one output report. Just like humans need rules and guidelines to come up with a result or output, ANNs also use a set of learning rules called backpropagation, an abbreviation for backward propagation of error, to perfect their output results [4].
An ANN initially goes through a training phase where it learns to recognize patterns in data, whether visually, aurally, or textually. During this supervised phase, the network compares its actual output produced with what it was meant to produce—the desired output. The difference between both outcomes is adjusted using backpropagation. This means that the network works backward, going from the output unit to the input units to adjust the weight of its connections between the units until the difference between the actual and desired outcome produces the lowest possible error [4].
During the training and supervisory stage, the ANN is taught what to look for and what its output should be, using yes/no question types with binary numbers. For example, a bank that wants to detect credit card fraud on time may have four input units fed with these questions: (1) Is the transaction in a different country from the user's resident country? (2) Is the website the card is being used at affiliated with companies or countries on the bank's watch list? (3) Is the transaction amount larger than $2,000? (4) Is the name on the transaction bill the same as the name of the cardholder?
The bank wants the "fraud detected" responses to be Yes Yes Yes No, which in binary format would be 1 1 1 0. If the network's actual output is 1 0 1 0, it adjusts its results until it delivers an output that coincides with 1 1 1 0. After training, the computer system can alert the bank of pending fraudulent transactions, saving the bank lots of money.
Artificial neural networks are paving the way for life-changing applications to be developed for use in all sectors of the economy. Artificial intelligence platforms that are built on ANNs are disrupting the traditional ways of doing things. From translating web pages into other languages to having a virtual assistant order groceries online to conversing with chatbots to solve problems, AI platforms are simplifying transactions and making services accessible to all at negligible costs[5].
Artificial neural networks have been applied in all areas of operations. Email service providers use ANNs to detect and delete spam from a user's inbox; asset managers use it to forecast the direction of a company's stock; credit rating firms use it to improve their credit scoring methods; e-commerce platforms use it to personalize recommendations to their audience; chatbots are developed with ANNs for natural language processing; deep learning algorithms use ANN to predict the likelihood of an event; and the list of ANN incorporation goes on across multiple sectors, industries, and countries[5].

LITERATURE REVIEW

Artificial Neural Networks have been used many fields. In Education such as: Predicting Student Performance in the Faculty of Engineering and Information Technology using ANN, Prediction of the Academic Warning of Students in the Faculty of Engineering and Information Technology in Al-Azhar University-Gaza using ANN, Arabic Text Summarization Using AraBERT Model Using Extractive Text Summarization Approach[5].

In the field of Health such as: Parkinson's Disease Prediction, Classification Prediction of SBRCTs Cancers Using ANN, Predicting Medical Expenses Using ANN, Predicting Antibiotic Susceptibility Using Artificial Neural Network, Predicting Liver Patients using Artificial Neural Network, Blood Donation Prediction using Artificial Neural Network, Predicting DNA Lung Cancer using Artificial Neural Network, Diagnosis of Hepatitis Virus Using Artificial Neural Network, COVID-19 Detection using Artificial Intelligence[5].

In the field of Agriculture: Plant Seedlings Classification Using Deep Learning , Prediction of Whether Mushroom is Edible or Poisonous Using Back-propagation Neural Network, Analyzing Types of Cherry Using Deep Learning, Banana Classification Using Deep Learning, Mango Classification Using Deep Learning, Type of Grapefruit Classification Using Deep Learning, Grape Type Classification Using Deep Learning, Classifying Nuts Types Using Convolutional Neural Network, Potato Classification Using Deep Learning, Age and Gender Prediction and Validation Through Single User Images Using CNN[6].

In other fields such as : Predicting Software Analysis Process Risks Using Linear Stepwise Discriminant Analysis: Statistical Methods, Predicting Overall Car Performance Using Artificial Neural Network, Glass Classification Using Artificial Neural Network, Tic-Tac-Toe Learning Using Artificial Neural Networks, Energy Efficiency Predicting using Artificial Neural Network, Predicting Titanic Survivors using Artificial Neural Network, Classification of Software Risks with Discriminant Analysis Techniques in Software planning Development Process, Handwritten Signature Verification using Deep Learning, Email

Classification Using Artificial Neural Network, Predicting Temperature and Humidity in the Surrounding Environment Using Artificial Neural Network, English Alphabet Prediction Using Artificial Neural Networks[6].

In balance scale field such as: The first computational simulation of balance scale phenomena for modeling of each the four stages in terms of production rules. This work described central features of the child's performance at each stage, but did not explain transitions between stages. Since their model appeared well before the torque difference effect was known, it did not deal with that effect. It is interesting to note that the torque difference effect is not explainable by these sorts of rules since any such rule would apply regardless of the torque difference involved in a particular problem. For example, the weight or distance on one side is greater than that on the other side regardless of how much greater it is [5].
The first balance scale model to address the transition issue in a serious way was by [6]. He used a production system that modified its existing, overly-general rules through discrimination learning. The learning mechanism searched for differences between cases where correct predictions were made and cases where errors were made. Unfortunately, there was no detailed assessment of stages in this model. It was evaluated only by noting that there was an increased percentage of problems correct as training progressed. On the negative side, it was reported that the model learned rules that children never showed, failed to focus on weight in formation before distance information, and never reached stage 4. The fact that it did not focus on weight information before distance suggests that the model did not capture stages 1 and 2. It was explained that the model could not reach stage 4 since torque could not be described in the representation language that was employed and the program could not construct new representations. The model did not try for the torque difference effect and presumably could not capture it because of its exclusive reliance on symbolic rules.
A rule learning program commonly used in contemporary cognitive modeling is Soar [7], which constructs its own rules by caching the results of look-ahead search. It too has been applied to balance scale phenomena with some success [8]. Soar reportedly acquired stages 1, 2, and 3 but, like Langley's model, did not manage to reach the performance characteristic of rule 4. Moreover, it is unclear how dependent the Soar model was on getting balance scale problems in a certain order. It may well be that different problem orders would yield different orders of acquisition of rules. Like the other rule-based models, the author did not try for the torque difference effect, nor is it apparent how it could in principle capture this effect. The author in [9] reported a simulation of balance scale stages using a connectionist network with the back-propagation learning rule. His model required a number of limiting assumptions, including a strong bias in the training patterns favoring equal distance problems (i.e., *balance* and *weight* problems) and a forced segregation of weight vs. distance information in connections to the hidden units. The network did progress through the first three stages of the balance scale, but there was a great deal of shifting back and forth between rules 3 and 4, with stage 4 never being clearly established[10].

METHODOLOGY

Dataset of the balance Scale

We collected the dataset form UCI Machine learning repository [11]. The dataset consists of 625 samples with 5 attributes as can be seen in Table 1.

Table 1: Attributes of the dataset

#	Attribute	Possible values	Type
1	Left-Weight	1,2,3,4,5	Input
2	Left-Distance	1,2,3,4,5	Input
3	Right-Weight	1,2,3,4,5	Input
4	Right-Distance	1,2,3,4,5	Input
5	Class Name	L, B, R	Output

Building the ANN Model

We have used Just Neural Network tool [64] to build a multilayer ANN model. The proposed model consists of 3 Layers: Input Layer with 4 nodes, Hidden Layer with 8 nodes, and Output Layer with one node as can be seen in Figure 1.

We have sat the parameters of the proposed model as follows: Learning Rate 0.185 and the Momentum to be 0.185, and Average Error rate to be 0.01 (as shown in Figure 2).

International Journal of Academic Information Systems Research (IJAISR)
ISSN: 2643-9026
Vol. 4, Issue 9, September – 2020, Pages: 74-81

Evaluating the ANN model

The Balance Scale Dataset consists of 625 samples with 5 attributes as Table 1. We imported the CSV file of the Balance Scale data into the JNN environment (as seen in Figure 3). We divided the imported dataset into two groups (Training and Validation) randomly using the JNN tool. The Training consists of approximately 67% (425 samples) and the validation set consists of 33% of the dataset (200 samples). After making sure that the parameter control was sat properly, we started training the ANN model and keeping eye on the learning curve, loss error and validation accuracy. We kept training the ANN model for 182769 cycles. The best accuracy we got was 93% (as seen in Figure 4). We determined the most influential factors in the Balance Scale Dataset as in Figure 5. Figure 6 shows the summary of the proposed model.

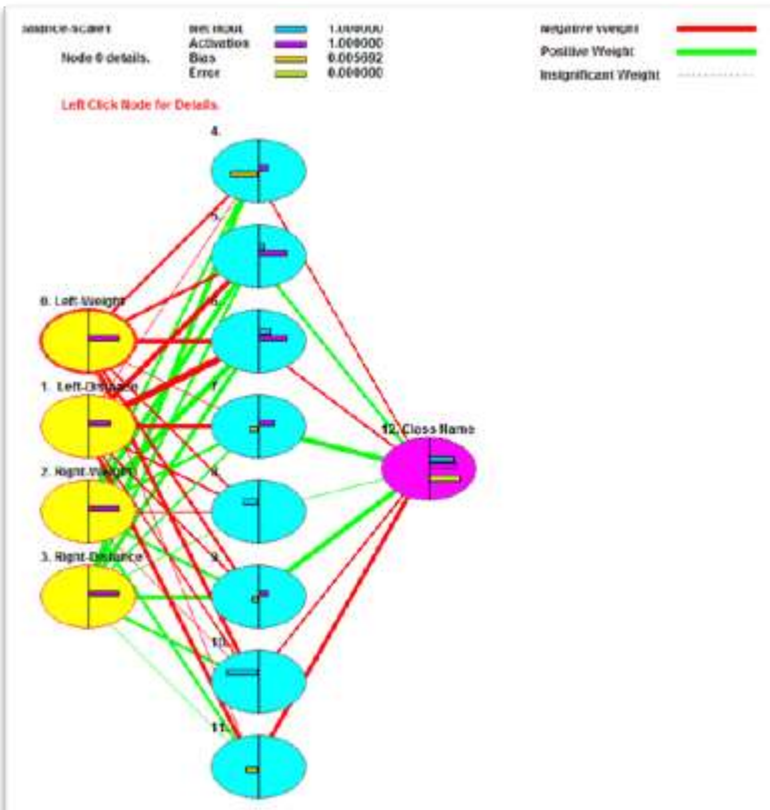


Figure 1: Architecture of Proposed ANN Model

International Journal of Academic Information Systems Research (IJAISR)
ISSN: 2643-9026
Vol. 4, Issue 9, September – 2020, Pages: 74-81

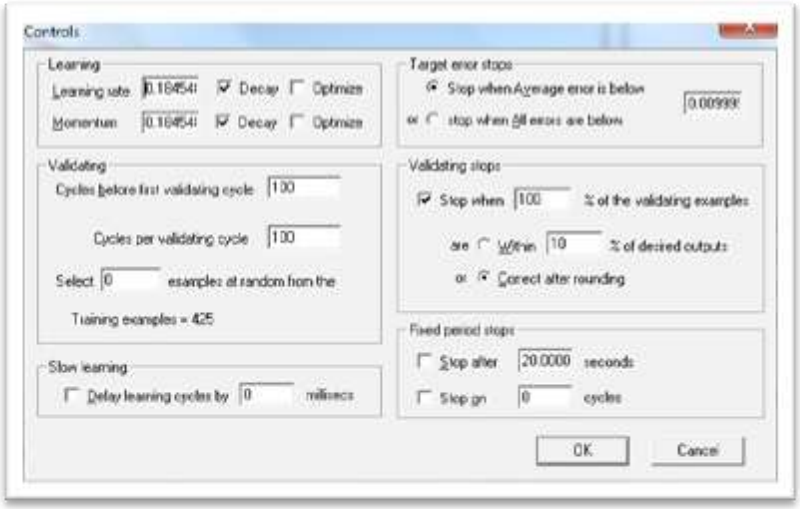
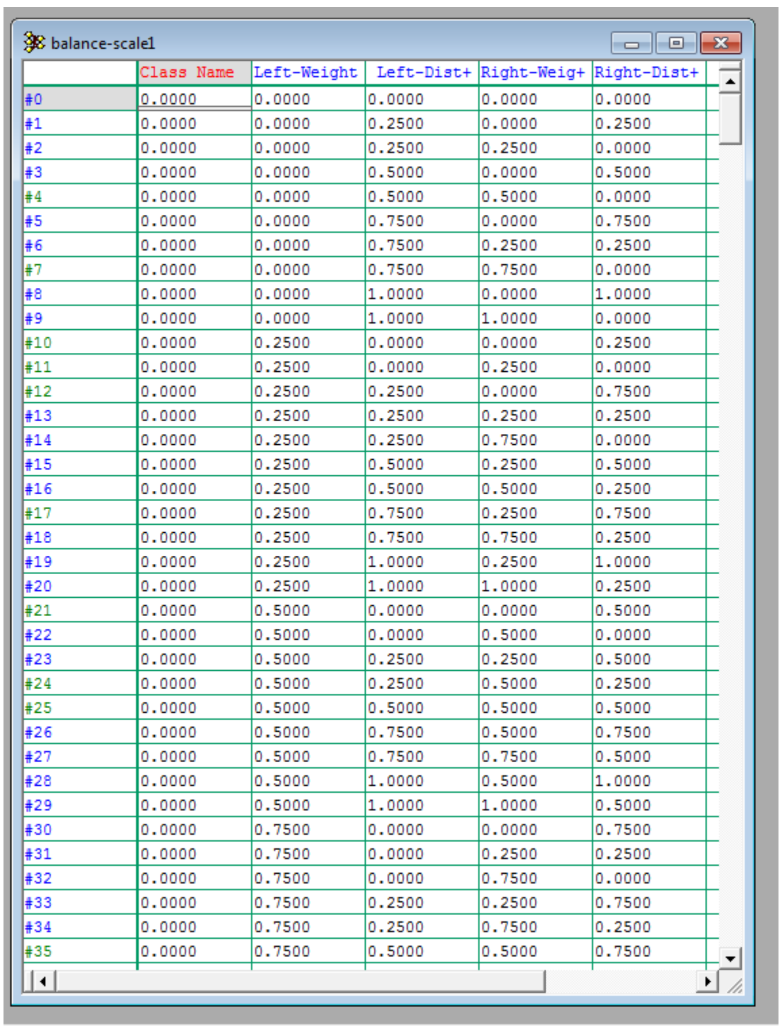


Figure 2: Control parameters of Proposed ANN model


balance-scale1

	Class Name	Left-Weight	Left-Dist+	Right-Weig+	Right-Dist+
#0	0.0000	0.0000	0.0000	0.0000	0.0000
#1	0.0000	0.0000	0.2500	0.0000	0.2500
#2	0.0000	0.0000	0.2500	0.2500	0.0000
#3	0.0000	0.0000	0.5000	0.0000	0.5000
#4	0.0000	0.0000	0.5000	0.5000	0.0000
#5	0.0000	0.0000	0.7500	0.0000	0.7500
#6	0.0000	0.0000	0.7500	0.2500	0.2500
#7	0.0000	0.0000	0.7500	0.7500	0.0000
#8	0.0000	0.0000	1.0000	0.0000	1.0000
#9	0.0000	0.0000	1.0000	1.0000	0.0000
#10	0.0000	0.2500	0.0000	0.0000	0.2500
#11	0.0000	0.2500	0.0000	0.2500	0.0000
#12	0.0000	0.2500	0.2500	0.0000	0.7500
#13	0.0000	0.2500	0.2500	0.2500	0.2500
#14	0.0000	0.2500	0.2500	0.7500	0.0000
#15	0.0000	0.2500	0.5000	0.2500	0.5000
#16	0.0000	0.2500	0.5000	0.5000	0.2500
#17	0.0000	0.2500	0.7500	0.2500	0.7500
#18	0.0000	0.2500	0.7500	0.7500	0.2500
#19	0.0000	0.2500	1.0000	0.2500	1.0000
#20	0.0000	0.2500	1.0000	1.0000	0.2500
#21	0.0000	0.5000	0.0000	0.0000	0.5000
#22	0.0000	0.5000	0.0000	0.5000	0.0000
#23	0.0000	0.5000	0.2500	0.2500	0.5000
#24	0.0000	0.5000	0.2500	0.5000	0.2500
#25	0.0000	0.5000	0.5000	0.5000	0.5000
#26	0.0000	0.5000	0.7500	0.5000	0.7500
#27	0.0000	0.5000	0.7500	0.7500	0.5000
#28	0.0000	0.5000	1.0000	0.5000	1.0000
#29	0.0000	0.5000	1.0000	1.0000	0.5000
#30	0.0000	0.7500	0.0000	0.0000	0.7500
#31	0.0000	0.7500	0.0000	0.2500	0.2500
#32	0.0000	0.7500	0.0000	0.7500	0.0000
#33	0.0000	0.7500	0.2500	0.2500	0.7500
#34	0.0000	0.7500	0.2500	0.7500	0.2500
#35	0.0000	0.7500	0.5000	0.5000	0.7500

Figure 3: Imported Balance Scale Dataset in JNN Environment

International Journal of Academic Information Systems Research (IJAISR)
ISSN: 2643-9026
Vol. 4, Issue 9, September – 2020, Pages: 74-81

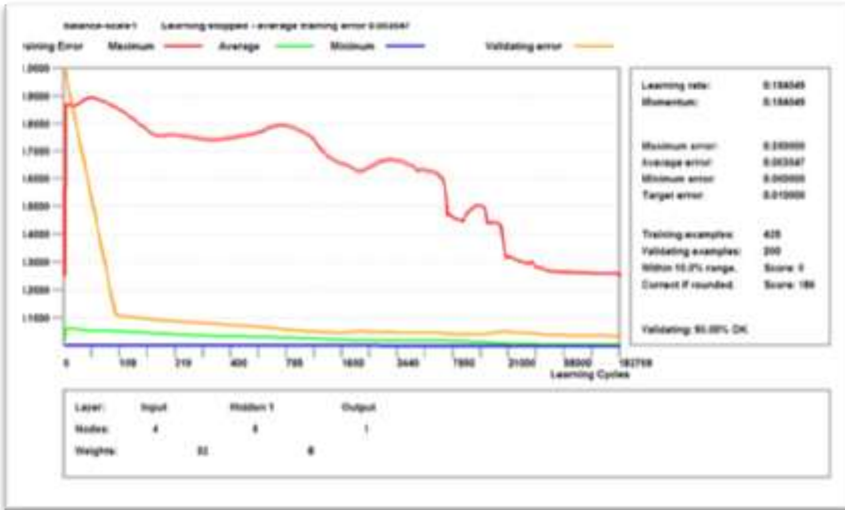


Figure 4: Training and validating the proposed ANN model in JNN environment

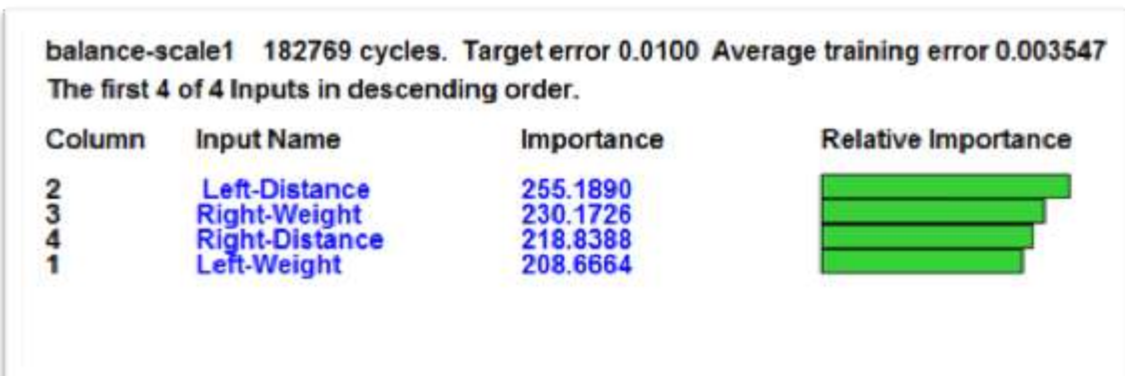


Figure 5: Most influential factors in the Balance Scale Dataset

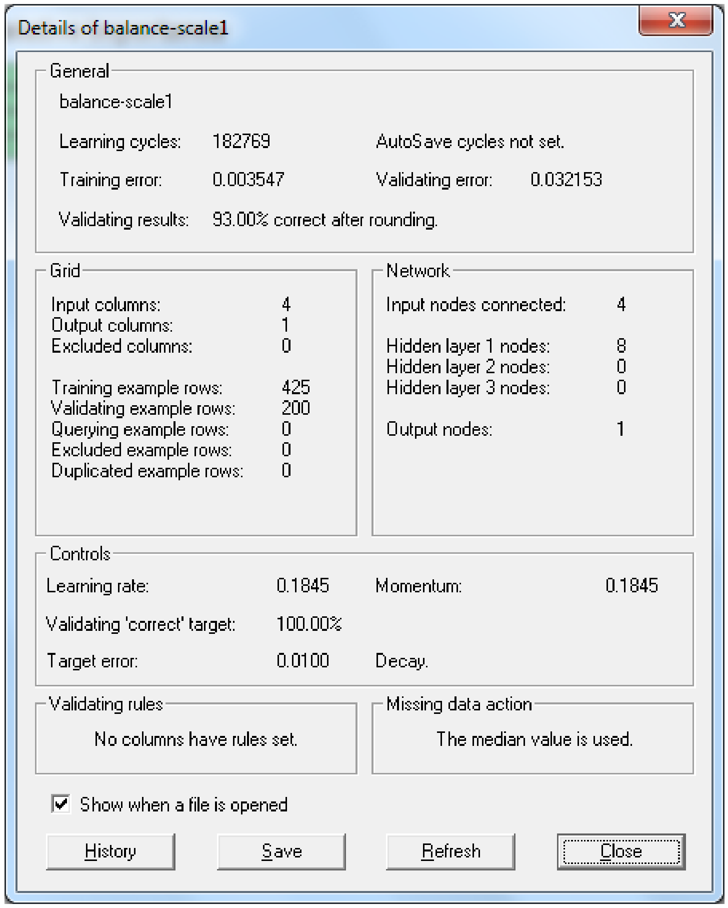


Figure 6: The summary of the proposed model

CONCLUSION

The ANN proposed model reported here suggest that Artificial Neural Networks can capture the main features of cognitive development on the balance scale. Combined with other simulations, this suggests that JNN is a particularly promising tool for modeling cognitive development. Such successful models could well provoke new theories of cognitive development, including explanations of performance on the balance scale. Much further computational and psychological work will be required to formulate any such theory in full, but we close with some speculations about the broad outline of such a theory. The proposed ANN model was tested and gave 93% as an initial experiment.

References

1. Bates, E.A. and Elman, J.L. (1993). Connectionism and the study of change. In M.H. Johnson (Ed.), Brain development and cognition, pp. 623-642. Oxford: Blackwell.
2. Belew, R. (1993). Interposing an ontogenetic model between genetic algorithms and neural networks. In C.L. Giles, S.J. Hanson, and J.D. Cowan (Eds.), Proceedings of the Neural Information Processing Society 5. San Mateo, CA: Morgan Kaufman.
3. Brainerd, C.J. (1973). Judgments and explanations as criteria for the presence of cognitive structures. Psychological Bulletin 79, 172-179.
4. Nolfi, S., Elman, J.L., and Parisi, D. (1990). Learning and evolution in neural networks. Technical Report 9019,
5. McClelland, J.L. (1989). Parallel distributed processing: Implications for cognition and development. In Morris, R.G.M. (Ed.), Parallel distributed processing: Implications for psychology and neurobiology, pp. 8-45. Oxford University Press.
6. Newell, A. (1990). Unified theories of cognition. Cambridge, MA: Harvard University Press.
7. Nisbett, R.E. and Wilson, T.D. (1977). Telling more than we can know: Verbal reports on mental processes. Psychological Review, 84, 231-259.

8. Nolfi, S., Elman, J.L., and Parisi, D. (1990). Learning and evolution in neural networks. Technical Report 9019, Center for Research in Language, University of California at San Diego.
9. Oden, G.C. (1987). Concept, knowledge, and thought. Annual Review of Psychology, 38, 203-227.
10. Piaget, J. (1963). The origins of intelligence in children (M. Cook, Trans.), New York: Norton.
11. UCI Machine Learning repository (https://archive.ics.uci.edu/ml/datasets.html)
12. JustNN Tool.