GLOBAL CONFERENCE ON BUSINESS AND FINANCE PROCEEDINGS

VOLUME 6, NUMBER 1
2011
ISSN 1931-0285 CD
ISSN 1941-9589 ONLINE

The Institute for Business and Finance Research

Las Vegas, Nevada
January 2-5, 2011

Global IBFR

TheIBFR.ORG

ECHO ARTMAP: CONTEXT SENSITIVITY WITH NEURAL NETWORKS IN FINANCIAL DECISION-MAKING

Charles Wong, Boston University
Massimiliano Versace, Boston University

ABSTRACT

Context modifies the influence of any trading indicator. Ceteris paribus, a buyer would be more cautious buying in a selling market context than in a buying market. In order for automated, adaptive systems like neural networks to better emulate and assist human decision-making, they need to be context sensitive. Most prior research applying neural networks to trading decision support systems neglected to extract contextual cues, rendering the systems blind to market conditions. This paper explores the theoretical development and quantitative evaluation of context sensitivity in a novel fast learning neural network architecture, Echo ARTMAP. The simulated risk and cost adjusted trading results compare very favorably on a 10-year, random stock study against the market random walk, regression, auto-regression, and multiple neural network models typically used in prior studies. By combining human trader techniques with biologically inspired neural network models, Echo ARTMAP may represent a new tool with which to assist in financial decision-making and to explore life-like context sensitivity.

INTRODUCTION

Stock prices refer to the latest mutually decided transaction price and time between a voluntary buyer and seller. If the stock prices over time are increasing, they indicate that the buying interest exceeds the selling interest. This signals a bullish or optimistic market context favorable to investment, all else equal. A successful trader (Schwager, 1994) often considers the underlying market sentiment when making decisions. This sensitivity to context in decision-making is one of the hallmarks of human intelligence (Akman, 2002).

Human subjects often treat similar tasks differently under different contexts (e.g. Carraher, Carraher, & Schliemann, 1985; Bjorklund & Rosenblum, 2002). Working memory allows features to be tracked over time to extract a context (Kane & Engle, 2002; Baddeley & Logie, 1999). Context sensitivity theoretically enables the decision-maker to disambiguate different feature inputs that may be identical at single points in time (Kane & Engle, 2002).

To better model human decision-making with context sensitivity, an automatic decision system must be context sensitive (see Figure 1, left). Tracking the price over time to determine whether the market is uptrending (bullish) or downtrending (bearish) intuitively provides contextual cues (Schwager, 1994).

Neural networks are biologically inspired, automated, and adaptive analysis models that can better accommodate non-linear, non-random, and non-stationary financial time series than alternatives (e.g. Lo, 2001; Lo & Repin, 2002; Lo, 2007; Gaganis, Pasiouras, & Doumpos, 2007; Yu, Wang, & Lai, 2008). Much research in the past decade applies them to financial time series with typically strong and compelling empirical results. Table 1 summarizes our survey of 25 studies published in the past decade (Wong & Versace, 2010a).

Figure 1: One-year daily prices for the Dow Jones Industrial Average

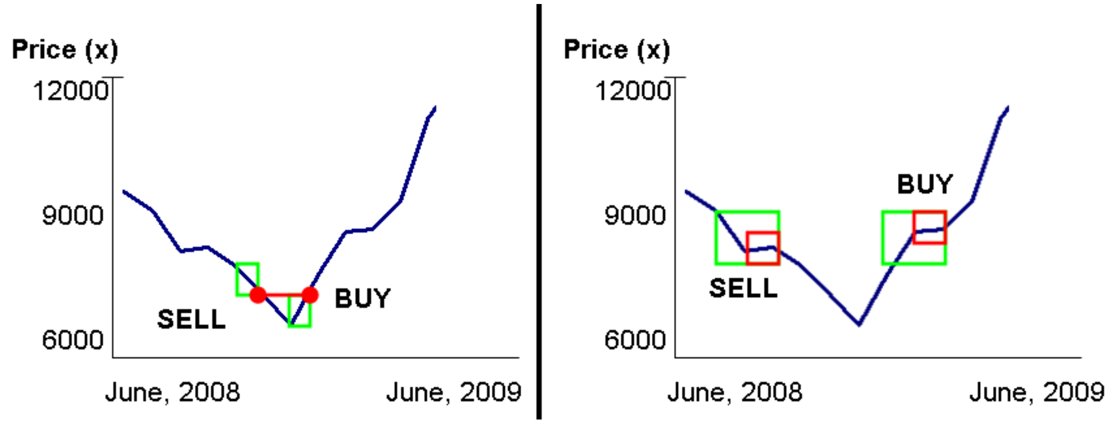

Figure 1: *One-year daily prices for the Dow Jones Industrial Average. (left) Identical point inputs (the horizontal line indicates same price) lead to different classes. The boxes show different contexts that can disambiguate the two inputs. (right) Identical short duration inputs (small boxes) lead to different classes. The longer duration input (large green boxes) can disambiguate the two inputs.*

Table 1: Survey

	Slow Learning	Fast Learning
Context Blind	(14) - Outperforms	(9) - Outperforms
Context Sensitive	(6) - Outperforms	(0) - N/A

Table 1*: A survey of 25 studies that applied neural networks to financial stock time series analysis over the past decade, divided into network learning rules (slow or fast) and context sensitivity. Instances and their typical results vs. the market are included.*

Two trends appear to be dominant from Table 1. First, neural networks appear to outperform the market, which is typically defined as a random walk or buying-and-holding of an index. Second, the results appear robust regardless of network learning rule or context sensitivity. The bias towards slow learning networks in Table 1 probably reflects their earlier availability (Rumelhart, Hinton, & Williams, 1986). Of the six studies able to track context via temporal sensitivity as in Figure 1, all relied on slow learning rules incorporated in Jordan and Elman networks (e.g. Versace et al, 2004; Yu, Wang, & Lai, 2008; Freitas, Souza, & Almeida, 2009; Jordan, 1986; Elman, 1990). Studies directly comparing fast learning, slow learning, and slow learning context sensitive networks have found no significant differences in empirical results (e.g. Saad et al, 1998).

This paper explores the disagreement between the intuition supporting the importance of context sensitivity and the empirical results (Table 1) showing no differential benefit relative to existing neural network models. Table 1 indicates existing models tend not to incorporate fast learning with context in finance. Therefore, this paper introduces a novel context sensitive fast learning network for transparent analysis (Moody & Darken, 1989; Carpenter, Grossberg, & Reynolds, 1991; Parsons & Carpenter, 2003). The base fast learning component model is ARTMAP (Amis & Carpenter, 2007) from the Adaptive Resonance Theory class of models. While ARTMAP is not a perfect blend of all existing fast learning characteristics (e.g. it differs in learning vs. Radial Basis Function networks), it can be regarded as a general purpose, default network that automatically adapts and scales its topology to a generic dataset (e.g. Carpenter, 2003). For this paper, benchmarks include random walk, regression, auto-regression, a slow learning backpropagation (Rumelhart, Hinton, & Williams, 1986), a fast learning ARTMAP (Amis & Carpenter, 2007), and a slow learning context sensitive model (Jordan, 1986).

The remaining sections of this paper divide as follows: Section II provides a brief review of the ARTMAP kernel mechanism as a base for later extension; Section III demonstrates how to theoretically adapt the ARTMAP to context sensitivity; Section IV outlines the quantitative experimental setup and evaluation; Section V provides the results and discussion; and Section VI contains concluding remarks.

KERNEL REVIEW FOR A TYPICAL FAST LEARNING MODEL: ARTMAP

Slow learning networks possess hidden layers that have opaque representations relating inputs to outputs. In contrast, fast learning allows immediate and transparent convergence for independent storage layer nodes. ARTMAP is a type of fast learning network that was inspired by biological constraints and can be adapted to a variety of uses. Extensive literature shows its capabilities and interprets its mathematical bases (e.g. Amis & Carpenter, 2007; Parsons & Carpenter, 2003). Figure 2 (left) shows the default ARTMAP flow diagram.

Figure 2: (Left) a Default ARTMAP Network Diagram Showing The Three-Layer Architecture and Circles Represent Individual Components of the Pattern (Right)

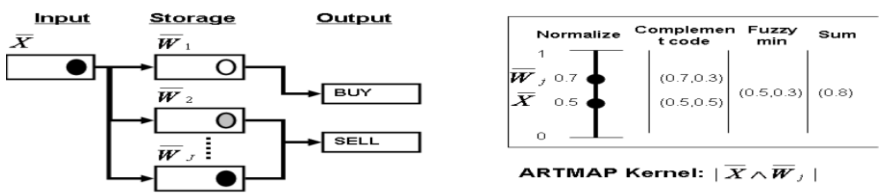

Figure 2: (left) A default ARTMAP network diagram showing the three-layer architecture. For a particular input pattern, $\overline{X}$ *, the ARTMAP kernel finds the most similar stored pattern,* $\overline{W}_j$ *, which maps to a specific output node. Boxes, or nodes, represent patterns (responses in the output layer) and circles represent individual components of the pattern. (right) An example of the ARTMAP kernel calculating the similarity between an input pattern and a single stored pattern. See the text for details.*

There are three layers in the default ARTMAP network. The input layer receives input patterns, each represented by a vector with one or more components, $\overline{X}$. Given this vector, the network finds the most similar vector $\overline{W}_j$ from the storage layer, where *j = {1..J}* and *J* is the number of storage nodes. The output layer node associated with the most similar storage node dictates the network response. The ARTMAP kernel, which is a function that determines the similarity between vectors (Bishop, 2006), models pattern recognition as per equation (1):

$$T_j = |\overline{X} \wedge \overline{W}_j|, \qquad (1)$$

where T_j is the similarity score for storage node *j*. The kernel procedure has four steps: normalize all vector component values to between 0 and 1; complement code both vectors such that $\overline{X} = (x_1, 1 - x_1)$; apply the fuzzy min operator (^) on the vectors; and sum (||). For example, given a normalized input value of 0.5 and a particular normalized storage node of 0.7, the complement codes would be (0.5, 0.5) and (0.7, 0.3). The fuzzy min would be the lesser of each component, or (0.5, 0.3) and their sum would be 0.8, which as a normalized value can also be read as 80% similar. The default ARTMAP learning rules that update and add storage layer nodes with their associated output nodes are not modified and are not treated here. For references on previously published ARTMAP models, please see http://techlab.bu.edu.

The following section provides the theoretical modifications to this kernel.

EXTENDING ARTMAP TO EXTRACT CONTEXT: ECHO ARTMAP

The approach taken here explores fast learning network rules with context sensitivity. Figure 3 shows how a fast learning ARTMAP model can be modified to process time patterns in the data with three steps via input delays, decays, and output-to-input recurrence to create the novel Echo ARTMAP model.

Figure 3: Learning ARTMAP Model

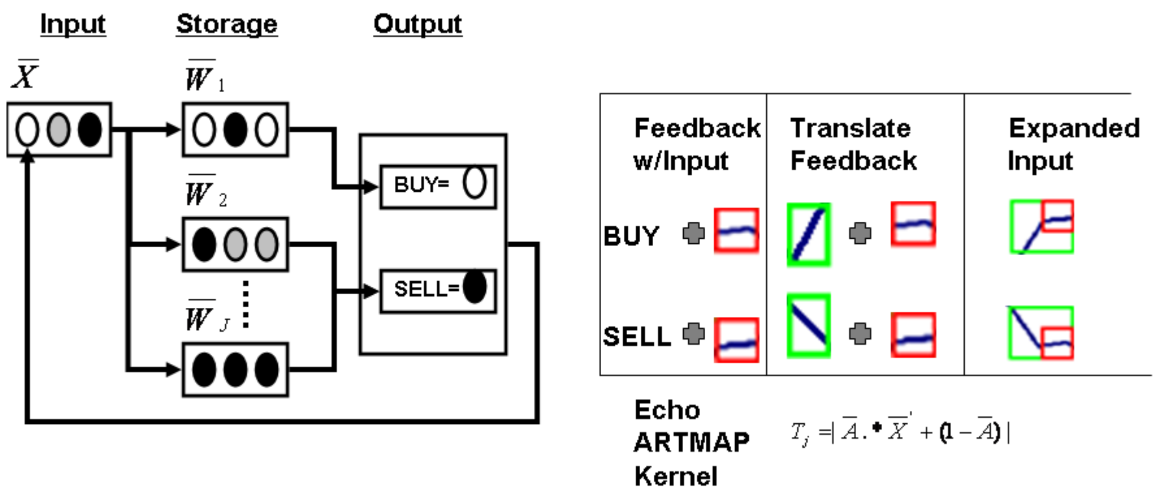

Figure 3: (left) The full Echo ARTMAP architecture with time delay, decay, and recurrence. See text for the breakdown of the three steps. (right) Excerpted from figure 1, the feedback provides additional input information from past storage values. Translating the feedback back into its component pattern allows more information to be input into the network. This example assumes the patterns in Figure 1, left, have already been stored, for instance allowing the feedback Buy value to be translated into an uptrend. This process can be repeated infinitely, allowing greatly expanded inputs.

Figure 4: The Influence of Past Inputs at Time t.

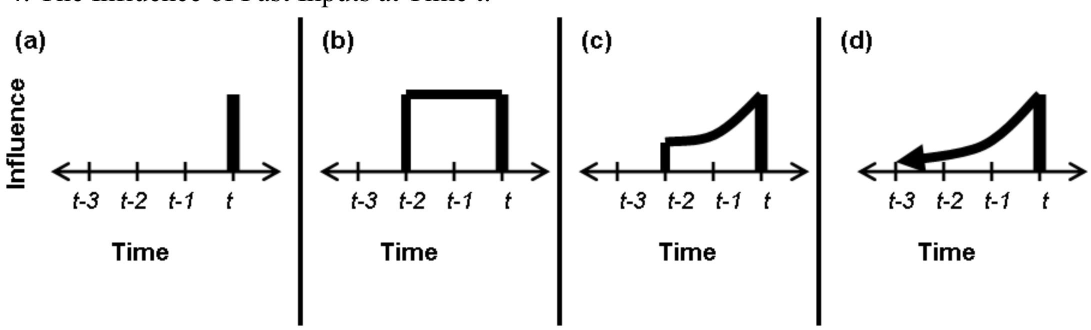

Figure 4: The influence of past inputs at time t. (a) At time t, a model with no time delay only considers the current input from t. (b) A model with a delay of 2 considers both the input from t and the past two inputs equally. (c) A model with delay of 2 with decay considers the input from t and the past two inputs, but with more emphasis on more current inputs. (d) A model with delay of 2 with decay and output-to-input recurrence theoretically considers all prior inputs albeit with very little emphasis on distant inputs in time.

Implementing input time delays allows an ARTMAP network to model one aspect of working memory. Figure 3 (left) shows the ARTMAP network from figure 2 with multiple components in each node, the right two being the same feature at different points in time. Similarity proceeds from equation (1), but depends on multiple points in time. Figure 4 (a and b) compares the influence of a given input over time when introducing input time delay. With no delay, the network at time *t* can only consider inputs from time *t*.

Implementing time decay allows an ARTMAP network to model a more complex, non-stationary working memory. In a non-stationary data set, proximal points in time should have more influence than distal points in time (Hamilton, 1994). The underlying state or context is shifting over time, such that feature values within the current state are more relevant. Equation (2) shows how to scale the contextual importance:

$$T_j = |\overline{A} .* \overline{X}' + (1 - \overline{A})|, \qquad (2)$$

where $\overline{X}' = \overline{X} \wedge \overline{W}_j$, or the component-wise collection after the first three steps in equation (1), $\overline{A} = (a_1, a_2, ...a_M)$, $0 <= a_m < \infty$, a is monotonically decreasing with m, $m = \{1..M\}$, and M is the number of time delays. The (.*) operator indicates an array multiplication that performs element-by-element factoring. The vector $\overline{A}$ scales the importance of the components of $\overline{X}'$. Figure 4 (c) shows the scaling effect of monotonically decreasing *A* components as the influence of inputs from time *t* diminishes after time *t*. The $(1 - \overline{A})$ term ensures the T_j score has a maximum of 1. Although negative scores are possible, the default ARTMAP matching rule selects the highest positive T_j score (Amis & Carpenter, 2007).

Finally, implementing output-to-input recurrence is useful since it allows a compressed version of past inputs (i.e. as implied by the output decision based on these inputs) to remain available to influence decisions at future times. For example, a fixed small window in time may not fully capture a large pattern (Figure 1, right). A fixed large window captures non-pattern noise that leads to over-fitting. Output-to-input recurrence can simulate a dynamic window size. Figure 3 (right) shows a short pattern preceded by a recurrent output label. Since each output label associates with a specific storage pattern, using the label as a further input creates sensitivity to patterns that are composites of past short window patterns as needed, generating the profile in Figure 4(d). This combines the flexibility of long windows with the lower susceptibility to over-fitting of short windows. The novel Echo ARTMAP model performs this implicit composition of past inputs.

QUANTITATIVE TESTING OF ECHO ARTMAP

Figure 5 shows the overall study framework for collecting and evaluating benchmark performance via risk-adjusted rates of return for the Echo ARTMAP network and benchmarks.

This study randomly selects five stocks from the Dow Jones Industrial Average index as of 1999 (American Express, Exxon-Mobil, IBM, JP Morgan Chase, and United Technologies Corporation) for daily online training spanning ten years from 2000-2009. This generates 12,500 sample data points, which exceeds the sample size required for 95% confidence given estimated population standard deviation under both normal and non-normal assumptions (Higgins, 2004). Online means the fast learning networks continually expand their training set after testing on each trading day; the slow learning networks replicate this process by using rolling training window sizes of two years and averaging the results. Supervised classes derive from whether the forward one-day price change is positive, negative, or neutral. The single input feature uses a moving average period of 10 days subtracted from the current price. From this single feature, each benchmark model receives up to an 11-dimensional derived input set for each stock: 10 input delays from the single feature plus one from the benchmark's output-to-input recurrence where applicable.

Figure 5: The Outline Process Flow for Evaluating the Echo ARTMAP and Benchmarks

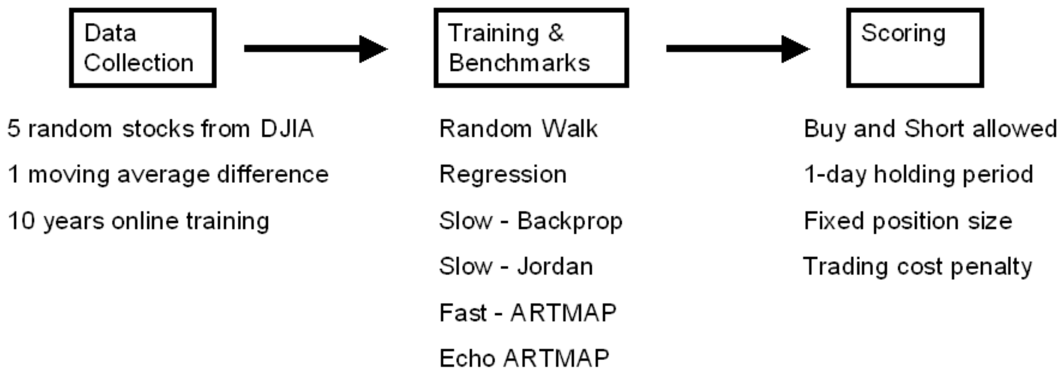

Figure 5: The outline process flow for evaluating the Echo ARTMAP and benchmarks. Data collection proceeds from http://finance.yahoo.com. This study uses ten years of online training. One moving average difference with the price provides the basic input. Trading costs and risk adjustment are included in fixed trading sizes (no trades are assumed to yield zero percent; gains and losses are replaced to maintain fixed trading size).

The benchmarks include: an industry standard random walk; regression and auto-regression (Box & Jenkins, 1970); a slow learning static input neural network, backpropagation (Rumelhart, Hinton, & Williams, 1986); a fast learning static input neural network, ARTMAP (Amis & Carpenter, 2007); a slow learning context sensitive neural network, the Jordan network (Jordan, 1986); and the novel Echo ARTMAP fast learning context sensitive network.

For scoring purposes, buying, not trading, and selling short are allowed (i.e. 3-class predictor). Each decision lasts for one day. Position sizes are fixed, with gains being removed and losses being replaced. Not trading is valued at zero gains and zero costs. Round trip trading costs deduct 0.1% per active trading decision. To counter this trading cost, the supervised learning classifies trading days with daily variance of less than 1% as not trading. In addition, the Sharpe Ratio (Chartered Financial Analyst Institute, 2010) divides the average return by the standard deviation of the returns. This provides an additional, singular, and objective measure of the risk/reward profiles for each benchmark.

RESULTS AND DISCUSSION

Figure 6 shows the six benchmarks' average annualized performance over five random Dow Jones Industrial Average stocks over ten years. For reporting purposes, the ten-year period divides into three periods of 3.33 years each to demonstrate a possible range of results. The Sharpe Ratios provide single, numerical measures of risk and reward for each benchmark.

Trading costs penalize each trade, which accounts for the random walk having a slightly negative annual rate. Without trading costs a random walk should consistently generate near zero average gains due to perfect hedging of buying and shorting. In agreement with Yu et al. (2008), the regressive benchmarks both had more difficulty than neural networks due to the non-linear nature of financial time series data and the penalties incurred from the trading costs. Results are combined in figure 6 for simplicity since both benchmarks had similar performance. In agreement with Saad et al. (1998) and Table 1, the Slow-Backprop (backpropagation), Slow-Jordan, and Fast-ARTMAP networks all had similar risk-adjusted performances that outperformed the random walk. The networks can generate high gains, but the transaction costs and the volatility reduce much of the benefits. The prior studies that contributed to Table 1 did not typically include transaction costs or risk adjustment in their analyses.

Figure 6: The 10-Year Financial Data Set Annualized Gains per Benchmark

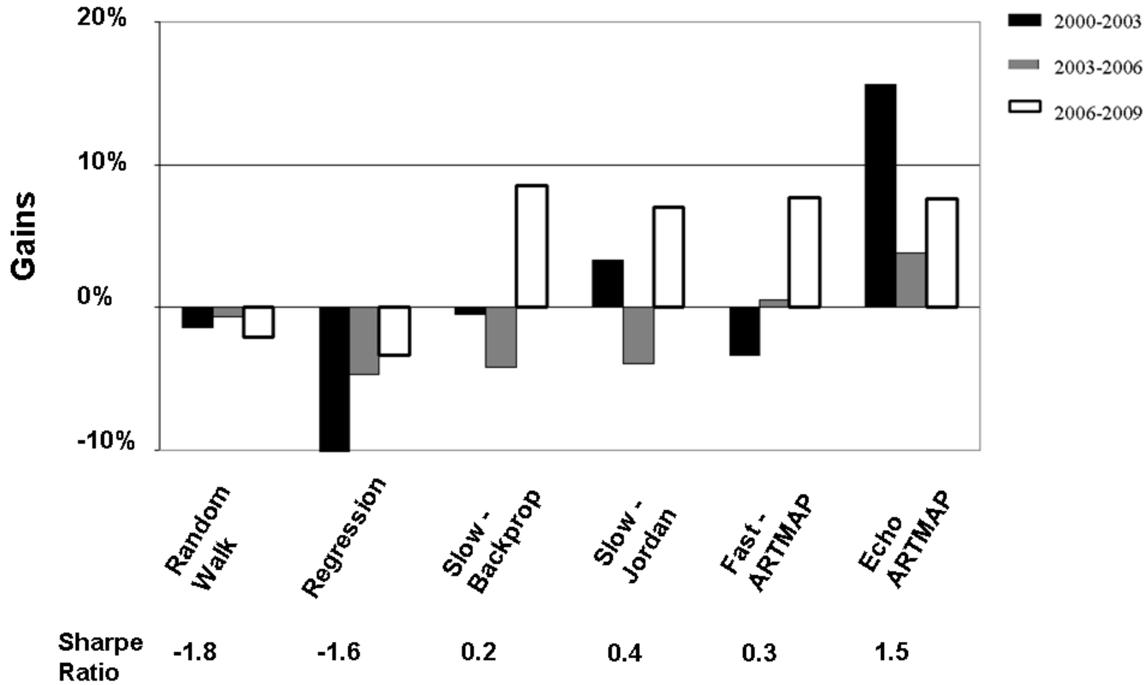

Figure 6: *The 10-year financial data set annualized gains per benchmark. The 10-year average for each benchmark is broken into three equal reporting periods for further granularity. Regression and auto-regression provided similar results and are combined for simplicity. All results include trading costs. The Sharpe Ratio is the average return divided by the standard deviation of returns. Typical mutual fund Sharpe Ratios range from -1.7 to 2.5 per www.morningstar.com.*

The novel Echo ARTMAP network strongly outperformed all other benchmarks on the sample of five stocks, with a mean 10-year annual gain net of costs of 9% and a Sharpe Ratio of 1.5. This quantitatively shows that neural network topology can have significant empirical effects and that adding context can greatly improve fast learning neural network performance on financial decision making.

CONCLUSION

In a financial time series, decision makers are best served by being cognizant of past and current indicators. This builds context into trading decisions. For automated systems like neural networks to emulate and assist in the decision-making process, they should be context sensitive. For neural networks to be adaptive and reactive to fluid changes in the environment, they should also rely on fast learning rules. The novel, fast learning context sensitive Echo ARTMAP can quickly and transparently incorporate the current market conditions into its decisions. Echo ARTMAP empirically outperforms the benchmarks ranging from autoregression to context sensitive slow learning Jordan to fast learning ARTMAP networks on randomly selected stock data over ten years of varying market conditions. While context-blind models cannot modulate their decisions based on extant environments and slow learning models react very slowly and poorly to ever-changing environments, the theory behind the enhancements in a fast learning, context sensitive model supports the Echo ARTMAP empirical findings. This supports the concept of working memory as a means of extracting the context that disambiguates feature inputs over time and leads to more intelligent decision-making.

REFERENCES

Akman, V. (2002). Context in artificial intelligence: A fleeting overview, *In: La Svolta Contestuale, C. Penco, ed.*, McGraw-Hill, Milano.

Amis, G. & Carpenter, G. (2007). Default ARTMAP. *Proceedings of the International Joint Conference on Neural Networks (IJCNN,07) Orlando, Florida,* 777-782.

Baddeley, A. D., & Logie, R. (1999). Working memory: The multiple component model. In A. Miyake & P. Shah (Eds.), *Models of working memory: Mechanisms of active maintenance and executive control* (pp. 28-61). New York: Cambridge University Press.

Bishop, C. (2006). *Pattern recognition and machine learning,* Springer Science.

Bjorklund, D. & Rosenblum, K. (2002). Context effects in children's selection and use of simple arithmetic strategies, *Journal of Cognition & Development, 3*, 225-242.

Box, G. & Jenkins, G. (1970). *Time series analysis: Forecasting and control*, San Francisco: Holden-Day

Carpenter, G. (2003). Default ARTMAP. *Proceedings of the International Joint Conference on Neural Networks (IJCNN,03) Portland, Oregon, 1396-1401.*

Carpenter, G., Grossberg, S., & Reynolds, J. (1991), ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network, *Neural Networks, 4,* 565-588.

Carraher, T., Carraher, D., & Schliemann, A.(1985). Mathematics in the streets and in schools, *British Journal of Developmental Psychology, 3*, 21-29.

Elman, J. (1990). Finding structure in time, *Cognitive Science, 14*, 179-211.

Freitas, F., Souza, A., & Almeida, A. (2009) Prediction-based portfolio optimization model using neural networks, *Neurocomputing, 72(10),* 2155-2170.

Gaganis, C., Pasiouras, F., & Doumpos, M. (2007). Probabilistic neural networks for the identification of qualified audit opinions, *Expert Systems with Applications, 32*, 114–124.

Hamilton, J. (1994). Time series analysis, *Princeton University Press.*

Higgins, J. (2004). Introduction to modern nonparametric statistics, *Brooks/Cole-Thomson Learning*

Jordan, M. (1986). Serial order: A parallel distributed processing approach*, Institute for Cognitive Science Report 8604*, University of California, San Diego.

Kaizoji, T., & Sornette, D. (2008). Market bubbles and crashes, *Encyclopedia of Quantitative Finance*

Kane, M. & Engle, R. (2002). The role of prefrontal cortex in working-memory capacity, executive attention, and general fluid intelligence: An individual-differences perspective, *Psychonomic Bulletin & Review, 9(4),* 637-671.

Leung, C. & Tsoi, A. (2005). Combined learning and pruning for recurrent radial basis function networks based on recursive least square algorithms, *Neural Computing & Applications, 15*, 62-78.

Lo, A. (2001). Bubble, rubble, finance in trouble?, *Journal of Psychology and Financial Markets, 3*, 76-86.

Lo, A. (2007). The efficient markets hypothesis, in *The Palgrave Dictionary of Economics, Palgrave Macmillan*

Lo, A. & Repin, D. (2002). The psychophysiology of real-time financial risk processing, *Journal of Cognitive Neuroscience 14(3)*, 323- 339.

Moody, J. & Darken, C. (1989) Fast learning in networks of locally tuned processing units, *Neural Computation, 1*, 281-294.

Morning Star (2010). *Retrieved from* http://www.morningstar.com

Parsons, O. & Carpenter, G. (2003). ARTMAP neural networks for information fusion and data mining: Map production and target recognition methodologies. *Neural Networks, 16*, 1075-1089.

Rumelhart, D., Hinton, G., & Williams, R. (1986). Learning internal representations by error propagation, *in Parallel distributed processing: explorations in the microstructure of cognition, vol. 1*, MIT Press.

Saad, E., Prokhorov, E., & Wunsch, D. (1998). Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks, *IEEE Transactions on Neural Networks, 9(6),* 1456-1470.

Schwager, J. (1995). *The new market wizards: Conversations with America's top traders*, Wiley.

Versace, M., Bhatt, R., Hinds, O., & Schiffer, M. (2004) Predicting the exchange traded fund DIA with a combination of Genetic Algorithms and Neural Networks*, Expert Systems with Applications, 27(3)*, 417-425.

Wong, C., & Versace, M. (2010a). Rethinking neural networks in financial decision-making studies: Seven cardinal confounds, *in preparation, extended from* Adding back dollars and sense to financial forecasting with neural networks*, 14th International Conference on Cognitive and Neural Systems, Boston, Poster Session.*

Yu, L., Wang, S., & Lai, K. (2008) Neural network-based mean–variance–skewness model for portfolio selection, *Computers & Operations Research, 35*, 34– 46.

BIOGRAPHY

Charles Wong is a PhD candidate at the Cognitive and Neural Systems program, Boston University. He has previously worked for Deutsche Bank AG and KPMG LLP in New York. He can be contacted at ccwong@bu.edu.

Max Versace (PhD, Cognitive and Neural Systems, Boston University, 2007) is a Senior Research Scientist at the Department of Cognitive and Neural Systems at Boston University, Director of Neuromorphics Lab, and co-Director of Technology Outreach at the NSF Science of Learning Center

CELEST: Center of Excellence for Learning in Education, Science, and Technology. He is a co-PI of the Boston University subcontract with Hewlett Packard in the DARPA Systems of Neuromorphic Adaptive Plastic Scalable Electronics (SyNAPSE) project.