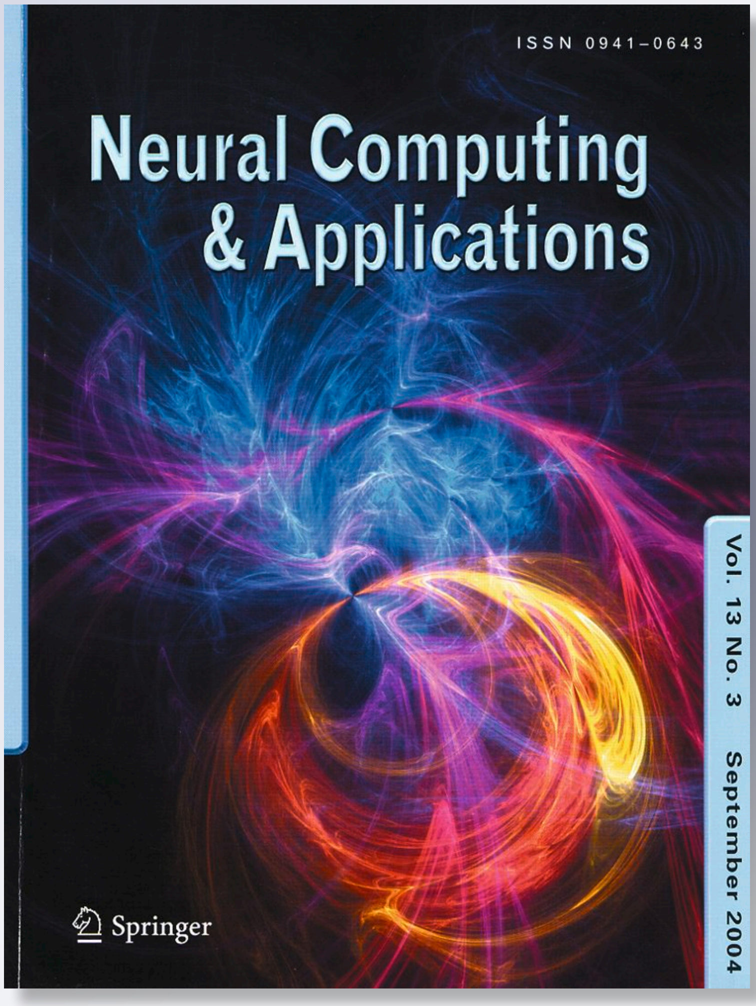
CARTMAP: a neural network method for automated feature selection in financial time series forecasting

Charles Wong & Massimiliano Versace

Neural Computing and Applications

ISSN 0941-0643

Neural Comput & Applic
DOI 10.1007/s00521-012-0830-8



Neural Comput & Applic
DOI 10.1007/s00521-012-0830-8

ORIGINAL ARTICLE

CARTMAP: a neural network method for automated feature selection in financial time series forecasting

Charles Wong · Massimiliano Versace

Received: 22 June 2010 / Accepted: 5 January 2012

Abstract In the past two decades, there has been much interest in applying neural networks to financial time series forecasting. Yet, there has been relatively little attention paid to selecting the input features for training these networks. This paper presents a novel CARTMAP neural network based on Adaptive Resonance Theory that incorporates automatic, intuitive, transparent, and parsimonious feature selection with fast learning. On average, over three separate 4-year simulations spanning 2004–2009 of Dow Jones Industrial Average stocks, CARTMAP outperformed related and classical alternatives. The alternatives were an industry standard random walk, a regression model, a general purpose ARTMAP, and ARTMAP with stepwise feature selection. This paper also discusses why the novel feature selection scheme outperforms the alternatives and how it can represent a step toward more transparency in financial modeling.

Keywords Feature selection · Neural networks · Financial time series · ARTMAP · Clustering · Adaptive Resonance Theory

1 Introduction to financial time series feature selection

Financial time series represent the aggregate sum of trading decisions under uncertainty [29, 36, 43, 50]. A quantitative stock recommendation system seeks to establish a consistent and objective method for generating trading decisions (e.g., [11, 34]). Financial indicators, as input features, form the core data presented to such a system that helps traders to determine a course of action. A key task in developing an effective trading strategy is identifying a set of features on which to focus.

Feature selection is critical to the success of large-scale automated classification schemes [48]. Different features reflect different aspects of the dataset, and carefully selecting features can improve classification [30]. Including too many features may introduce irrelevant or distracting attributes that impair performance [51]. Some features that are ineffective individually may be predictive when combined with others. Other features that are predictive individually may become contradictory when combined. Choosing a useful set of features may be a complex process that requires extensive experience and a deep understanding of the problem domain [51].

For the financial time series problem domain, expert human analysts typically attempt to examine and evaluate a set of features whose unique patterns precede predictable movements in the time series. These features are termed financial or economic indicators and are chosen carefully after much experience combining and validating them in a myriad of methods. Recent research in evaluating preselected features using neural networks on various financial time series features shows that they can outperform regression models [5, 12] and multiple discriminant models [6]. However, research specifically targeted at carefully selecting and filtering the features before adding them to time series models is relatively sparse [21, 46]. Doing so can greatly aid the financial time series analysis by presenting a cleaner and less contradictory set of features for evaluation.

C. Wong · M. Versace (✉)
Department of Cognitive and Neural Systems, Boston University, 677 Beacon Street, Boston, MA 02215, USA
e-mail: versace@cns.bu.edu
URL: http://cns.bu.edu/~versace

C. Wong
e-mail: cncsquared@gmail.com
URL: http://techlab.bu.edu/members/ccwong/

Published online: 15 January 2012

In general practice, a financial analyst will focus on a relatively small set of key features to confirm a predictive signal [24]. To address the theoretical issue of multicollinearity, correlated features, and related phenomena [4, 11], the analyst generally needs to ensure that the selected features are sufficiently different and uncorrelated. Intuitively, this makes sense because if the analyst uses similarly behaving features to confirm each other, it is a false confirmation. It is common sense that the two features will agree on particular signals and using both adds no new information while potentially introducing bias. See Fig. 1.

This article develops CARTMAP, a novel fast-learning neural network model that first clusters all correlated features together and then chooses at most one feature per cluster for training with an ARTMAP network [1]. This automatically builds in the common sense to ensure that the ultimately selected features are uncorrelated.

Data includes a time series of 1,000 consecutive trading-day prices for five randomly selected Dow Jones Industrial Average stocks (American Express, Bank of America, Coca-Cola, IBM, and Walmart) spanning 2004–2007. Prices from 2004 to 2006 provide the training and validation sets, while prices from 2007 provide the test set. The time series for each stock provides a set of simple moving averages calculated across 1-day to 10-day time intervals. Comparing each average with the price produces a set of moving-average crossovers, which provide direct, objective *bullish (buy)*, *bearish (sell)*, or *neutral (no trade)* recommendations. Ten moving-average crossover series from the ten time windows serve as the feature pool for the dataset. For simplicity, permitted actions of this study are limited to buying long, selling short, or not trading, and

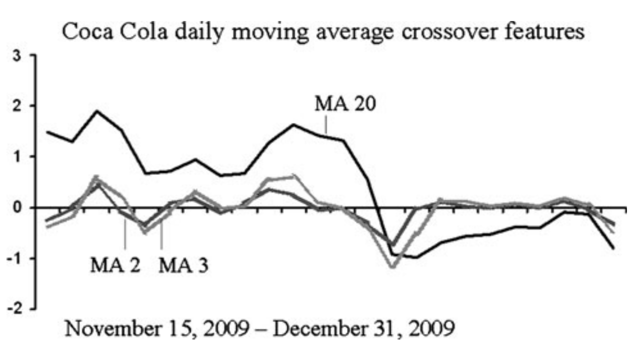


Fig. 1 The dangers of choosing correlated features. Three different example features are plotted on Coca-Cola from November to December 2009. The features are moving average crossovers with the price where information is contained at the zero crossing points. Thus, features MA2 and MA3 are very similar in where they cross the zero line. MA20 has only one zero crossing and is different than MA2 and MA3. If a model consists of the highly correlated features MA2 and MA3, the effect of one single feature is erroneously doubled. The analyst may get confirmation bias. On the other hand, if a model consists of the less correlated features MA2 and MA20, any agreement is more likely to be due to real signal detection effects

holding periods are set to a 1-day trading horizon. This study repeats the simulation on two additional 1-year walk-forward datasets spanning 2005–2008 and 2006–2009. In addition, this study repeats the three simulations using the entire Dow Jones Industrial Average composite index for CARTMAP only.

For the same data, this paper uses four different alternative methods for comparison with CARTMAP. The first method is a random walk of the testing prices. Random walks use Monte Carlo simulations of randomly selecting from among the same *buy, sell, or no trade* recommendations. The second method is a multiple linear least squares regression, a basic version of a common econometric approach to flexibly associate relationships between the inputs and outputs [11]. The third method uses a default ARTMAP neural network [1] that trains and tests on all features with no separate feature selection. The fourth is the ARTMAP neural network [1] enhanced with a stepwise feature selection method [40].

Section 2 provides a brief literature review on applying neural networks and feature selection in financial decision-making environments. Section 3 describes in detail the data collection and preprocessing that generates the features. Section 4 details the CARTMAP and the four alternative techniques. Section 5 presents comparative results and discussion, while Sect. 6 presents conclusions.

2 Brief literature review

Over the past decade, an increasing number of studies have used various types of artificial neural models for analyzing financial time series (e.g., [13, 27, 46, 47, 56]). Neural models are among the best adaptive analysis tools due to their biologically inspired learning rules since they better accommodate non-linear, non-random, and non-stationary financial time series than alternative techniques such as regression and random walks [31–33]. However, all neural models regardless of the pattern matching and learning rule are non-parametric and thus highly reliant on the quality and clarity of input data and any preprocessing [3]. Intelligent decisions rely not only on the optimal mapping between the input features and the output actions, but also on the careful selection of those inputs. Including too many unfiltered features may introduce irrelevant or conflicting attributes that impair performance [51].

Most prior existing neural modeling research focuses on discovering variations on the internal mappings between given features and the desired output and applying them to a financial problem (e.g., [6, 19, 23, 25]); this paper seeks to synergistically combine the two disciplines to solve a common problem. This is especially true in a financial domain, where the number of financial indicators as potential features is continually expanding and the number of potential combinations quickly approaches infinity [39]. The naïve approach to providing high-quality data is to preprocess the data by exhaustively evaluating candidate sets of features and all their combinations. However, exhaustive evaluation is infeasible and provides no insight into how features should be considered in a model of human trader decision making. The attention resources of humans are quite limited (e.g., [38, 41]). Extensive research shows that young adult humans with intact prefrontal cortices effectively focus their limited attention on a select subset of the incoming stimuli while discarding distracters (e.g., [10, 14, 54]). This exhibits a form of selective attention [20, 45]. In practice, a professional financial analyst will focus on a relatively small set of key features to determine a course of action [24, 34, 42].

Research specifically targeted at carefully selecting and filtering the features before adding them to financial neural models is relatively sparse [52]. The majority of studies either rely on unfiltered input features or on preselected features identified by expert human experience, which introduces intergenerational data dredging [37]. More advanced methods apply genetic algorithms (e.g., [21, 22, 47]) and stepwise regressive techniques (e.g., [40]). Genetic algorithms rely on random variation to discover successful feature parameters [15]. This likely provides a weak approximation—at best—of human-like selective attention. The stepwise approach typically first incorporates all input features into the model unfiltered and iteratively prunes individual features. This implausibly strains known biological limitations to attention resources and tends to overlook features that may function optimally in conjunction with others.

While research in the precise biological mechanism of selective attention is ongoing [20], one approach as delineated by professional traders is to ensure the features are uncorrelated so as to avoid multicollinearity issues [4, 42]. If the trader uses correlated and similarly behaving features to confirm each other, it is a false confirmation. The two features will necessarily agree on particular signals. Using both adds no new information while potentially introducing bias. Therefore, there may be some utility in preprocessing to focus on non-correlated features and to discard the remainder as distracters.

3 Data collection and preprocessing methods

Simulation datasets are derived from 1,000 days (4 years) of adjusted closing prices for each of five randomly selected blue chip stocks from the Dow Jones Industrial Average index: American Express (AXP), Bank of America (BAC), Coca-Cola (KO), IBM (IBM), and Walmart (WMT). Closing price adjustments correct for stock splits, provide a common base price for comparison over time, and ignore dividend information. Each method uses an initial interval of 750 trading days (3 years) for training, feature selection, and parameter selection (validation), and a final 250 trading-day interval (1 year) for testing. Each stock for each method constitutes a separate, independent simulation, with no data shared between stocks. This allows cleaner analysis of the feature selection on the stocks and better replicates the trader's process of tracking individual stock behavior [42].

Relying on the adjusted prices, the dataset includes ten moving averages for each stock. A *moving average* is the average of past prices in a trailing window for the price time series. The moving-average window sizes in this study range from one to ten trading days. Combining the price with each moving average generates ten features based on the moving average crossover rule. Each feature contains a sparse series of bullish *buy* and bearish *sell* indicators based on whether the price crossed from under to over the moving average (buy) or from over to under the moving average (sell) (Fig. 2). If the price did not cross the moving average, the feature indicates a neutral signal (no trade). A "feature" corresponds to one of the moving-average time intervals (1–10 days), and a feature selection method bases trading decisions on the *buy* and *sell* recommendations of a chosen subset of these moving averages. Ten features produce $2^{10} - 1 = 1{,}023$ possible feature subsets to explore, less the empty set. This underlies the difficulty of optimal feature selection, an NP-complete problem (e.g., [2]).

While all approaches in this paper can use the commonly applied raw prices or raw moving averages as features (e.g., [28, 47]), such data do not provide any readily interpretable trading rules by itself for analysis for comparison with the machine learning. The same argument applies to simple moving average differences from the price (e.g., [26]). For example, it becomes difficult to manually provide an objective, readily justifiable reason why a raw price of \$50 or a price \$2 below its moving average should constitute a buy or sell. In contrast, the moving average crossover is simple, discrete, readily justifiable, and provides each feature with an objective measurement while containing some indication of a change in environment. Also, all possible feature values are conveniently circumscribed. The range of all future possible values in out-of-sample data is necessarily known and always positive, which satisfies all known machine-learning methods without needing modification.

For purposes of comparing feature selection methods, this study treats a *buy* indicator as buying \$10,000 worth of stock at the close of that day and selling at the closing price 1 day later. A *sell* indicator is simulated as shorting \$10,000 worth of stock and covering 1 day later. In all simulations, the cost of each buy/sell trading pair is set at 0.1%, for example, \$10

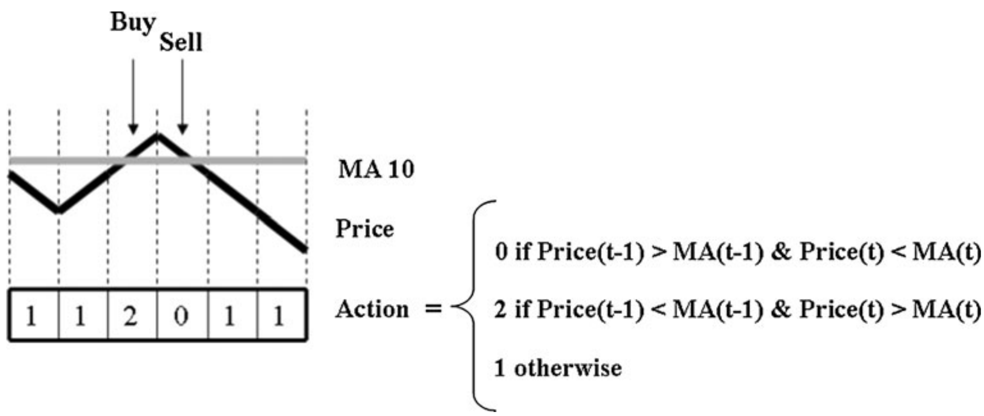


Fig. 2 This example shows how the closing price (Price, *dark line*) crossing with the ten-day moving average (MA 10, *light gray line*) generates the feature vector component for the ten-day moving average crossover for 6 days. Buy (2) occurs when the price crosses from below to above the moving average. Sell (0) occurs when the price crosses from above to below the moving average. Neutral indicators (1) occur when there is no crossover. The feature vector is a series of predictions using different moving average crossovers that can be compared with the 1-day ground truth price change. In this example, the sell (0) recommendation was a good one, while the buy (2) recommendation was not

per realized trading position. This figure derives from a typical sample commission rate of $5 per trade available from TradeKing (http://www.tradeking.com/) and constant, non-compounded simulated entry position values of $10,000 each. This study ignores additional risks such as liquidity and bid/ask spread risks due to the Dow Jones Industrial Average index size and volume.

Each method is tested on its ability to maximize rates of return with active trading on out-of-sample test data of 1 year (2007) after training and validating on 3 years of data (2004–2006. In addition, the entire simulation is repeated twice more on 1-year walk-forwards of 1 year (i.e., train and validate on 2005–2007 and test on 2008; train and validate on 2006–2008 and test on 2009). Therefore, there are 3,750 total testing days. This is sufficient to establish a 95% confidence data sample [18], while still being small enough to allow detailed and meaningful analysis of each time step.

4 CARTMAP and four alternative approaches

For comparison with the novel CARTMAP, the alternatives include: (1) an industry standard random walk [16, 32, 35] that ignores all features; (2) multiple linear regression; (3) general-purpose ARTMAP that fits all features; and (4) ARTMAP with stepwise selection.

4.1 Random walk

Random walks are Monte Carlo simulations that sample the underlying market behavior. Each trading day during the test period has random *buy* (5%), *sell* (5%), or *no trade* (90%) orders. There is no training period since there is no trading logic on each decision besides randomness. This paper averages the results from five separate random walks for each stock.

4.2 Regression

Multiple linear regression is a basic form of a commonly used decision-making tool used in financial analysis [3, 5, 11, 46, 49, 55]. Its decision weighting structure conforms to a neural network with no hidden layers. Its basic form sums the weighted input features directly toward a particular output decision. For this paper, MATLAB provides basic linear least squares multiple regression calculations.

4.3 Adaptive Resonance Theory: general-purpose default ARTMAP

Adaptive Resonance Theory models the interaction between top-down expectation and bottom-up input in memory to focus attention. Attended features, in turn, constitute the coded memory pattern of each learned category. Unsupervised Adaptive Resonance Theory (ART) [7] and supervised ARTMAP neural networks [8, 9] exhibit fast, stable, and incremental learning. The version of ARTMAP implemented in this paper is the general-purpose default ARTMAP 2 [1], which trains in winner-take-all mode and predicts in distributed mode. Winner-take-all learning partitions the training set into disjoint clusters, each represented as a *critical feature pattern* in memory. During testing, activation distributed across coding nodes allows a novel input to make an output class prediction based on partial evidence across multiple learned categories. When ARTMAP inputs include all available features during learning, the network adaptively adjusts the contribution that each feature makes to the net system input. The learning system hereby carries out a form of internal, category-specific feature selection, even without a priori feature selection applied to the external input.

Since this paper emphasizes model comparisons and minimizing data dredging issues [52] over maximizing pure output results, there is no validation or adjusting of internal network parameters. Also for this reason, the most recent default version of ARTMAP provides a baseline, general-purpose platform for testing an off-the-shelf neural network on financial data.

4.4 ARTMAP with stepwise feature selection

An ARTMAP feature selection method [40] evaluates the internal structure of the network trained on inputs that specify all feature values. The method then identifies a subset of these features that are critical to the network's predictive success. A new ARTMAP network is then trained on the selected feature subset. ARTMAP feature selection orders input features according to their differential power or ability to separate different output classes. If knowing the value of one feature alone were sufficient to identify the classes of all training inputs, then that feature would have the maximum possible rank. A feature would have the lowest possible rank if the span of its values were the same across all classes. More generally, the degree of overlap of class-specific feature-value intervals indicates how well each feature differentiates a given class from all other output classes, producing an index of the differential power of each feature.

On the financial prediction problem, feature selection proceeds as follows. A validation subset comprising the final year of the 3-year training set is reserved for feature selection. This subset was not used for training the network that produced the differential power indices. First, ARTMAP predicts the output class of each validation set exemplar, given an input that specifies only the value of the one feature with the greatest differential power. Network performance is recorded, and this feature is added to the selected set. Next, ARTMAP predicts output classes with inputs that specify feature values from the selected set plus the value of the feature with the next greatest differential power. If adding the new feature improves performance, this feature is added to the selected set. By iteration, the algorithm assesses the marginal predictive utility of each feature, producing a selected feature subset. Note that a given feature that is itself a good predictor may be rejected because its information is redundant with respect to features already in the subset. On the other hand, a feature that is by itself a poor predictor may be selected if it carries some useful information not already in the subset.

After selecting the features, ARTMAP trains on all 3 years of data using only selected features.

4.5 Clustered ARTMAP: CARTMAP

The novel clustered ARTMAP (CARTMAP) combines the clustering action of Fuzzy ART with the default, general-purpose ARTMAP. There are three basic steps to CARTMAP: (1) use Fuzzy ART [8] to cluster the features by similarity, generating a fixed number of clusters; (2) select at most one feature per feature cluster; (3) train ARTMAP on those selected features. Fuzzy ART can be regarded as the unsupervised form of ARTMAP since it relies on the same matching kernel. While ARTMAP relies on supervised labels, Fuzzy ART requires a free vigilance parameter to determine the threshold by which two vectors are considered sufficiently similar to be clustered. Instead of clustering similar sample days using behaviors over the ten features, this paper transposes the training set for clustering such that it clusters similar features using behaviors over the sample days. In this manner, similar features are features whose behaviors are highly correlated over the training dataset. Per Fig. 1, when the price crosses over the 3-day moving average, it is highly likely it has also crossed the 2-day moving average. A clustering approach like Fuzzy ART can help quantify these correlations and separate the feature set into clusters. The free vigilance parameter may be manually adjusted to increase or decrease the number of resulting clusters. This paper ensures there are between 3 and 5 clusters for each stock to best replicate the professional trader experience of 3–5 features [42].

While there may be randomized approaches to step (2), this paper uses a feature-ranking scheme adapted for ARTMAP models [40] to determine the most contributing feature in each cluster. New ARTMAP neural models are then trained on each possible combination of these most contributing features using the full training set. If the number of clusters were fixed at 3, there would always be at most $2^3 - 1 = 7$ combinations to evaluate (less the empty set), regardless of increases in the number of original features. This improves scalability and addresses complexity concerns.

5 Results and discussion

Figure 3 shows the aggregate averages with full trading costs for the five randomly selected stocks for the Dow Jones Industrial Average over all approaches over all test years.

Sharpe ratios [44] provide a simple, objective measure for comparing between alternative models. All models employ an absolute return strategy. That is, the baseline for each model is a zero rate of return, regardless of the underlying market index since a perfect transaction-free hedging produces a net rate of zero. The random walk

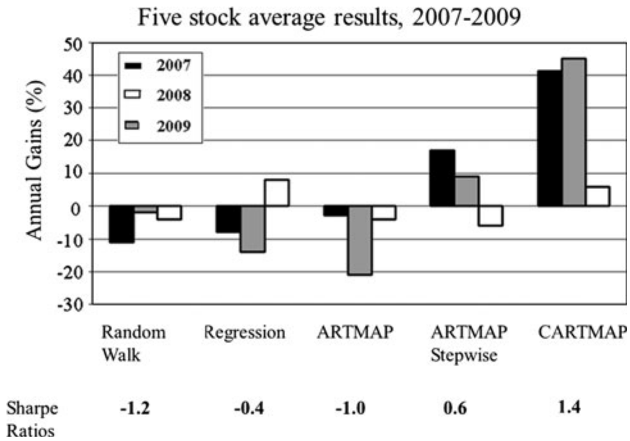


Fig. 3 The annualized out-of-sample test results for testing years 2007, 2008, and 2009, all performed with independent training and validation data. CARTMAP compares favorably with four different alternatives: random walk, multiple regression, ARTMAP, and ARTMAP with stepwise selection. In addition, simple buy-only (buying) and simple short-only (shorting) approaches represent the indices for additional reference. These results all include trading costs. Sharpe ratios are included as a simple objective measurement

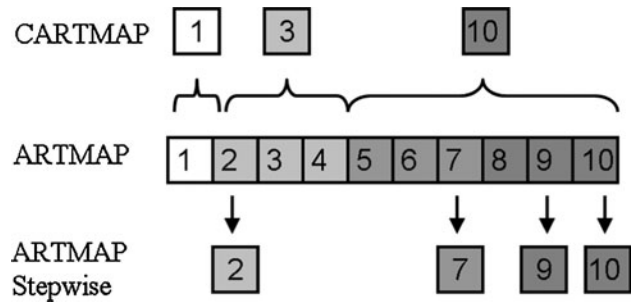


Fig. 4 Sample selected features by model on American Express over the training period 2004–2006. The ten features are ordered 1…10 and *shaded* to demonstrate correlations. *Closer shades* indicate closer relative correlation. ARTMAP (at *center*) uses all ten features. Regression follows the same rule, but is omitted for clarity. Adjacent features tend to generate noisy inputs as they are closely related. The ARTMAP with stepwise selection (*bottom*) iteratively selects features, in this case retaining features 2, 7, 9, and 10. However, features 7, 9, and 10 are still closely related and their combination is likely to continue to generate noise in the test period. CARTMAP (at *top*) divides the ten features into three separate clusters and evaluates only one feature per cluster. This ensures the processing is focused on less correlated features to reduce noise. Selection results shown here are typical of results across the other five stocks and other test years

randomly hedges (i.e., balances buying and shorting) or does not trade and its average should naturally be zero less the trading costs. The total transaction costs paid by the random walks were similar to the average total paid by the alternative approaches. Regression and ARTMAP use all ten features to produce net annual losses and negative Sharpe ratios during the test periods. Adjacent features are by design highly correlated. For example, the 3-day moving average crossover is highly correlated with the 2-day moving average crossover. Including all ten features would produce potential biases and unfiltered noise that impairs prediction performance. The regression and general-purpose neural models typically automatically adjust the weighted influence for each feature but cannot internally detect and filter out redundant or conflicting features. The annualized results are not significantly different from that of a random walk.

Enhancing ARTMAP with a non-clustering stepwise selection provides some improvement. The average annual gain and Sharpe ratio become positive, though still not significantly different from random walk's at 95% confidence. The filtering of the features by the stepwise process depends solely on training and validation performance and is not specifically targeted at removing correlated features. Therefore, the selected features may still include significant confounding effects.

CARTMAP shows a significantly higher annual rate of gain than the alternatives while not being accompanied by a commensurately high variance. The Sharpe ratio shows the performance is significantly different from the random walk. This is most likely because CARTMAP automatically ensures that none of the features are highly correlated with each other. These results show that CARTMAP's dimensionality reduction by removing correlated features and selecting an optimal combination from the remainder can improve the risk-adjusted rate of return over the market and related alternatives. The clustering via biologically inspired unsupervised learning algorithms loosely mirrors reported expert trader techniques [42].

These results warrant three additional lines of inquiry: (1) a closer examination of the specific features each model selected and their effects; (2) whether the CARTMAP results remain true for additional stocks; and (3) why the CARTMAP results for 2009 differed markedly from those of 2007–2008. Figure 4 shows the breakdown of the sample feature selection.

In the extreme case, incorporating near-identical features (e.g., Fig. 4, features 9 and 10) adds very little incremental information value to the decision system while erroneously effectively doubling the influence of one feature and forming a permanent confirmation bias. CARTMAP automatically detects correlated features and divides them such that only non-correlated features can be considered.

It should be noted that the CARTMAP algorithm as presented currently does not filter out negatively correlated features. Fully negatively correlated features provide fully opposite behavior. A neural network would be able to correct for the direction, making the fully negatively correlated features behave identically. CARTMAP as presented would still indicate the features were not positively correlated. Negative correlations do not appear in the moving average crossover features. Including the complement coded mirror images of each feature into the clustering (e.g., Fuzzy ART) step and retaining the closer match of each mirror image pair can enhance CARTMAP to detect negative correlations.

Figure 5 addresses both the expansion of the stock sample to the entire Dow Jones Industrial Average index (left) and a short foray into using shorter training periods as a standard technique to compensate for heteroskedastic data (right). Transaction costs are included.

The three test years (2007, 2008, and 2009) use the prior 3 years as training in each case, with three simulations per each of the 30 stocks. Results appear similar to that of Fig. 3. This implies that the results achieved with the sample of five stocks could be replicated with the entire index. Note that the test results for 2009 are again markedly weaker when compared with the results for 2007–2008. This is likely due to the strong mismatch between the training period of 2006–2008 and the test period of 2009. When CARTMAP traded in 2009, it was biased towards shorting activity from its training while 2009 was a strong year for buying. This is a well-known problem with non-stationary time series. The training period may not be representative of the testing period if the characteristics of the time series shifted over time.

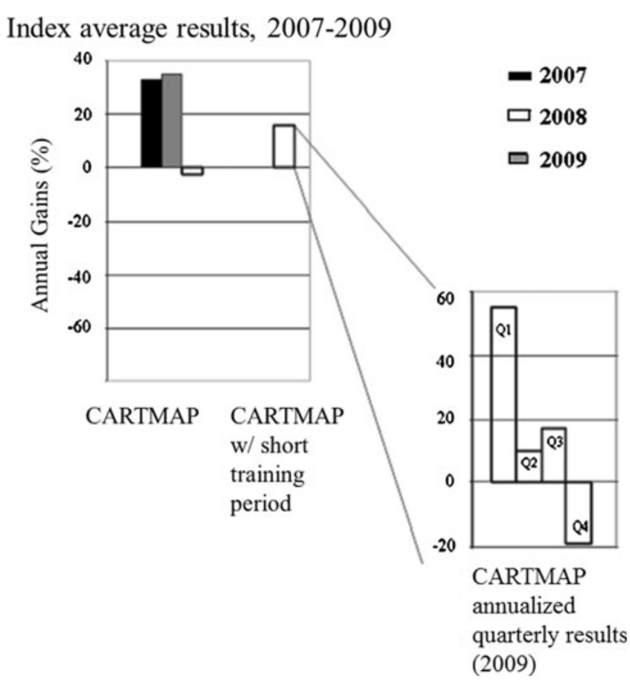


Fig. 5 The annualized out-of-sample test results for testing years 2007, 2008, and 2009, all performed with independent training and validation data on the entire Dow Jones Industrial Average index. Transaction costs are included. At right is a detailed view of CARTMAP's average performance over short training (3 quarters) and testing (1 quarter) in four one-quarter walk-forward simulations for 2009

Intuitively, the standard technique of forcing the neural network to train on a smaller window of data closer in time—and presumably more similar—to the test data should improve the testing results (e.g., [17]). Figure 5 (right) shows that on average, CARTMAP trained on a shorter dataset could produce higher average rates of return of highly time varying data. To obtain this average, the training and testing data for this demonstration across 30 stocks is shortened from 4 years to four quarters, while keeping the same training and testing proportions.

Nevertheless, the last of the four quarters tested in this manner still provided negative results. The return variation was also higher due to lesser amount of training data, ceteris paribus. The Sharpe ratio falls to 0.6. In addition to feature selection techniques, much more research is required to compensate for time-varying, non-stationary data. While there exists research on these topics (e.g., [53]), more may be needed in terms of determining training window size, testing performance decay, and context sensitivity in pattern recognition.

6 Conclusions

In making investment decisions, there is a plethora of systems, models, advisors, and indicators available that together provide virtually limitless amounts of information. A first step then is to decide on which information sources to follow and in so doing, convert information to actionable intelligence. This is feature selection. While most automatic feature selection techniques are purely validation result driven and fall on the spectrum between parsimony and performance, this paper presents a novel feature selection with CARTMAP that combines both simplicity and results. CARTMAP's flexibility and ability to both eliminate correlated features and evaluate combinations of the remainders could demonstrate a different approach to feature selection that emulates human input and experience. Future studies will address increasing the feature space, evaluating additional classes of features, exploring scalability concerns vis-à-vis traditional feature selection techniques, and enhancing simulation techniques on highly time varying data.

Acknowledgments This research was supported by CELEST, an NSF Science of Learning Center (SBE-0354378).

References

1. Amis G, Carpenter G (2007) Default ARTMAP 2. In: Proceedings of the international joint conference on neural networks (IJCNN, 07), Orlando, Florida, pp 777–782
2. Aragones E, Gilboa I, Postlewaite A, Schmeidler D (2005) Fact-free learning. Am Econ Rev 95:1355–1368

3. Bishop C (2006) Pattern recognition and machine learning. Springer, New York
4. Blalock HM Jr (1963) Correlated independent variables: the problem of multicollinearity. Soc Forces, University of North Carolina Press
5. Bodyanskiy Y, Popov S (2006) Neural network approach to forecasting quasiperiodic financial time series. Eur J Oper Res 175:1357–1366
6. Brockett P, Golden L, Jang J, Yang C (2006) A comparison of neural network, statistical methods, and variable choice for life insurers' financial distress prediction. J Risk Insur 73(3):397–419
7. Carpenter G, Grossberg S (1987) A massively parallel architecture for a self-organizing neural pattern recognition machine. Comput Vis Graph Image Process 37(54):115
8. Carpenter G, Grossberg S, Reynolds J (1991) ARTMAP: supervised real-time learning and classification of nonstationary data by a self-organizing neural network. Neural Netw 4:565–588
9. Carpenter G, Grossberg S, Rosen D (1991) Fuzzy ART: fast stable learning and categorization of analog patterns by an adaptive resonance system. Neural Netw 4:759–771
10. Chao L, Knight R (1998) Contribution of human prefrontal cortex to delay performance. J Cogn Neurosci 10:167–177
11. Chartered Financial Analyst Institute (2010) CFA program curriculum. CFA Institute, Pearson
12. Chen W, Shih J (2006) Comparison of support-vector machines and back propagation neural networks in forecasting the six major Asian stock markets. Int J Electron Finance 1(1):49–67
13. Chen W, Shih J, Wu S (2006) Comparison of support-vector machines and back propagation neural networks in forecasting the six major Asian stock markets. Int J Electron Finance 1(1):49–67
14. Conway A, Cowan N, Bunting M (2001) The cocktail party phenomenon revisited: the importance of working memory capacity. Psychon Bull Rev 8:331–335
15. Duda R, Hart P, Stork D (2001) Pattern classification. Wiley-Interscience, New York
16. Fama E (1970) Efficient capital markets: a review of theory and empirical work. J Finance 25(2):383–417
17. Hamilton J (1994) Time series analysis. Princeton University Press, Princeton
18. Higgins J (2004) Introduction to modern nonparametric statistics. Brooks/Cole-Thomson Learning, Pacific Grove
19. Kaastra I, Boyd M (1996) Designing a neural network for forecasting financial time series. Neurocomputing 10:215–236
20. Kane M, Engle R (2002) The role of prefrontal cortex in working-memory capacity, executive attention, and general fluid intelligence: an individual-differences perspective. Psychon Bull Rev 9(4):637–671
21. Kim K (2006) Artificial neural networks with evolutionary instance selection for financial forecasting. Expert Syst Appl 30:519–526
22. Kim K, Han I (2000) Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. Expert Syst Appl 19(2):125–132
23. Kirkos E, Spathis C, Manolopoulos Y (2007) Data mining techniques for the detection of fraudulent financial statements. Expert Syst Appl 32:995–1003
24. Kirkpatrick C, Dahlquist J (2006) Technical analysis: the complete resource for financial market technicians. FT Press, Upper Saddle River
25. Kumar K, Bhattacharya S (2006) Artificial neural network vs linear discriminant analysis in credit ratings forecast. Rev Account Finance 5(3):216–227
26. Kwon Y, Moon B (2007) A hybrid neurogenetic approach for stock forecasting. IEEE Trans Neural Netw 18(3):851–864
27. Leung C, Tsoi A (2005) Combined learning and pruning for recurrent radial basis function networks based on recursive least square algorithms. Neural Comput Appl 15:62–78
28. Lina C, Huang J, Gena M, Tzeng G (2006) Recurrent neural network for dynamic portfolio selection. Appl Math Comput 175(2):1139–1146
29. Lintner J (1965) The valuation of risk assets and selection of risky investments in stock portfolios and capital budgets. Rev Econ Stat 47:13–37
30. Liu B, Cui Q, Jiang T, Ma S (2004) A combinational feature selection and ensemble neural network method for classification of gene expression data. BMC Bioinform 5:136. doi: 10.1186/1471-2105-5-136
31. Lo A (2001) Bubble, rubble, finance in trouble? J Psychol Financial Mark 3:76–86
32. Lo A (2007) The efficient markets hypothesis. In: Blume L, Durlauf S (eds) The new Palgrave dictionary of economics. Palgrave Macmillan, New York
33. Lo A, Repin D (2002) The psychophysiology of real-time financial risk processing. J Cogn Neurosci 14(3):323–339
34. Loeb G (1996) The battle for investment survival. Wiley, Malden
35. Malkiel B (1973) A random walk down Wall Street. W. W. Norton & Company, New York
36. Markowitz H (1952) The utility of wealth. J Political Econ 2:151–158
37. McQueen G, Thorley S (1999) Mining fool's gold. Financial Analysts J 55(2):61–72
38. Miller P, Seier W (1994) Strategy utilization deficiencies in children: when, where, and why. In: Reese H (ed) Advances in child development and behavior, vol 25. Academic Press, Salt Lake City
39. Murphy J (1999) Technical analysis of the financial markets: a comprehensive guide to trading methods and applications. New York Institute of Finance, New York
40. Parsons O, Carpenter G (2003) ARTMAP neural networks for information fusion and data mining: map production and target recognition methodologies. Neural Netw 16:1075–1089
41. Posner M, Rothbart M, Thomas-Thrapp L, Gerardi G (1998) Development of orienting to locations and objects. In: Wright R (ed) Visual attention. Oxford University Press, New York
42. Schwager J (1995) The new market wizards: conversations with America's top traders. Wiley, Hoboken
43. Sharpe W (1965) Capital asset prices: a theory of market equilibrium under conditions of risk. J Finance 19(3):425–442
44. Sharpe W (1994) The sharpe ratio. J Portfolio Manag 21(1):49–58
45. Siegler R, Alibali M (2005) Children's thinking. Prentice Hall, Upper Saddle River
46. Thawornwong S, Enke D (2004) The adaptive selection of financial and economic variables for use with artificial neural networks. Neurocomputing 56:205–232
47. 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 Syst Appl 27(3):417–425
48. Wang K (2006) Neural network approach to vibration feature selection and multiple fault detection for mechanical systems. Innov Comput Inf Control 3(30):431–434
49. West D (2000) Neural network credit scoring models. Comput Oper Res 27:1131–1152
50. Williams J (1938) The theory of investment value. Harvard University Press, Cambridge
51. Witten I, Frank E (2005) *Data mining*. Elsevier, San Francisco
52. Wong C, Versace M (2011a) Rethinking neural networks in financial decision-making studies: seven cardinal confounds. In: Global conference on business and finance proceedings, Las Vegas, Nevada

53. Wong C, Versace M (2011) Echo ARTMAP: context sensitivity with neural networks in financial decision-making. Glob J Bus Res 5(5):27–43
54. Woods D, Knight R (1986) Electrophysiological evidence of increased distractibility after dorsolateral prefrontal lesions. Neurology 36:212–216
55. Yu L, Wang S, Lai K (2008) Neural network-based mean–variance–skewness model for portfolio selection. Comput Oper Res 35:34–46
56. Zhu X, Wang H, Xu L, Li H (2008) Predicting stock index increments by neural networks: the role of trading volume under different horizons. Expert Syst Appl 34:3043–3054