An Explicit Feature Selection Strategy for Predictive Models of the S&P 500 Index

Tim Chenoweth[1,2,3] Zoran Obradović[1,] *

[1] School of Electrical Engineering and Computer Science

[2] Department of Management and Systems

[3] Department of Economics

Washington State University, Pullman WA 99164-2752

Abstract

The focus of this study is the selection of an appropriate set of features for a feed forward neural network model used to predict both future market direction and future returns for the S&P 500 Index. The experimental results provide evidence that the proposed feature selection process may result in a more successful prediction model. However, the study also indicates that the problem domain may need to be limited to predicting monthly instead of daily movements. In addition, the proposed process could be more useful for predicting the future market direction rather than actual returns.

1. Introduction

While the application of neural networks to financial forecasting is beginning to receive academic attention [Freedman 1995], the issue of feature selection for financial forecasting problems has been largely ignored. Feature selection refers to choosing a subset of parameters (or features) from a larger pool of input information (technical and/or fundamental indicators) for designing a prediction system in a manner that preserves as much of the original information as possible. This issue is important because an appropriate combination of most significant features leads to faster computation and require fewer training examples for successful generalization as compared

*Research sponsored in part by the NSF research grant NSF-IRI-9308523.

to the use of all available features [Battiti 1994]. This is especially important for gradient-based neural network algorithms because they are computationally very expensive and do not scale up well [Orfanidis 1990]. Therefore it is important to reduce the feature set as much as possible while minimizing the predictive information loss.

The focus of this study is the development of a preprocessing procedure for the selection of a set of features used to predict future movements in the S&P 500 Composite Index. The selected features are used in a K-nearest neighbor classifier and in a feed forward neural network model utilizing the back propagation learning method. The objectives are to address both feature and model selection for different problem domains (monthly versus daily data, predicting market direction versus predicting returns). Previously, an interesting neural network architecture called SupNet has been applied to feature selection for currency exchange rate prediction problem [Tenorio 1993]. In the SupNet the training data is clustered into a predetermined number of groups and a distance to cluster centers is used to classify the test data. The total number of misclassifications on the test set is used to assign a numerical penalty values to individual features and groups of features clustered together. A sequential elimination technique (backward selection) is used to remove the features that are assigned large penalties. The algorithm stops when none of the feature subsets are assigned a penalty smaller that its parent.

Although quite interesting, the SupNet feature selection process is potentially fragile as it is based on a single selection technique (the SupNet algorithm) and a single selection measure (distance to cluster centers). Our study is proposing a potentially more robust feature selection process based on an integration of several combinations of the selection techniques and selection criteria. Sections 2.1 and 2.2 provide the details of the feature selection techniques and criteria used in this study. Section 2.3 discusses the proposed feature selection process and Section 2.4 describes the neural network model and learning process. Results and analysis for separate experiments using monthly and daily data are presented in Sections 3.1 and 3.2.

2. Methodology

This study considers financial forecasting both as a market direction problem (ie. predicting the value of a discrete variable) and a returns prediction problem (ie. predicting the value of a continuous variable). In the market direction prediction problem, the class one corresponds to a future positive move in the S&P Composite Index (ie. the value of the index increased during the period) while class two corresponds to a future negative move (ie. the index decreased in value during the period. For the returns prediction problem the actual future return for the next period is predicted, where the return is defined as the percentage increase/decrease in the index value for the future period.

2.1. Selection Techniques

In predicting the market direction, a feature selection technique is implemented as a search algorithm that attempts to determine a subset of the existing features which maximizes the differences between the two classes just described based on performance criteria explained in Section 2.2. This section briefly describes the selection techniques used in the proposed feature selection process. More details can be found in Fukunaga [1990].

The *Best Feature* selection technique orders the features from the best to the worst based on the utilized selection criterion. The *Sequential Forward Search* selection technique begins with an empty feature set and adds features to it one at a time. The first feature added is the one deemed to be the best according to the selection criterion. The next feature added is the one which results in the largest improvement when considered in conjunction with the first feature. Similarly, the i^{th} feature added is the one that results in the largest improvement when considered in conjunction with the previous $i - 1$ features. The *Sequential Backward Search* selection technique is similar to the sequential forward search, except that the initial set contains all the features, and features are removed from this set one at a time. The first feature removed is the one that results in the smallest degradation when the remaining features are considered together. This process repeats until the feature set reaches a predetermined size.

Each of the above selection techniques has its strengths and weaknesses. The Best Feature technique is very fast and as such is suited for a quick, rough partition of a large feature set. However, choosing the j features based only on the Best Feature technique is not likely to give satisfactory results as this technique considers features by themselves rather than in combination with other features.

Both the Sequential Forward Search and the Sequential Backward Search do consider features in combination and as such are finer grained techniques. However, both techniques are significantly more computationally expensive, demanding computing time that is quadratic in the number of features as compared to the linear time needed by the Best Feature technique. In addition, with Sequential Forward Search once a feature is added to the features set it cannot be removed. With Sequential Backward Search once a feature is removed it cannot be added later. As such, both Sequential Forward Search and Sequential Backward Search are greedy heuristics and neither can guarantee that an optimal set of features is achieved.

2.2. Selection Criteria

Feature selection is conducted considering only discrete variable prediction (the market direction problem), and the resulting feature set used for both the discrete and continuous variable prediction problems (the market direction and return prediction problems respectively). Each feature selection criterion is either an estimation of the classification error or a measure of the distance between classes. Therefore the selection criteria objective is to either maximize some measure of intra-class separation or to minimize the estimated classification error. This section describes the selection criteria used in the proposed feature selection process, all of which are described in more detail in Fukunaga [1990].

The *Estimated Minimal Error Probability* selection criterion is an estimate of Bayes error for the data set by applying the K-nearest neighbor classifier [Cover, Hart 1967] to the training set utilizing the leave one out approach. This tends to overestimate the error and as such gives a very conservative error estimate. The selection criterion then becomes finding a set of features that minimizes the estimated Bayes error.

The Euclidean, Patrick-Fisher, Mahalanobis, and Bhattacharyya distances are all means of measuring the multidimensional separation between two disjoint classes of data. The *Euclidean* distance is defined as

$$\sqrt{(M_2 - M_1)^T (M_2 - M_1)}, \tag{1}$$

where M_1 and M_2 are vectors whose components are the average values for each feature in data classes one and two respectively. For example, for data sets with 2 features, M_1 and M_2 would each have two components, $M_1 = (m_1^1,\ m_2^1)$, $M_2 = (m_1^2,\ m_2^2)$, where m_1^1 is the average value of the first feature for all data belonging to class one and m_2^1, m_1^2, and m_2^2 are defined similarly.

The *Patrick–Fisher* distance is defined as

$$\left(\frac{\Sigma_1 + \Sigma_2}{2}\right)^{-1} (M_2 - M_1), \tag{2}$$

where M_1 and M_2 are as previously defined and Σ_1 and Σ_2 are the covariance matrices for data classes one and two respectively. The *Mahalanobis* distance is defined as

$$(M_2 - M_1)^T \left(\frac{\Sigma_1 + \Sigma_2}{2}\right)^{-1} (M_2 - M_1), \tag{3}$$

where M_1, M_2, Σ_1 and Σ_2 are defined as above. And finally, the *Bhattacharyya* distance is defined as

$$\frac{1}{8}(M_2 - M_1)^T \left(\frac{\Sigma_1 + \Sigma_2}{2}\right)^{-1} (M_2 - M_1) + \frac{1}{2} \ln \frac{\left|\frac{\Sigma_1 + \Sigma_2}{2}\right|}{\sqrt{|\Sigma_1||\Sigma_2|}}. \tag{4}$$

where the first term in equation (4) measures the class separability due to the mean difference, while the second term measures the class separability due to the covariance-difference.

The *selection criteria* used in the proposed feature selection process is to maximize the separation between the market direction classes based on any of the above distance measures.

2.3. The Proposed Feature Selection Process

Depending on the size of the initial feature set, either a one or two phase feature selection process is proposed. If the initial feature set is relatively small, it may be possible to reduce it to the final set in one phase using Sequential Forward and Sequential Backward Search techniques. However, both techniques require computational time that is quadratic in the number of features being searched. For large feature sets the amount of computational work required may be reduced by application of a two phase approach. The first phase quickly reduces the initial feature set to an intermediate set of manageable size using the Best Feature approach. Then in the second phase the finer but more expensive Sequential Forward and Sequential Backward Selection techniques can be applied to the intermediate feature set to derive the final set. The first phase reduction is accomplished in computational time linear to the number of features in the initial feature set. The second phase takes computational time that is quadratic in the number of features remaining after the first phase reduction.

In addition to computational efficiency, another important issue is how to deal with instability problems. A feature selection procedure is *unstable* if a small change in the data used in the selection process results in drastic changes in the selected feature set. Recently, it was suggested by Breiman [1994] that unstable procedures can be stabilized by averaging the results from several different feature selection processes. Therefore, to reduce the level of instability this study is performing feature selection using several combinations of the selection techniques and selection criteria discussed earlier. The obtained partial results are integrated using either a frequency process or a ranking process explained as follows.

Frequency Process

The proposed *frequency process* counts the number of times a feature appears in the "top m features set" as determined by various combinations of selection techniques and criteria. A specific selection technique and performance criterion (ie. Sequential Forward Search attempting to maximize the Bhattacharyya distance) is used to determine a set of top m features. A counter for each of the m features selected in this manner is incremented by one and the process repeated using a different selection technique and selection criterion combination. When the feature selection process is completed for all combinations of the selection techniques and criteria, the counter for each feature contains the frequency with which that feature appeared in the set of "top m features." The reduced feature set comprises the m features with the largest frequency values.

Ranking Process

An alternative to the frequency process is to use a *ranking process*, where a specific selection technique and selection criterion combination is used to determine a rank ordering of the features from best to worst with scores assigned to the features based on this ordering (one is the best, p is the worst, where p is the total number of features). The process is repeated using other selection technique and selection criterion combinations and the scores summed. The reduced feature set is then comprised of the m features with the lowest scores.

2.4. The Learning Process

The stock market is modeled using a single hidden layer feed forward neural network with back propagation learning [Rumelhart, et. al 1986]. The input layer units correspond to the pattern features, while the value of the output layer unit indicates either future market direction for the discrete variable prediction problem or the predicted return for the continuous variable prediction problem. For both the discrete variable prediction problem and the continuous variable prediction problem the input patterns are the same; what changes is the target output value. For the discrete prediction problem a target value of one corresponds to an increase in the value of the index and minus one corresponds to a decrease in the value. For the continuous prediction problem the target value is the actual index return (percentage increase/decrease in the index) for the next period.

The learning scheme used in this work consists of a sequence of training/prediction sessions where the ANN is retrained after each session using more recent information. This is achieved

Table 1: ANN Parameter Values Used in All Experiments

by training the ANN using patterns from a fixed size window covering a continuous time segment of historic data. The target value for the ANN is either the market direction (for the discrete prediction problem) or the actual return (for the continuous prediction problem) for the time unit immediately following the training window. After a single prediction step the training window is shifted forward one time unit (ie. one trading period), the patterns from the new window are used to retrain the ANN, and a prediction is made for the next time unit. This process is repeated until the data set is exhausted. It is important to note that the ANN is retrained after every prediction. In other words the ANN is trained and one prediction is made. The window is shifted forward one period (which means the test pattern for the previous window is now in the training set) and another prediction made. The process is repeated until the data is exhausted.

3. Experimental Results and Analysis

The static ANN parameters used in all experiments discussed in this paper are shown in Table 1. The dynamic parameters are described in the specific experiment descriptions. For all experiments the data values were scaled into the (-1,1) range.

For monthly experiments, the 170 months of data from January 1973 to February 1987 comprised the initial training set with actual predictions (both discrete and continuous) made for the 70 month period from March 1987 to December 1992. For daily experiments, the initial training window was comprised of either 250 patterns (December 28, 1987 to December 16, 1988) or 500 patterns

(December 29, 1986 to December 16, 1988). For all daily experiment (both discrete and continuous) test prediction were made for the 1,273 patterns from December 19, 1988 to December 31, 1993.

The prediction performance metrics used in the experiments are the *annual rate of return* (ARR), the *directional symmetry* (DS), and the *sharpe ratio* ($Sharpe$). Following Hutchinson [1993], the annual rate of return is defined as

$$ARR = \frac{k}{n} \sum_{i=1}^{n} r_i, \tag{5}$$

where n is the total number of trading periods, k is the number of trading periods per year (k = 12 for monthly data, $k = 253$ for daily data), and r_i is defined as

$$r_i = \begin{cases} |r_i^a| & \text{if the market direction is correctly predicted} \\ -|r_i^a| & \text{otherwise} \end{cases} \tag{6}$$

where r_i^a is the actual return for the S&P 500 index for period i. For comparison purposes the simple buy and hold ARR for the monthly test data is 8.76% and for the daily test data is 11.49%. The ARR for an ideal model with perfect prediction ($r_i = r_i^a$ for all $1 \leq i \leq n$) is 42% for monthly data and 147% for daily data.

Following Caldwell [1995], the directional symmetry metric is defined as

$$DS = \frac{100 \sum_{i=1}^{n} d_i}{n} \tag{7}$$

where

$$d_i = \begin{cases} 1 & \text{if } (p_i)(r_i^a) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

The variable p_i is defined either as the predicted return for period i (for continuous variable prediction experiments) or as

$$p_i = \begin{cases} 1 & \text{if S\&P index increases in value over period } i \\ -1 & \text{otherwise} \end{cases} \tag{9}$$

for the predicted market direction experiments (discrete variable predictions). In addition to measuring the DS on the set of all n predictions, in our experiments the DS is also computed for sets of large and small market movements. The *overall prediction set* (All) is the set of all 2-tuples (p_i, r_i^a), $1 \leq j \leq n$, for a specific ANN experiment consisting of n trading periods. The *small prediction set* ($Small$) is the subset of the All 2-tuples (p_i, r_i^a) where r_i^a is below the median value. More precisely, $Small$ is defined as $Small = \{(p_i, r_i^a) : r_i^a < h\}$ where constant h is selected s.t. $\#\{r_i^a : r_i^a < h\} = \#\{r_i^a : r_i^a > h\}$. Similarly, the *large prediction set* ($Large$) is the subset of the All 2-tuples (p_i, r_i^a) where r_i^a is above the median value, defined more precisely as $Large = \{(p_i, r_i^a) : r_i^a > h\}$.

The Sharpe ration is a widely used metric [Caldwell 1995] that attempts to normalize returns according to the risk of the trading strategy. The sharp ration is defined as

$$Sharpe = \frac{\sum_{i=1}^{n} r_i^a - \sum_{i=1}^{n} r_i^f}{n\sigma_a} \tag{10}$$

where r_i^a is defined previously, σ_a is the standard deviation of the elements in the set $\{r_i^a : 1 \leq i \leq n\}$, and r_i^f is the risk free return for period i. For all experiments in this study the r_i^f is the return on U.S. Treasure Bills for period i.

3.1. Monthly Data Experiments

The complete monthly data set $m1$ contains 29 features. A second feature set $m2$ is comprised of seven features from the complete set selected based on an intuitive feel for which features would make a good classifier. The final monthly feature sets $m3$ and $m4$ were constructed by selecting eight features from the complete set based on the formal feature selection techniques and criteria presented in Sections 2.1 and 2.2. A list of all 29 monthly features and their set membership in $m1$, $m2$, $m3$ and $m4$ appears in Table 4 in the appendix. The ANN architecture parameters in common with all experiments in this study are displayed in Table 1 while the number of input units (UI) and hidden units (HU) for different monthly experiments are shown in Table 2.

The features that comprise $m3$ and $m4$ were selected using the frequency process and the rank process respectively, both of which are discussed in Section 2.3. To obtain $m3$ and $m4$ all combinations of the selection techniques and performance criteria discussed in Sections 2.1 and 2.2 are used for individual independent feature selection experiments. For example, Sequential Forward Search selection is performed first using the Estimated Minimal Error Probability criterion, then Mahalanobis Distance, etc. The frequency with which a feature appeared in the "seven best features set" in individual feature selection experiments was computed and the eight features with the highest frequency scores collected in $m3$. The initial objective was to have the same number of features as the intuitive set $m2$ (seven); however eight were included because of a frequency score tie between the seventh and eighth features. The features in $m4$ were selected by first scoring each feature in $m1$ as discussed in the rank process portion of Section 2.3 and selecting the eight features with the lowest scores.

K-Nearest Neighbor Experiments

First, the quality of the feature selection process was tested using a simple K-nearest neighbor (KNN) classifier [Cover, Hart 1967]. For each of the four data sets ($m1$, $m2$, $m3$, $m4$) three KNN experiments were conducted with K equal first to one, then two, and finally three. The average DS results for the three experiments are displayed in Figure 1 for each data set along with the high DS value representing the best result for the three experiments.

A comparison of the average DS of the KNN classifier for all four data sets indicates that the feature selection process did reduce $m1$ to $m3$ and $m4$ with no significant reduction in performance (in fact performance increased). The KNN classifier also indicates that $m4$ results in a better classifier than $m3$, indicating that the ranking process works better than the frequency process. In addition, Figure 1 shows a decrease in performance for $m2$ indicating that the intuitive feature set ($m2$) contains little predictive information.

Figure 1: KNN for Discrete Prediction on Monthly Data

ANN Discrete Prediction Experiments

Next, for each monthly feature set the following ANN experiments were conducted. First the problem was treated as a discrete variable prediction problem where the goal was to predict the market direction. As shown in Table 2, the results support the KNN finding that $m1$ was reduced to $m3$ and $m4$ with little or no reduction in performance (somewhat smaller ARR achieved with smaller number of trades). In addition, $m3$ and $m4$ indicated better performance for the DS metrics when compared to $m1$ as evident from Figure 2. It is important to note, in contrast to the KNN system, that an ANN trained using intuitively selected $m2$ data had better results than one trained using all $m1$ data. This indicates that the ANN was able to detect relationships the KNN classifier could not.

Comparing the ARR values to those achieved using the buy and hold strategy (8.76%) shows that the ANN was able to achieve returns comparable to the buy and hold strategy. However, it is important to note that transaction costs were not considered in this study. Including transaction costs would undoubtedly result in the buy and hold strategy having a superior ARR in all instances.

ANN Continuous Prediction Experiments

In the final set of experiments on monthly data, for each monthly feature set the goal was to predict the actual market return (continuous variable prediction problem) for the following month. As shown in Table 2 and Figure 3, the $m4$ based ANN predictor did better than $m3$ on ARR, $Sharpe$, and DS metrics and also better than the buy and hold, indicating that the ranking

Figure 3: DS for Continuous Variable Prediction Using Monthly Data

feature selection process transfers better to a prediction domain than the frequency feature selection process. Therefore, the daily data experiments reported in Section 3.2 use only the ranking selection process.

3.2. Daily Data Experiments

The complete daily data set $d1$ contains 67 features. The features that comprise the reduced feature set $d3$ were selected using the two phase process discussed in Section 2.3. In phase one the 67 features in the initial set $d1$ were reduced to the 30 features in the intermediate set $d2$. This involved the use of the Best Feature selection technique along with the Estimated Minimal Error Probability, Bhattacharyya distance, and Euclidean distance criteria. The size of the intermediate feature set was limited to thirty. It was desirable to include as many features in $d2$ as possible. However, with more than 30 features the feature selection techniques used in phase two become very computationally expensive. In phase two the 30 features in $d2$ were reduced to the seven features in $d3$. This was accomplished using the Sequential Forward and Sequential Backward Search selection techniques along with the same criteria as used in phase one. The individual selection technique and performance criteria scores were integrated using the ranking process described in Section 2.3. A list of all 67 monthly features and their membership in sets $d1$, $d2$, and $d3$ are shown in Tables 5 and 6. The ANN architecture parameters in common with all experiments in this study are displayed in Table 1 while the number of input units (UI) and hidden units (HU) for different daily experiments are shown in Table 3.

K-Nearest Neighbor Experiments

The daily problem was first treated as a discrete variable prediction problem where the goal was to predict the direction of the S&P 500 index. For each of the three data sets ($d1$, $d2$, $d3$) three KNN experiments were conducted with K equal first to one, then two, and finally three. The average DS results for the three experiments are displayed in Figure 4 for each data set along with the high DS value representing the best result for the three experiments. The KNN classifier results

Figure 4: KNN for Discrete Prediction Using Daily Data

show an increase in the overall directional symmetry (the *All* set) from $d1$ to $d2$ and from $d2$ to $d3$, indicating that both phases of the feature selection process improved the classification capabilities of the system.

ANN Discrete Prediction Experiments

The initial ANN experiment uses a training window size of 250 patterns. The ARR results, illustrated in Table 3, show improvement from the ANN trained using $d1$ to the one trained using $d2$. However, the improvements from reducing $d2$ to $d3 - 250$ (ie. data set $d2$ with training window of size 250) noticed in the KNN classifier were not evident in the ARR for the ANN. The $Small$ and $Large$ data set DS values shown in Figure 5 indicate that there was no significant difference in the performance of the the three feature sets. One possible explanation is that the training window size of 250 patterns was to small. Therefore an additional experiment using $d3$ and a training window size of 500 (ie. d3-500) was conducted to determine if a larger window size improved the results. As shown in Figure 5, the DS results using a larger training window was uniformly better across all measures. However, in no instance was the ANN able to achieve a better ARR (Table 3) than that achieved by the buy and hold strategy (11.49%).

ANN Continuous Prediction Experiments

Finally the problem was treated as a continuous variable prediction problem in which the goal was to predict the actual market return for the next day. Initially these experiment also used a window size of 250 patterns. Results for these experiments were universally worse than the results

Figure 5: DS for Discrete Variable Prediction Using Daily Data

for the discrete variable prediction problem and as such are not worth presenting. An additional experiment using $d3$ and a training window size of 500 was conducted to determine if a larger window size improved the results. The results (ARR =-4.64%, $Sharpe$ =-0.042, and DS values 50.75, 51.49, and 50.00 for sets All, $Small$, and $Large$ respectively) were not significantly different from those achieved with a training window of size 250 and much worse than those achieved for the discrete variable prediction problem.

4. Conclusions

The results from this study provide evidence that some form of explicit feature selection should be considered in determining the feature set used in building a predictive model of the S&P 500 index. While a feature set based on the experience and intuition of the developer may seem reasonable on the surface, it might actually contain very little predictive information and should be verified using formal feature selection techniques. In this study, the intuitive features and the formally selected reduced feature set had no features in common. For the KNN classifier, the reduced feature set performed much better than the intuitive set. However, it should be noted that the ANN based on the intuitive feature set did perform adequately.

This study indicates that a feature selection process based on a classification problem does not readily transfer to a prediction problem. Results from the continuous variable prediction problem using reduced sets were unsatisfactory for both monthly and daily experiments.

Results for the discrete variable prediction problem indicate that an ANN could be useful in predicting future stock market movements. In particular, it is interesting to note that the ANN appeared to perform satisfactorily when predicting movements greater than the median return. However, the study does indicate that the problem domain may need to be limited to predicting monthly movements, not daily movements, and that the ANN could be more useful to predicting future market directions rather than actual returns. The results provide partial evidence that the feature reduction approaches presented here may result in a more successful prediction model.

Acknowledgements

The authors are greatful to an anonymous reviewer for the suggested modifications and fast handling of the manuscript.

References

Battiti, R., [1994] "Using Mutual Information for Selecting Features in Supervised Neural Net Learning," *IEEE Trans. Neural Networks*, Vol 5, No. 4, pp. 537-550.

Breiman, L., [1994] "The Heuristics of Instability in Model Selection," Technical Report No. 416, Statistics Department, University of California, Berkeley.

Caldwell, R., [1995] "Performance Metrics for Neural Network-based Trading System Development," *NeuroVe$t Journal*, Vol. 3 Num 2, pp. 13-23.

Cover, T.M. and Hart, P.E., [1967] "Nearest Neighbor Pattern Classification," *IEEE Trans. Inform. Theory*, Vol. IT-13, pp. 21-27.

Freedman, R (Editor)., [1995] *Proceedings of the Third International Conference on Artificial Intelligence Applications on Wall Street*, New York, NY.

Fukunaga, K., [1990] *Introduction to Statistical Pattern Recognition,* Academic Press, New York, NY.

Hutchinson, J., [1993] *A Radial Basis Function Approach to Financial Time Series Analysis,* PhD Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA.

Orfanidis, S. J., [1990] "Gram-Schmidt Neural Nets," *Neural Computation,* Vol. 2 , pp. 116-126.

Rumelhart, et. al., [1986] *Parallel Distributed Processing: Explorations in the Microstructure of Cognition,* Vols. 1 and 2, MIT Press, Cambridge, MA.

Tenorio, T. and Hsu, W., [1993] "Selecting Indicators for Improved financial Prediction," *NeuroVe$st Journal,* Nov/Dec, pp. 16-20.

Tim Chenoweth (tchenowe@eecs.wsu.edu) received a B.S. degree in Mathematics in 1981 from the Coast Guard Academy, a M.B.A in Finance from Washington State University in 1991, and is currently completing a M.S. in Computer Science and an Individual Interdisciplinary Ph.D. combining Business and Computer Science, both from Washington State University. He was an active duty officer in the Coast Guard from 1981 to 1989. The objective of his current research is to use advanced technologies to model financial markets.

Zoran Obradovic (zoran@eecs.wsu.edu) received the B.S. degree in Applied Mathematics, Information and Computer Sciences in 1985; the M.S. degree in Mathematics and Computer Science

in 1987, both from the University of Belgrade; and the Ph.D. degree in Computer Science from the Pennsylvania State University in 1991. He was a systems programmer at the Department for Computer Design at the Vinca Institute, Belgrade, from 1984 to 1986, and has been a research scientist at the Mathematical Institute of the Serbian Academy of Sciences and Arts, Belgrade, since then. At present, he is an Assistant Professor in the School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA 99164-2752, USA. The objective of his current research is to explore applicability of neural networks technology to large scale classification and time series prediction problems in very noisy domains.

Table 4: Complete Set of Features Used in Monthly Experiments

Table 5: Top Thirty Features Used in Daily Experiments

Table 6: Remaining Features Used in Daily Experiments