Available online at www.sciencedirect.com
SCIENCE DIRECT®

Decision Support Systems 37 (2004) 567–581

Decision Support Systems

www.elsevier.com/locate/dsw

Neural network techniques for financial performance prediction: integrating fundamental and technical analysis

Monica Lam*

Management Information Science, College of Business Administration, California State University, Sacramento, 6000 J Street, Sacramento, CA 95819-6088, USA

Available online 12 July 2003

Abstract

This research project investigates the ability of neural networks, specifically, the backpropagation algorithm, to integrate fundamental and technical analysis for financial performance prediction. The predictor attributes include 16 financial statement variables and 11 macroeconomic variables. The rate of return on common shareholders' equity is used as the to-be-predicted variable. Financial data of 364 S&P companies are extracted from the CompuStat database, and macroeconomic variables are extracted from the Citibase database for the study period of 1985–1995. Used as predictors in Experiments 1, 2, and 3 are the 1 year's, the 2 years', and the 3 years' financial data, respectively. Experiment 4 has 3 years' financial data and macroeconomic data as predictors. Moreover, in order to compensate for data noise and parameter misspecification as well as to reveal prediction logic and procedure, we apply a rule extraction technique to convert the connection weights from trained neural networks to symbolic classification rules. The performance of neural networks is compared with the average return from the top one-third returns in the market (maximum benchmark) that approximates the return from perfect information as well as with the overall market average return (minimum benchmark) that approximates the return from highly diversified portfolios. Paired *t* tests are carried out to calculate the statistical significance of mean differences. Experimental results indicate that neural networks using 1 year's or multiple years' financial data consistently and significantly outperform the minimum benchmark, but not the maximum benchmark. As for neural networks with both financial and macroeconomic predictors, they do not outperform the minimum or maximum benchmark in this study. The experimental results also show that the average return of 0.25398 from extracted rules is the only compatible result to the maximum benchmark of 0.2786. Consequentially, we demonstrate rule extraction as a postprocessing technique for improving prediction accuracy and for explaining the prediction logic to financial decision makers.

Keywords: Neural networks; Financial performance; Forecasting; Technical analysis; Fundamental analysis; Postprocessing techniques; Data mining; Rule extraction

1. Introduction

Neural networks have become a popular tool for financial decision making [2,5,16–18,25,33,35,36,43,45]. There are mixed research results concerning the ability of neural networks to predict financial performance. Due to a variety of research design and evaluation criteria, it is difficult to compare the results of different studies [9,27]. Past studies in this area are subject to several problems. First, time horizons for experiments are short. When time hori-

* Tel.: +1-916-278-7037; fax: +1-916-278-6757.
E-mail address: lamsm@csus.edu (M. Lam).

doi:10.1016/S0167-9236(03)00088-5

zons are short, experimental results may be tampered by situational effect and economic fluctuations. Second, sample sizes are small. When sample sizes are small, experimental results may be biased and cannot be generalized to the future. Third, many studies do not investigate the statistical significance of performance differences. Because variance is a significant factor in investment, ignoring performance variance in the forecasting process is undesirable at least [3,6]. Fourth, the selection of predictor attributes in past studies is based on either fundamental or technical analysis. Fundamental analysts believe that an investment instrument has its intrinsic value that can be derived from the behavior and performance of its company [7,8,14,22,28,29,34,40,42]. The fundamental approach utilizes quantitative tools, mainly the financial ratios compiled from financial statements as well as qualitative indicators, such as management policy, marketing strategy, and product innovation, to determine the value of an investment instrument. Technical analysts, on the other hand, believe that the trends and patterns of an investment instrument's price, volume, breadth, and trading activities reflect most of the relevant market information a decision maker can utilize to determine its value [20,24,37]. Instead of analyzing fundamental information about companies, the technical approach tries to identify turning points, momentum, levels, and directions of an investment instrument, using tools such as charting, relative strength index, moving averages, on balance volume, momentum and rate of change, breadth advance decline indicator, directional movement indicator, and detrended price oscillator. There are divergent opinions about what other trends in the macroeconomic, political, monetary, and societal sentiment spheres should be incorporated in technical analysis [4]. It is the purpose of this research project to address the above four problems for neural network as a data mining tool for financial forecasting. This project applies neural networks to a sample of 364 S&P companies for the period of 1985–1995. We attempt to present a formal study on the complex phenomenon of financial performance using company financial and macroeconomic data as predictor variables, neural networks as the data mining tool, and rate of return on common shareholders' equity as the to-be-predicted variable. Paired t tests are adopted to verify the statistical significance of performance differences between neural networks and the market's top performers as well as overall averages.

We believe that neural networks are an excellent tool for forecasting financial performance for the following reasons. First, neural networks are numeric in nature, which is especially suitable for processing numeric data such as financial information and economic indicators. The numeric nature of neural networks is in contrast to symbolic manipulation techniques such as ID3 [26] and AQ [21], which were designed for processing nominal variables. Because numeric data must be converted into nominal values before they can be used as input to symbolic manipulation techniques, there are the problems of losing information, inappropriate data intervals, and different conversion methods leading to different mining results. Neural networks, on the other hand, can accept numeric data directly as input for mining purposes. Second, neural networks do not require any data distribution assumptions for input data. This feature allows neural networks be applicable to a wider collection of problems than statistical techniques such as regression or discriminant analysis. Third, neural networks are an incremental mining technique that permits new data be submitted to a trained neural network in order to update the previous training result. In contrast, many symbolic manipulation and statistical techniques are batch-oriented, which must have both new and old data submitted as a single batch to the model, in order to generate new mining results. For financial applications with new data being available from time to time, neural networks can accommodate new information without reprocessing old information. Fourth, neural networks are model-free estimators. This feature allows interaction effect among variables be captured without explicit model formulations from users [13,45]. Basically, the more hidden layers in a neural network, the more complicated the interaction effect can be modeled.

Although neural networks as a data mining tool have the above merits, they have their fair share of problems. One common difficulty for neural network applications involves the determination of the optimal combination of training parameters including the network architecture (the number of hidden layers and the number of hidden nodes), the learning rate,