2M−3

# Neural Networks for Time-Series prediction; Stock Exchange Forecasting

**Alexandra Ioana Cristea and Toshio Okamoto**
Lab. Prof. T.Okamoto, Artificial Intelligence(AI) Info-Systems Graduated School(IS)
University of Electro-Communication, 1-5-1 Chofugaoka, Chofu, Tokyo 182, Japan
E-mail: alex@ai.is.uec.ac.jp

## 1 Introduction

This study attempts to simplify the complexity of the systems present in the economical world and to create a handy tool for economists, that usually deal with little mathematics (mostly linear functions, that can hardly be a correct approximation of real world behaviour) and a lot of "common sense". It is the authors belief that this "common sense" can be structured, organized and made reproducible by a computer.

Most of the economical functions can be represented as Time Series(TS). Stock Exchange(SE) events are representable as TS (see [Ankenbrand *et al.*, 1995]), which can be forecasted with certain accuracy with Neural Networks(NN) (see [Gas *et al.*, 1993]). In [Cristea *et al.*, 1996] a forecasting theory based on NNs was developed, for general TS as well as for the particular case of Stock Exchange(SE), examining the components: *Trend, Cyclus, Season* and *Irregular events*. NNs were used for their parallel processing power and ability to learn three of the four components mentioned above. For the forth, an economy based calculus was proposed.

This theory was implemented here with help of a program with Motif interface and the results were studied.

In [Komo *et al.*, 1994] a similar forecasting tool was developed. Stock Market values are analized, divided into a training set and a test set, and presented to a NN tool. Differences exist though in the data-window size, both for inputs and outputs. While [Komo *et al.*, 1994] works with a 1-prediction-step, the current study aims at a prediction for a variable number of future values. Also, input data is not only from the previous step, but considers the history of the TS.

Therefore, data input has a three level hierarchy: stock market prices for training, a previous weights matrix for new data (testing) and current stock market prices,for active forecasting (prediction).

The learning method in [Komo *et al.*, 1994] is based on sequential data input ("epoch") while the current study proposes the more efficient bootstrapping method.

Some partial results that are available look promising, although the study is not yet completed. The main difference from previous systems consists in the usage of both mathematical and economical rules for the system construction.

## 2 System

The implemented system can perform **training** (learning) through weight computing, **forecasting** with a *Hetero-Associative Network (HAN)* (see [Cristea *et al.*, 1996]) and serve as an **user-interface**. These processes are implemented as independent programs. The communication between these processes is assured by the common resources, which are files, but can also be share memory resources or TCP/IP based internet information exchange resources. The reason why this was not implemented at the time-being is mainly our concern with theoretical aspects at the moment. Also for comparative results of different forecasting tools only similar hardware is needed, so a sequential tool serves this purpose as well as any. Further on different parallel implementational aspects can be considered for speed increase, but they present no theoretical interest. According to a flexible design, the input files can be given to the system as fixed-format files, or can be input from the window interface. The three independent processes build a three level data hierarchy, as can be seen in fig.1. The interaction

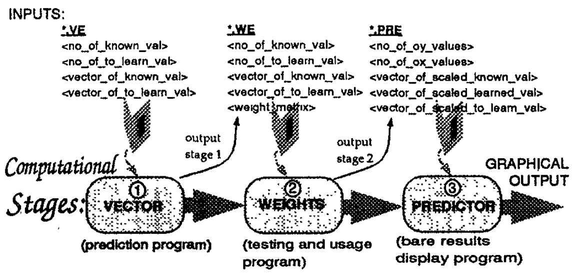


Figure 1: The 3 input levels

of the user with these levels is shown in fig.2.

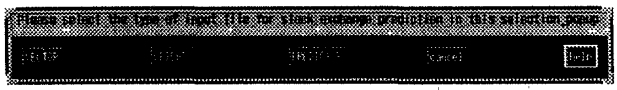

**Figure 2: The 3 input levels, selectable from windows**

The **data** are values (prices on SE market) over any desired period of days and are represented by the user-interface program on a screen on the 0y axis (see fig.3). The 0x axis represents the time. The data are scaled for a better representation and for a uniformization of input data for the learning algorithm, so they range from [0-min; 1-max]. The known, predicted and true future values (if available) are represented on the same screen.

The used data were fictious values that present a high degree of irregularity. Real-world data were not available yet.

The **learning process** is supervised gradient-descent based Lyapunov(see [Cristea *et al.*, 1996]). The computed forecasted data are compared to the true future values. Triggered visualization of the learning process during training is possible.

For avoidance of over-fitting, early stopping of the learning process was considered. Some of the error results that were obtained (see fig.4) were due to extremely early stops in the algorithm. A procedure of stop-moment selection is to test the results of the training, that is to test the computed weights on the set of test values, that were not used for training ( see [Komo *et al.*, 1994]).

A full help support is provided at each step of the program usage, both for current state and for possible steps to take.

The **prediction error** is displayed in percentage to the maximal value(price) that occured in the given time-period in a "Error dispersion" window (see fig.4). Information about *Mean Square Error (MSE)* is also available to the user, although it is less informative in respect to local errors.

## 3 Results

Some partial learning results of some of the presented inputs is already available, and shows chances of further success (fig.3). The computational error display is shown, for a better comprehension, on a bidimensional display (fig.3, fig.4). SE are irregular patterns, so a boot-

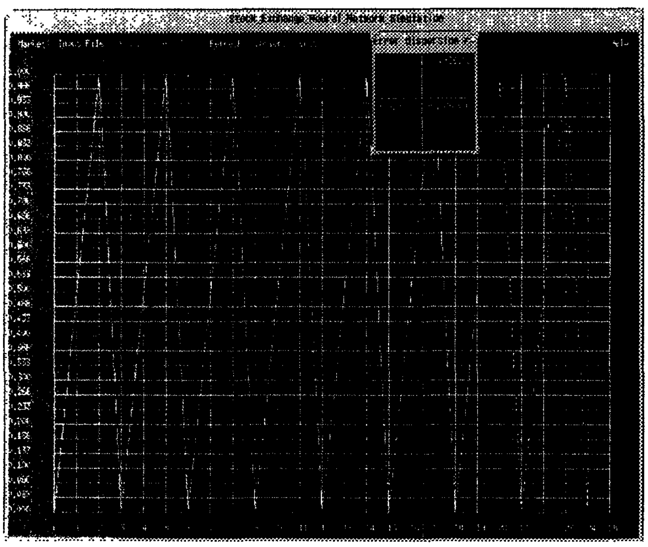

Figure 3: A "perfectly" trained net

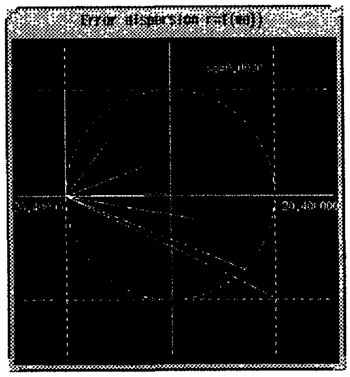

Figure 4: Error display window

strapping type of training shows a better convergence chance than the FIFO, or even pseudo-aleatory method.

## 4 Discussion

A SE forecasting tool was constructed, based on the TS behaviour of SE. Previous researches were considered, optimizations proposed. The system showed the possibility of learning with any degree of accuracy and precision, depending on the allocated time. For early stopping, errors were observed. Nevertheless, early stopping proved to be necessary in order to avoid over-fitting. The main difference from previous systems consists in the mixt solution of mathematical and economical rules ( see [Cristea *et al.*, 1996]).As at the current state of the research the comparison with former systems is purely theoretical, an implementational comparison is to be performed in near future. Also, we intend to use a real-world data set to test the viability of the presented forecasting tool.

## References

[Ankenbrand *et al.*, 1995] T. Ankenbrand and M. Tomassini. *Multivariate time series modeling of financial markets with artificial neural networks.* Artificial Neural Nets and Genetic Algorithms, Springer Verlag,Wien, pages 257-260.

[Gas *et al.*, 1993] B. Gas and R. Natowicz. *Unsupervised learning of temporal sequences by neural networks.* Artificial Neural Nets and Genetic Algorithms, Springer Verlag,Wien, pages 253-256 .

[Komo *et al.*, 1994] D. Komo, C.I. Chang and H. Ko. *Neural Network Technology for Stock Market Index Prediction.* International Symposium on Speech, Image Processing and Neural Networks, 13-16 April, Hong-Kong, IEEE, pages 543-546.

[Cristea *et al.*, 1996] A. Cristea and Toshio Okamoto. In *Proceedings of the National Conference of IPSJ, Artificial Intelligence, Osaka, Japan, 9/4-6,1996,vol. 2,*Information Processing Society of Japan, pages 2-201/2.