Stock Prediction Based on Financial Correlation

Yung-Keun Kwon
School of Computer Science & Engineering
Seoul National University
Sillim-dong, Gwanak-gu, Seoul, 151-744 Korea
kwon@soar.snu.ac.kr

Sung-Soon Choi
School of Computer Science & Engineering
Seoul National University
Sillim-dong, Gwanak-gu, Seoul, 151-744 Korea
sschoi@soar.snu.ac.kr

Byung-Ro Moon
School of Computer Science & Engineering
Seoul National University
Sillim-dong, Gwanak-gu, Seoul, 151-744 Korea
moon@soar.snu.ac.kr

ABSTRACT

In this paper, we propose a neuro-genetic stock prediction system based on financial correlation between companies. A number of input variables are produced from the relatively highly correlated companies. The genetic algorithm selects a set of informative input features among them for a recurrent neural network. It showed notable improvement over not only the buy-and-hold strategy but also the recurrent neural network using only the input variables from the target company.

Categories and Subject Descriptors

J.1 [**Computer Applications**]: Administrative Data Processing—*Financial*

General Terms

Experimentation

Keywords

Stock prediction, financial network, cross-correlation, feed-forward neural network

1. INTRODUCTION

It is a practically interesting topic to predict the trends of a stock price. Although it is not an easy job due to its nonlinearity and uncertainty, there were many trials using a variety of methods including artificial neural networks [1] [2], decision trees [3], rule induction [4], Bayesian belief networks [5], evolutionary algorithms [6] [7], classifier systems [8], fuzzy sets [9] [10], and association rules [11]. In developing a stock prediction system, one of the most important tasks is to define input variables. For example, only one-day return of a closing price of a stock was used in [12]. The difference between the price and the moving average, highest and lowest prices were also used in [13]. In addition to using a price series, volume of transactions, macro economic data and market indicators were considered for input variables [14].

Recently, many studies have been performed on the fluctuations and the correlations in stock price changes between different companies in physics communities by using concepts and methods in physics [15] [16]. They showed that stock price changes in some companies were influenced by the other companies. Thus, the cross-correlation coefficient between different companies can be an important factor to understand the cooperative trends in stock market. Recently, there have been many efforts to find the correlations in stock price changes using random matrix theories and they showed that there exist cooperative behaviors of the entire market [17] [18]. Based on the correlations, there is room for extension in the existing approache which predicts a direction of a company's stock price using the input variables generated from only its own information.

In [19] and [20], neuro-genetic hybrids for stock prediction were proposed and showed notable success on many companies. They generated 75 input variables using a variety of technical indicators for each company for the recurrent neural networks (RNN). The genetic algorithm (GA) was used to optimize weights in the neural network. In this paper, we extend the system proposed in [19] and [20] by utilizing the cross-correlation concept. To predict a company's price, we first find the other companies which are highly correlated with the company. We then collect all the input variables generated from not only the target company but also the selected companies. However, it is hard to use all of them for input nodes in the neural network since the network becomes too complex. Thus, a genetic algorithm is used to select a set of salient features among a number of input variables. Thus, it is different from the genetic algorithms used in [19] and [20].

The rest of this paper is organized as follows. In Section 2, the daily stock prediction problem and the objective are explained. In Section 3, we explain the correlation coefficient between different companies used in this paper. In Section 4, we describe our hybrid genetic algorithm for predicting the stock price. In Section 5, we provide our experimental results. Finally, conclusions are given in Section 6.

2. STOCK TRADING PROBLEM

We have a database with years of daily trading data. Each record includes daily information which consists of the closing price, the highest price, the lowest price, and the trad-

GECCO'05, June 25–29, 2005, Washington, DC, USA.

$$
\begin{aligned}
&\textbf{if } (\text{ signal is } \mathit{SELL} \text{ }) \; \{ \\
&\quad C_{t+1} \leftarrow C_t + \min(B, S_t) \times (1 - T) \\
&\quad S_{t+1} \leftarrow S_t - \min(B, S_t) \\
&\} \\
&\textbf{if } (\text{ signal is } \mathit{BUY} \text{ }) \; \{ \\
&\quad C_{t+1} \leftarrow C_t - \min(B, C_t) \\
&\quad S_{t+1} \leftarrow S_t + \min(B, C_t) \\
&\} \\
&S_{t+1} \leftarrow S_t \times \frac{x_{t+1}}{x_t}
\end{aligned}
$$

Figure 1: Investing Strategy and Change of The Property

ing volume. We name those at day t as $x(t)$, $x_h(t)$, $x_l(t)$, and $v(t)$, respectively. If we expect $x(t+1)$ is considerably higher than $x(t)$, we buy the stocks; if lower, we sell them; otherwise, we do not take any action. In [19] and [20], the problem was formulated as a kind of time-series problem as follows:

$$\frac{x(t+1) - x(t)}{x(t)} = f(g_1, g_2, \ldots, g_m)$$

where g_k's ($k = 1, \ldots, m$) are technical indicators or signals.

We have four daily data, x, x_h, x_l, and v, but we do not use them for the input variables as they are. We utilize a number of technical indicators being used by financial experts such as moving average, golden-cross, dead-cross, relative strength index, and so on [21]. As in [20], we generated totally 75 input variables using the technical indicators in this paper. One example among the input variables is

$$g_i := \frac{MA(t) - MA(t-1)}{MA(t-1)},$$

where *MA(t)* is the numerical average value of the stock prices over a period of time. The technical indicators and input variables used in this paper are explained in detail in [19] and [20].

There can be a number of measures to evaluate the performance of the trading system. Figure 1 shows the investing strategy and change of property at day $t+1$ according to the signal at day t of the trading system. In the figure, C_t and S_t mean the cash and stock balances at day t ($t = 1, \ldots, N$), respectively. We start with C, i.e, $C_1 = C$ and $S_1 = 0$. In the strategy, the constant B is the upper bound of stock trade per day and T is the transaction cost. The transaction cost was set to 0.3% in this work. This corresponds to the situation of online trading in Korea stock market. We have the final property ratio P as follows:

$$P = \frac{C_N + S_N}{C_1 + S_1}.$$

3. FINANCIAL CORRELATION BETWEEN COMPANIES

There were some studies on correlation in stock prices of different companies to understand the cooperative behaviors in stock market [22]. In this paper, we applied the cross-correlation used in the previous studies to construct a financial network.

Let $x_i(t)$ be the closing price at day t of a company i. The return of the stock-price after a time interval Δt is defined

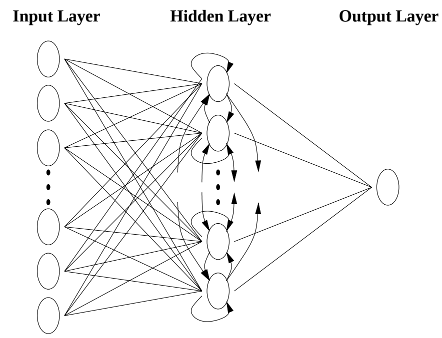

Figure 3: The Recurrent Neural Network Architecture

as

$$r_i(t) = \ln x_i(t + \Delta t) - \ln x_i(t).$$

Then, the cross-correlation between company i and j is given by

$$c_{ij} := \frac{\langle r_i r_j \rangle - \langle r_i \rangle \langle r_j \rangle}{\sqrt{(\langle r_i^2 \rangle - \langle s_i \rangle^2)(\langle r_j^2 \rangle - \langle s_j \rangle^2)}}$$

where $\langle \cdot \rangle$ means a temporal average over the given period. Then, $c_{ij} \in [-1, 1]$ and $c_{ij} = 1(-1)$ means that two companies are completely correlated (anti-correlated), while $c_{ij} = 0$ means that they are uncorrelated.

We examined the cross-correlations between 196 companies in Korea Stock Exchange (KSE) during the 4-year period 2000-2003. Figure 2 shows five companies having the highest absolute value of cross-correlation with each of two companies: SAMSUNG Electronics which is one of the largest electronic company in the world, and SAMSUNG Fire & Marine Insurance which is the largest indemnity insurance company in Korea. Most of the selected companies are in the similar category of business with the target company or the affiliates belonging to the same conglomeration.

4. THE PROPOSED SYSTEM

4.1 Artificial Neural Network

We use a recurrent neural network architecture which is a variant of Elman's network [23]. It consists of input, hidden, and output layers as shown in Figure 3. Each hidden unit is connected to itself and also connected to all the other hidden units. The network is trained by a backpropagation-based algorithm.

Only one node exists in the output layer for $\frac{x(t+1)-x(t)}{x(t)}$. On the other hand, the number of input nodes varies according to the feature selection of the genetic algorithm which will be described in Section 4.2.

4.2 Feature Selection Genetic Algorithm (FSGA)

To predict a company, K companies having the highest absolute value of the cross-correlation described in Section 3 are selected. (The company itself is necessarily selected because $c_{ii} = 1$.) Then, there are totally $75 \times K$ candidate input variables. The genetic algorithm selects a set of salient variables among them.

We use a parallel GA. It is a global single-population master-slave [24] and the structure is shown in Figure 4. In

- Example 1
  - **SAMSUNG Electronics** : An electronic company producing mobile phones, TFT-LCD, semiconductors, digital TV, and so on.
  - Highly correlated or anti-correlated companies
    * **SAMSUNG Electro-Mechanics** : A manufacturer producing key electronic components such as MLCC, PCB boards, chip registers, and so on. It is an affiliated company of the same group as SAMSUNG Electronics.
    * **DONGBUANAM Semiconductor** : A semiconductor providing CMOS wafer foundry service.
    * **SAMSUNG Techwin** : A company producing semiconductor system, optics, digital cameras, national defense program, and so on. It is an affiliated company of the same group as SAMSUNG Electronics.
    * **SAMSUNG SDI** : A company producing flat display, PDP, LCD, Lithium-ion Battery, and so on. It is an affiliated company of the same group as SAMSUNG Electronics.
    * **TRIGEM Computer** : A company producing personal computers.
- Example 2
  - **SAMSUNG Fire & Marine Insurance** : A company managing insurance against loss.
  - Highly correlated or anti-correlated companies
    * **SAMSUNG Securities** : A security corporation which is an affiliated company of the same group as SAMSUNG Fire & Marine Insurance.
    * **LG Insurance** : A insurance company of LG group.
    * **DASHIN Securities** : A security corporation.
    * **LG Invest & Securities** : A security corporation of LG group.
    * **HYUNDAI Marine & Fire Insurance** : A insurance company.

Figure 2: Cross-Correlation Example of a Company

this neuro-genetic hybrid approach, the fitness evaluation is dominant in running time. The backpropagation-based algorithm trains the network with a set of training data to evaluate an offspring. We distribute the load of evaluation to the clients (slaves) of a Linux cluster system. The main genetic parts locate in the server (master). When a new RNN is created by crossover and mutation, the GA passes it to one of the clients. When the evaluation is completed in the client, the result is sent back to the server. The server communicates with the clients in an asynchronous mode. This eliminates the need to synchronize every generation and it can maintain a high level of processor utilization, even if the slave processors operate at different speeds. This is possible because we use a steady-state GA which does not wait until a set of offspring is generated. All these are achieved with the help of MPI (Message Passing Interface), a popular interface specification for programming distributed memory systems. As shown in Figure 4, the process in the server is a parallel variant of traditional steady-state GA. In the following, we describe each part of the GA.

- *Representation:* We represent a chromosome by a one-dimensional binary vector of size $75 \times K$, where each bit of the vector means whether the corresponding input variable is included or not. It is a typical approach of genetic algorithms for feature selection [25].
- *Selection, crossover, and mutation:* Roulette-wheel selection is used for parent selection. The offspring is produced by 5-point crossover. The mutation operator flips each bit with a low probability. All these three operators are performed in the server.
- *Evaluation:* After crossover and mutation, the offspring is evaluated by backpropagation. Its result provides the quality of the selected set of input variables. As mentioned, it is performed in the client and the result is sent back to the server.
- *Replacement and stopping criterion:* The offspring first attempts to replace the more similar parent to it. If it fails, it attempts to replace the other parent and the most inferior member of the population in order. Replacement is done only when the offspring is better than the replacee. The GA stops if it does not find an improved solution for a fixed number of generations.

5. EXPERIMENTAL RESULTS

We tested our approaches with the stocks of 91 companies in KSE. We evaluated the performance for 3 years from 2001 to 2003. We got the entire data from YAHOO[1]. The GA was trained with two consecutive years of data and validated with the third year's. The solution was tested with the fourth year's data. This process was shifted year by year.

Table 1 shows the experimental results. The values mean the final property ratio P defined in Section 2. In the table, *Hold* is *buy-and-hold* strategy which buys the stock at the first day and holds it all through the year. *RNN* is the recurrent neural network using only 75 input variables of the target company for input nodes. *FSGA* is the feature selection GA described in Section 4 and the number of candidate companies to provide the input variables was set to five in

[1]http://quote.yahoo.com

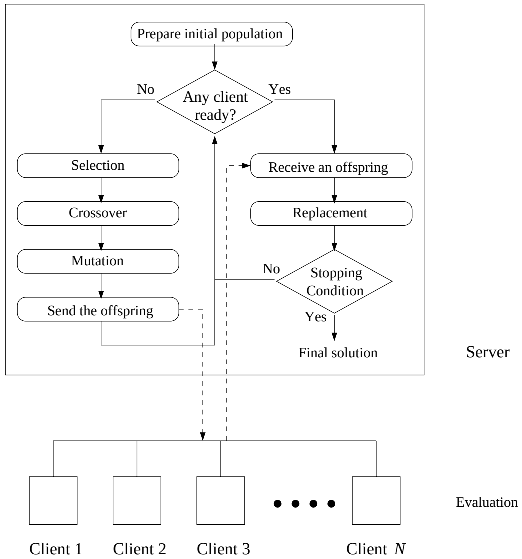

Figure 4: The Framework of the Parallel Genetic Algorithm

this paper. The minimum and the maximum number of features were set to 30 and 100, respectively, in this experiment. The results are the average results over 10 trials.

For a quantitative comparison, we summarized the relative performance in Table 2. It represents the relative performance of each approach over the *buy-and-hold* strategy. Since there are 91 companies tested for 3 years, we have 273 cases. In the table, *Up*, *Down*, and *NC* represent the situation of the stock market in each year. The *Up* and *Down* mean that the closing price has risen or fallen, respectively, against the year's starting price by 5% or more. *NC* means no notable difference. *Better* and *Worse* mean the number of cases where the P value of the learned strategy was at least 5% higher or lower than that of the *buy-and-hold*, respectively. *ANN* performed better than the *buy-and-hold* in 104 cases, worse in 110 cases, and comparable in 59 cases. On the other hand, *FSGA* performed better than the *buy-and-hold* in 136 cases, worse in 83 cases, and comparable in 54 cases. *FSGA* showed a notable performance improvement over not only *buy-and-hold* but also *ANN* on average. Specially, the performance improvement of *FSGA* over *ANN* happens in *Up* and *NC* cases. On the other hand, two approaches show similar performance in *Down*.

Figure 5 examines the cases that *FSGA* and *RNN* performed contrarily each other against *buy-and-hold*. We choose the cases where the relative performance of *RNN* was *Worse* or *Even* but that of *FSGA* was *Better*, or the reverse cases where the relative performance of *FSGA* was *Worse* or *Even* but that of *RNN* was *Better*. In the figure, y-axis is $\frac{P_{FSGA}-P_{RNN}}{P_{RNN}}$ and x-axis is P_{Hold} where P_{FSGA}, P_{RNN}, and P_{Hold} mean the property ratio of *FSGA*, *RNN*, and *buy-and-hold*, respectively. The number of cases where *FSGA* turns over the bad performance of *RNN* was 53 and the number of reverse cases was 20.

6. CONCLUSION

In this paper, we proposed a feature-selection genetic algorithm for recurrent neural networks for the stock trading. To reflect the correlation with other companies, a number candidate input variables from other companies are generated. The proposed algorithm showed significantly better performance than the "buy-and-hold" strategy and a recurrent neural network using only the input variables of the target company as input nodes.

Table 2: Relative performance over *buy-and-hold* Strategy

(1) *RNN*

	Better	*Worse*	*Even*	Total
Up	38	100	33	171
Down	59	6	13	78
NC	7	4	13	24
Total	**104**	**110**	**59**	273

(2) *FSGA*

	Better	*Worse*	*Even*	Total
Up	62	75	34	171
Down	61	5	12	78
NC	13	3	8	24
Total	**136**	**83**	**54**	273

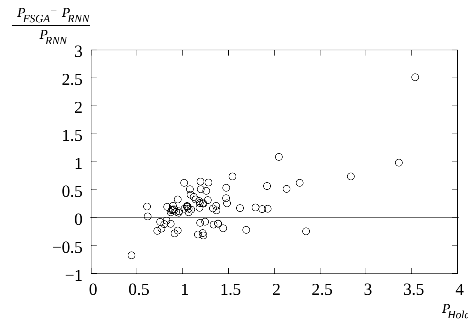

Figure 5: The Improvement of *FSGA* over *RNN*

Acknowledgments

This work was supported by the Brain Korea 21 Project. This was also partly supported by a grant of the International Mobile Telecommunications 2000 R&D Project, Ministry of Information & Communication, Republic of Korea. The ICT at Seoul National University provided research facilities for this study.

7. REFERENCES

[1] P. Tiňo, C. Schittenkopf, and G. Dorffner. Financial volatility trading using recurrent neural networks. *IEEE Transactions on Neural Networks*, 12:865–874, 2001.

[2] I. D. Wilson, S. D. Paris, J. A. Ware, and D. H. Jenkins. Residential property price time series forecasting with neural networks. *Knowledge-Based Systems*, 15(5):335–341, 2002.

[3] Y. M. Chae, S. H. Ho, K. W. Cho, D. H. Lee, and S. H. Ji. Data mining approach to policy analysis in a health insurance domain. *International Journal of Medical Informatics*, 62(2):103–111, 2001.

[4] J. A. Gentry, M. J. Shaw, A. C. Tessmer, and D. T. Whitford. Using inductive learning to predict

Table 1: P values

Symbols*	2001			2002			2003		
	Hold	*RNN*	*FSGA*	*Hold*	*RNN*	*FSGA*	*Hold*	*RNN*	*FSGA*
000060.KS	3.538	1.161	4.078	1.078	0.843	1.274	1.123	1.132	1.165
000070.KS	1.946	1.235	1.080	1.415	1.236	1.236	1.089	1.090	1.058
000140.KS	1.441	1.826	1.486	0.942	1.203	1.059	1.613	1.518	1.319
000180.KS	3.360	1.989	3.949	0.607	1.457	0.739	0.873	0.925	0.923
000210.KS	2.772	2.320	2.094	1.188	1.000	1.190	2.770	2.609	2.509
000420.KS	0.709	0.932	0.824	1.265	1.189	1.037	0.698	0.924	0.858
000430.KS	1.492	1.109	1.129	1.670	1.052	1.103	2.734	2.739	2.734
000490.KS	1.253	1.118	1.052	0.754	0.838	0.777	1.109	1.110	1.163
000830.KS	1.386	1.474	1.328	1.067	1.610	1.284	1.454	1.359	1.207
000880.KS	1.136	1.326	1.310	0.834	1.025	0.987	2.834	1.973	3.437
001120.KS	1.838	1.083	0.935	1.690	1.615	1.690	1.768	1.175	1.571
001200.KS	1.385	1.139	0.973	1.290	1.411	1.623	0.729	0.755	0.772
001300.KS	1.770	1.615	1.710	2.031	1.181	1.959	1.087	1.087	1.071
001310.KS	2.927	1.487	0.796	0.669	1.284	0.798	1.107	1.163	1.326
001450.KS	5.796	1.714	3.952	0.858	0.946	1.061	1.286	1.966	1.498
001510.KS	1.364	1.309	1.589	0.630	0.695	0.740	0.351	0.462	0.392
001720.KS	1.299	1.309	1.250	1.105	1.262	1.215	0.946	1.138	0.876
001750.KS	1.887	2.539	2.328	0.823	1.100	0.971	0.907	0.965	0.957
001880.KS	2.385	1.937	1.058	0.703	1.083	0.835	1.377	1.256	1.306
001940.KS	1.188	0.699	1.125	1.256	1.169	1.169	3.013	2.864	2.524
001970.KS	3.128	0.926	1.437	0.677	1.046	0.789	1.190	1.550	1.394
002000.KS	1.189	1.158	1.073	0.987	1.026	0.906	1.869	1.864	2.157
002020.KS	1.879	1.137	0.977	1.262	1.155	0.998	1.326	1.173	1.294
002170.KS	0.872	1.112	1.021	0.856	0.917	0.921	1.197	0.855	1.292
002320.KS	1.196	1.219	1.189	1.841	1.987	2.228	0.912	0.931	1.071
002410.KS	1.998	0.881	0.841	2.575	1.056	2.579	0.989	0.985	0.980
002550.KS	2.752	1.406	2.137	1.066	1.108	1.286	1.488	1.003	1.323
002620.KS	2.012	1.030	1.419	0.823	0.898	0.851	1.190	1.161	1.070
002700.KS	1.191	1.304	1.188	0.927	0.993	1.106	0.830	0.811	0.970
002720.KS	0.954	1.406	1.216	0.879	0.921	1.044	0.987	1.274	1.164
002790.KS	4.414	2.133	1.621	0.799	0.903	0.799	1.834	1.818	1.803
002900.KS	0.947	0.787	1.045	0.601	0.804	0.699	0.826	0.920	0.979
002990.KS	1.179	1.068	1.388	0.923	1.035	1.067	1.503	1.847	1.805
003070.KS	1.920	1.451	2.274	0.723	0.942	0.723	0.976	0.977	1.024
003120.KS	2.097	1.956	1.382	0.837	0.887	0.819	1.458	2.127	1.713
003410.KS	1.277	1.437	1.513	1.084	1.297	1.286	1.065	1.399	1.295
003450.KS	2.345	2.559	1.939	0.512	0.909	0.611	0.772	1.040	0.900
003460.KS	1.723	1.387	1.693	1.074	1.154	1.161	1.096	1.112	1.111
003480.KS	1.143	1.000	1.330	1.227	1.399	0.956	2.146	1.651	1.804
003490.KS	1.186	1.098	1.388	1.619	1.430	1.528	1.473	1.522	1.149
003530.KS	1.637	1.447	1.086	0.613	0.761	0.644	1.064	1.058	1.161
003540.KS	2.618	1.842	2.358	0.970	1.729	1.219	1.121	1.098	1.150
003600.KS	1.143	1.666	1.263	1.092	1.037	1.190	2.161	1.899	1.164
003610.KS	1.075	1.693	1.376	0.653	0.931	0.788	1.813	1.020	1.414
003680.KS	0.726	0.712	0.602	0.610	0.609	0.731	0.893	0.907	1.100
003690.KS	4.038	1.091	1.109	0.893	1.299	1.056	1.971	1.867	1.575

* If we enter it in http://quote.yahoo.com, we can get the information of the company.

bankruptcy. *Journal of Organizational Computing and Electronic Commerce*, 12(1):39–57, 2002.

[5] R. K. Wolfe. Turning point identification and Bayesian forecasting of a volatile time series. *Computers and Industrial Engineering*, 15:378–386, 1988.

[6] M. A. Kanoudan. Genetic programming prediction of stock prices. *Computational Economics*, 16:207–236, 2000.

[7] K. J. Kim. Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert Systems with Applications*, 19(2):125–132, 2000.

[8] S. Schulenburg and P. Ross. Explorations in LCS models of stock trading. In *Advances in Learning Classifier Systems*, pages 151–180, 2001.

[9] O. Castillo and P. Melin. Simulation and forecasting complex financial time series using neural networks and fuzzy logic. In *Proceedings of IEEE Conference on Systems, Man, and Cybernetics*, pages 2664–2669, 2001.

[10] Y. F. Wang. Predicting stock price using fuzzy gray prediction system. *Expert Systems with Applications*, 22(1):33–38, 2002.

[11] R. Veliev, A. Rubinov, and A. Stranieri. The use of an association rules matrix for economic modelling. In *International conference on neural information processing*, pages 836–841, 1999.

[12] H. White. Economic prediction using neural networks: the case of IBM daily stock returns. In *IEEE Conference on Neural Networks*, pages 451–458, 1988.

[13] E. P. K. Tsang, J. Li, and J. M. Butler. EDDIE beats the bookies. *International Journal of Software, Practice and Experience*, 28(10):1033–1043, 1998.

[14] D. S. Barr and G. Mani. Using neural nets to manage investments. In *AI EXPERT*, pages 16–22, 1994.

[15] R. N. Mantegna and H. E. Stanley. *An introduction to*

Table 1: Continued

Symbols	2001 Hold	2001 RNN	2001 FSGA	2002 Hold	2002 RNN	2002 FSGA	2003 Hold	2003 RNN	2003 FSGA
004270.KS	1.046	0.954	1.141	1.205	1.065	1.229	1.245	1.356	1.262
004550.KS	3.394	4.517	3.968	2.982	1.061	1.466	0.766	0.909	0.734
004700.KS	1.318	1.121	0.843	1.831	1.048	1.822	0.818	0.823	0.818
005270.KS	2.050	1.149	2.399	1.386	1.594	1.422	1.336	1.473	1.294
005380.KS	2.282	2.177	1.274	1.114	1.321	1.298	1.777	1.014	1.492
005490.KS	1.552	1.570	1.329	1.077	0.999	0.989	1.317	1.150	1.023
005720.KS	1.903	1.034	1.003	0.911	1.075	0.774	1.008	1.009	0.919
005810.KS	1.472	1.227	1.222	1.482	1.424	1.190	1.191	1.202	1.077
005930.KS	1.625	1.630	1.909	1.156	0.940	0.999	1.310	0.849	1.060
005940.KS	2.135	1.635	2.478	1.044	1.312	1.313	0.617	0.643	0.657
005950.KS	2.149	1.035	1.069	0.986	1.157	1.059	0.896	0.990	0.958
006090.KS	1.219	1.154	1.453	0.883	1.123	1.023	1.606	1.568	1.372
006120.KS	1.409	1.188	1.208	1.018	1.042	1.212	1.105	1.192	1.285
006260.KS	1.481	0.856	0.758	0.973	0.942	0.948	1.438	1.200	1.265
006340.KS	1.166	1.569	1.100	0.895	0.891	1.024	0.545	0.514	0.538
006350.KS	2.120	1.449	1.936	0.930	0.976	1.081	1.484	1.273	1.605
006360.KS	3.384	1.725	1.010	1.053	1.013	1.053	1.571	1.428	1.434
006400.KS	1.217	1.602	1.162	1.369	1.426	1.617	1.961	1.135	1.907
006570.KS	2.607	1.691	1.625	0.441	1.339	0.441	1.748	1.280	1.763
006600.KS	1.794	1.769	2.098	0.971	0.942	0.961	0.733	0.976	0.901
006800.KS	1.672	1.167	1.575	0.484	0.827	0.562	0.870	0.910	1.001
007160.KS	1.158	0.890	1.075	1.329	1.371	1.602	1.520	1.693	1.723
007690.KS	1.041	1.011	1.062	1.119	0.949	1.305	1.112	0.915	1.127
007810.KS	1.631	1.260	1.077	0.747	0.885	0.871	1.225	1.163	1.455
008670.KS	1.544	1.319	2.294	0.711	0.888	0.829	1.273	1.096	1.442
009150.KS	1.183	1.183	1.394	1.195	0.860	1.415	0.850	0.710	0.893
009320.KS	1.059	1.274	1.221	0.983	1.388	1.139	0.927	1.146	1.102
009830.KS	1.642	1.636	1.666	1.053	1.019	1.222	2.880	1.609	2.520
009970.KS	1.017	0.699	1.136	1.023	0.905	1.010	2.205	2.295	2.259
011050.KS	0.937	1.002	1.087	0.762	0.939	0.901	0.860	1.005	1.057
011280.KS	1.256	1.017	1.506	0.957	0.969	1.056	0.949	0.928	0.990
011780.KS	1.193	1.602	1.392	0.921	1.533	1.068	2.276	1.513	2.456
012450.KS	0.964	0.965	0.847	1.475	1.140	1.752	2.117	1.243	2.226
012510.KS	0.983	0.733	0.966	0.335	1.784	1.258	0.474	0.814	0.560
012800.KS	1.149	0.935	1.176	1.047	0.999	1.209	1.069	1.138	1.164
013890.KS	1.030	1.444	1.429	1.098	1.827	1.284	0.549	0.906	0.639
014130.KS	0.917	0.829	0.733	0.889	0.831	0.947	0.732	0.891	0.856
014350.KS	1.796	1.072	0.882	0.868	0.971	0.868	1.257	0.903	1.172
014900.KS	1.364	1.434	1.332	1.281	0.911	1.487	0.773	0.921	1.031
015760.KS	0.920	0.858	0.857	0.856	1.104	0.935	1.230	1.152	1.111
016360.KS	1.692	2.022	1.587	0.676	0.685	0.705	0.845	1.030	0.972
016380.KS	1.473	1.278	1.726	1.146	1.376	1.373	2.578	2.027	1.834
016570.KS	0.881	0.953	1.043	0.859	0.905	0.995	1.338	1.042	1.134
016610.KS	1.926	1.821	2.113	0.675	1.187	0.801	0.961	0.977	1.088
017670.KS	0.963	0.926	0.963	0.890	0.953	0.938	0.875	1.077	0.970

econophysics: Correlation and Complexity in Finance. Cambridge Univ. Press, Cambridge, 2000.

[16] J. P. Bouchaud and M. Potters. *Theory of Financial Risks: From Statistical Physics to Risk Management*. Cambridge Univ. Press, Cambridge, 2000.

[17] L. Laloux, P. Cizeau, J. P. Bouchaud, and M. Potters. Noise dressing of financial correlation matrices. *Pattern Recognition Letters*, 83:1467–1470, 1999.

[18] V. Plerou, P. Gopikrishnan, B. Rosenow, A. N. Amaral, and H. E. Stanley. Universal and nonuniversal properties of cross correlations in financial time series. *Pattern Recognition Letters*, 83:1471–1474, 1999.

[19] Y. K. Kwon and B. R. Moon. Daily stock prediction using neuro-genetic hybrids. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 2203–2214, 2003.

[20] Y. K. Kwon and B. R. Moon. Evolutionary ensemble for stock prediction. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1102–1113, 2004.

[21] P. J. Kaufman. *Trading Systems and Methods*. John Wiley & Sons, 1998.

[22] H. J. Kim, Y. K. Lee, B. N. Kahng, and I. M. Kim. Weighted scale-free network in financial correlation. *Journal of the Physical Society of Japan*, 71(9):2133–2136, 2002.

[23] J. L. Elman. Finding structure in time. *Cognitive Science*, 14:179–211, 1990.

[24] E. Cantu-Paz. A survey of parallel genetic algorithms. *Calculateurs Parallels*, 10(2):141–171, 1998.

[25] W. Siedlecki and J. Sklansky. A note on genetic algorithms for large-scale feature selection. *Pattern Recognition Letters*, 10:335–347, 1989.