Forecasting Stock Returns Using Genetic Programming in C++

M.A. Kaboudan
Management Science & Information Systems, Penn State Lehigh Valley
Fogelsville, PA 18051, USA
Email: mak7@psu.edu

Abstract

This is an investigation of forecasting stock returns using genetic programming. We first test the hypothesis that genetic programming is equally successful in predicting series produced by data generating processes of different structural complexity. After rejecting the hypothesis, we measure the complexity of thirty-two time series representing four different frequencies of eight stock returns. Then using symbolic regression, it is shown that less complex high frequency data are more predictable than more complex low frequency returns. Although no forecasts are generated here, this investigation provides new insights potentially useful in predicting stock prices.

1. Introduction

Genetic programming (or GP) is a search technique useful in finding a symbolic structural model that characterizes the dynamical behavior of sequential data sets. This technique may be employed to predict stock returns. Returns = ln (P_t / P_{t-1}), where P_t is a stock's closing price at the end of the current period and P_{t-1} is its closing price at the end of the previous period. Although GP seems logical and may in fact – under proper circumstances – yield a best-fit equation to describe the dynamical process generating a time series, its success in predicting series with different complexities has received little attention. The motivation for our research is the unique behavior and nature of stock prices. Given their high volatility, it is essential to investigate the effect of relative complexity on relative predictability of such series. In this paper, an attempt is made to establish such interdependence first using artificially generated data with known but different complexities. If such interdependence exists, then successful prediction of stock returns is dependent on their complexity. Accordingly, before using GP to model the dynamics of stock returns their complexity is measured. It is conceivable that complexity of stock returns is dependent on their frequency. The term "frequency" is used here to mean the number of times the percentage price change (or return) is calculated within a given time period.

Returns measured at four frequencies of eight Dow Jones stocks are included in this study. Each stock is represented by four data sets: two of relatively low frequencies, and two of relatively high frequencies. Traditionally, financial market analysts investigate time-stamped returns. The most commonly investigated frequency is daily data. Few investigate intradaily data such as hourly. The two low frequencies investigated in this study are returns time-stamped every half-hour and every ten minutes. High frequency returns are calculated using minute-to-minute prices and at every price change. For a typical Dow Jones stock that trades thousands of times a day and with its price changing more than once a minute, higher frequency returns may be more predictable and therefore easier to model. Low frequency returns miss too many price movements and may be difficult to model. Chen and Yeh (1997) use a time-variant and non-parametric approach to estimate volatility. The method estimates volatility by taking structural changes into account. Structural changes include external factors not related to the normal dynamics of price movements. External factors are typically breaking news concerning individual companies such as earnings reports or general economic news affecting future interest rates that affect stock prices. Their (Chen and Yeh's) work is appropriate and fits investigating daily stock returns when structural changes are apt to occur. Also appropriate and fitting to investigate daily returns is a study by Fernandez and Evett (1997) who evaluate the effects of external influencing factors on profitability. However, structural changes may not occur in very short periods. For example, there may be no structural changes for an individual stock when observing trades of less than a two-day period. Thus, this investigation provides an alternative method to analyze time-series returns for time-periods too short for external influencing factors to affect structure. If external factors do change, new models must be developed to capture their effects.

This is not the first study that investigates the use of GP in predicting time-series with differing levels of complexity. Prior studies include Fogel and Fogel (1996), Hiden et al. (1997), Jonsson and Barklund (1996), Mulloy and Savit (1996), Oakley (1992), and Oakley (1996). This study is similar to those in attempting to model nonlinear chaotic and noisy data. This paper differs insofar as it endeavors to establish a link between a series' measurable complexity and the ability of GP to model its dynamics. The degree to which this linkage can be established indicates the degree to which GP can be used to predict stock market data.

The GP package used in this study is Andy Singleton's GPQuick (1995), written in C++. Preliminary investigations found that GPQuick produces more reliable results than other GP systems. The GPQuick code to perform symbolic regressions was modified to accept time-series as input files and produce output files containing an equation describing the data structure.

The plan of this investigation is as follows: First, a relationship between data generating processes' levels of complexity and the ability of symbolic regressions to find their possible dynamical structures is established. This is accomplished by evaluating the performance of symbolic regressions in identifying the dynamical structures of artificially generated data sets with known characteristics and complexities. Complexity is quantified using a method developed by Kaboudan (1998). The results indicate that complexity and predictability using GP are inversely related. Following the same logic, complexity and predictability of stock data are analyzed. A brief conclusion based on the small sample of stock returns investigated is made.

2. Linking Complexity with Predictability

In forecasting time-series, we assess the hypothesis that a lower complexity data generating process enjoys correspondingly greater predictability of its process dynamics. To test this hypothesis, eight sets of artificial data from structures with known characteristics are generated. These include time-series data generated from linear, linear-stochastic, nonlinear, nonlinear-chaotic, nonlinear-stochastic, and random processes. Time-series data are a sequence of observed values Y_t that are a function of previous values of the same variable, or

$$Y_t = f(Y_{t-1}, Y_{t-2}, \ldots, Y_{t-n}, \varepsilon_t), \qquad (1)$$

where t = 1, 2,T time periods, n is an integer < T, and ε is noise. (In this study T = 100 and n = 12.) The eight data generating processes investigated in this study are:

1. A simple linear model: The Ozaki equation (Tong, 1990, p. 76) – OZ:

$$Y_t = 1.8708 \quad Y_{t-1} - Y_{t-2}, \qquad (2)$$

2. A nonlinear chaotic function often cited in chaos theory: The logistic map (Grassberger and Procaccia, 1983) - LG:

$$Y_t = 4 \quad Y_{t-1}(1 - Y_{t-1}), \qquad (3)$$

3. A nonlinear chaotic function also widely studied in chaos theory: The Henon map (Grassberger and Procaccia, 1983) - HN:

$$Y_t = 0.3 \quad Y_{t-2} + 1 - 1.4 \quad Y_{t-1}^2, \qquad (4)$$

4. A simple nonlinear trigonometric function - TF:

$$Y_t = 3.9 \quad \sin \quad Y_{t-1} + 0.85 \quad \cos \quad Y_{t-2}, \qquad (5)$$

5. A difference equation with complex roots: Exponentially weighted coefficients function (Tong, 1990, p. 71) - EF:

$$Y_t = (1.43 - 4.5 \quad e^{-Y_{t-1}^2})Y_{t-2}, \qquad (6)$$

6. A second order autoregressive model: AR2 model - AR:

$$Y_t = 0.6 \quad Y_{t-1} + 0.15 \quad Y_{t-2} + \varepsilon_t, \qquad (7)$$

7. A generalized autoregressive model with conditional heteroscedasticity: GARCH(1,1): (Hsieh, 1989) - GR:

$$Y_t = \varepsilon_t \sqrt{h_t}, \qquad (8)$$
$$h_t = 1 + 0.25 \quad Y_{t-1}^2 + 0.7 \quad h_{t-1}$$

8. A pseudo-random data set with Gaussian characteristics – GS. This was generated using the statistical software package RATS.

Symbolic regression is technique for identifying a formula that accurately describes the dynamics of a time-series. Such an equation is best insofar as it maximizes a given fitness function. The program is given a set of sequential data, the dependent variable, to model and predict. It is also given possible explanatory variables (terminals) along with a set of operators (arithmetic functions). For time series, the explanatory variables are histories of the dependent variables as shown in equation (1) above. The operators included in the selected program are addition, subtraction, multiplication, division, logarithmic, trigonometric, exponential, and square root functions. Table 1 illustrates the GPQuick parameters used to effect our symbolic regression runs. The technique used here is very similar to Koza (1992) who provides several symbolic regression examples.

Before using any forecasting technique to search for the underlying data generating process or DGP, it is logical to measure the complexity of that DGP first. Kaboudan (1998) measured complexity using a two-step procedure. A series Y is filtered from linearity first using an autoregressive model with lag determined according to the AIC criterion. The proportion of variation in the data resulting from a linear process, if any, is measured by the R^2 statistic from the filtering process. Complexity of the linear-free or filtered data is then measured by θ. The statistic is a ratio of the correlation dimension measure of the nonlinear series (after filtering) to the dimension after that series is randomly shuffled. Its idea is based on the notion that shuffling ordered data from a deterministic DGP dismembers its structure and increases its dimensionality. A measure value close to zero indicates low complexity, while that approaching one indicates high complexity.

Table 1
Specifications for GPQuick Configuration Files

Generations	100,000
Populations	1,000
Error	0.00001
Sample	100
Terminals	12
Max. expression	50
Init. Expression	6
Mutation rate	100
Cross self	1
Unrestrict. Wt.	70
Cross. Wt.	100
Mut. Wt.	30
Mute node Wt.	100
Mute const. Wt.	100
Mute shrink st.	100
Copy Wt.	10
Select. Method	4
Tourn. Size	7
Mate radius	500
Kill tourn.	2
Max. age	2,000

Table 2 contains results on the data from the eight known structures. The information in the first row identifies each function. The complexity metrics follow. The rest of the Table contains symbolic regression fitness measure results. Fitness is measured using R^2 and sum of squared error or SSE. To obtain the best symbolic regression for each set of data, GPQuick was run 100 times. Results of the fittest equation and the average of the top 25% are reported for each tested function. The results in the Table clearly show that an inverse relationship exists between complexity and predictability. The simple linear or nonlinear functions (the first five) have low complexity and are very predictable. The linear-stochastic function is not as simple even though its linear filtering indicates simplicity. The residuals after filtering are pure noise. Once noise tarnishes an existing signal, complexity increases and predictability becomes more difficult. The level of noise in the nonlinear-stochastic GARCH data is so high; the data is almost as unpredictable as the Gaussian random. The Gaussian data is most complex of all and is least predictable.

3. Evaluating Predictability of Stocks Returns

This Section contains application of the methodology in the previous Section to stock returns. Stock data for six months on CD-ROM were obtained from the TAQ Database produced by the New York Stock Exchange, Inc. The six months are October 1996 through March 1997. The eight Dow Jones stocks selected and represented by four frequencies each are Boeing (BA), General Electric (GE), IBM, Sears (S), AT&T (T), Wall Mart (WMT), and Exxon (XON). Table 3 summarizes the results. They are consistent with some anomalies. Although the relationship between complexity and predictability seems fairly consistent, the highest R^2 statistics are not consistent with the complexity ones. For example, IBM is least complex while WMT is most predictable for 30-minute returns. For 10-minute data, Sears returns are least complex but least predictable. Wall Mart returns are most predictable even though they appear fairly complex. Such inconsistencies disappear when observing the averages or mean R^2 statistics. Since GP is a random search mechanism, it is only natural to find the highest R^2 statistics inconsistent while the means consistent. The averages are more important here. They show that GP was superior in predicting PCRs relative to any of the time-stamped returns including one-minute data. Given that this investigation involves only a small sample of stocks, there is need to investigate a larger sample to obtain statistically irrefutable conclusions. Yet these results suggest that it may be possible to actually invest profitably in the stock market based on predictions using GP for stocks that do not trade often. Stocks investigated in this study normally trade often and price changes are frequent.

Table2
Complexity Versus Predictability of Functions with Known Complexity

Function	OZ	LG	HN	TF	EF	AR	GR	GS
Complexity:								
Linear Filter R^2	0.87	0.00	0.27	0.00	0.00	0.72	0.00	0.00
θ	0.31	0.39	0.54	0.50	0.51	0.99	0.92	1.01
Predictability:								
Highest R^2	1.000	0.9951	0.9917	0.9998	0.9728	0.8211	0.3826	0.3191
Mean R^2	0.9977	0.9951	0.9569	0.9864	0.9053	0.7561	0.3235	0.2559
SSE	0.0000	0.0103	0.4055	0.1693	15.404	60.201	1,064.75	61.600

The average number of price changes in a given hour is about 200, or a price-change every three seconds.

Analysis of the complexity metrics confirms GP's forecasting ability. Clearly the complexity index θ for PCRs is much lower on the average than all the rest with one exception, WMT. However, the linear filtering R^2 for that stock was fairly high. This tells us that these returns are the result of an almost linear process. The R^2 statistic from GP prediction is a confirmation of such simplicity or predictability. Further, WMT returns taken every minute display low complexity. Their complexity is only 0.19 which means that the data is the result of a nonlinear process. Since the linear filter R^2 = 0.30, then the generating process is a combination of linear-nonlinear processes. This may explain the low R^2 from GP prediction. All other results are consistent with logical expectations about the relationship between complexity and predictability.

Table 3
Complexity Versus Predictability of Stock Returns

Function	BA	GE	GM	IBM	S	T	WMT	XON
30-Minute:								
Complexity:								
Linear Filter R^2	0.07	0.03	0.49	0.46	0.01	0.08	0.07	0.47
θ	0.92	0.63	0.80	0.36	0.93	0.67	0.88	0.81
Predictability:								
Highest R^2	0.53	0.34	0.32	0.38	0.43	0.35	0.60	0.43
Mean R^2	0.31	0.27	0.26	0.26	0.26	0.29	0.39	0.28
SSE	7.58	10.42	9.42	10.41	25.11	7.22	27.82	5.77
10-Minute:								
Complexity:								
Linear Filter R^2	0.04	0.05	0.04	0.02	0.02	0.07	0.06	0.01
θ	0.89	0.91	0.87	0.96	0.63	0.90	0.89	0.89
Predictability:								
Highest R^2	0.43	0.34	0.52	0.33	0.28	0.34	0.73	0.53
Mean R^2	0.36	0.28	0.42	0.24	0.20	0.21	0.60	0.42
SSE	1.74	1.92	5.22	6.30	4.60	5.66	12.91	2.17
1-Minute:								
Complexity:								
Linear Filter R^2	0.12	0.10	0.13	0.21	0.11	0.45	0.30	0.11
θ	0.90	1.48	0.31	0.91	0.31	0.68	0.19	0.58
Predictability:								
Highest R^2	0.53	0.42	0.26	0.34	0.25	0.48	0.44	0.38
Mean R^2	0.44	0.33	0.22	0.26	0.22	0.35	0.28	0.25
SSE	5.86	0.47	2.59	0.73	2.33	2.25	5.25	0.66
PCRs:								
Complexity:								
Linear Filter R^2	0.32	0.59	0.54	0.28	0.42	0.59	0.83	0.56
θ	0.27	0.18	0.32	0.44	0.24	0.35	0.80	0.41
Predictability:								
Highest R^2	0.70	0.79	0.72	0.54	0.78	0.62	0.96	0.71
Mean R^2	0.62	0.75	0.69	0.46	0.76	0.54	0.92	0.64
SSE	0.68	0.38	1.89	0.60	1.16	6.68	1.09	0.64

4. Conclusion

This paper presented analysis of stock returns' complexity that led to determining their predictability. In today's fast pace market trading, daily price changes are too many for any model to trace back the source of change from one day to the next. This study suggests that price-change returns are most predictable when using genetic programming. GP fails to handle data points that miss too many observations critical to determining the real DGP. GP seems to perhaps outperform all other available forecasting techniques. Benefits from using GP are enhanced by selecting the appropriate frequency to analyze. These results invite much needed analysis to determine the optimum forecasting conditions for data sensitive to the nature in which data is gathered.

References

Chen, S., and Yeh, C. 1997. Using genetic Programming to Model Volatility in Financial Time Series. In Genetic Programming: Proceedings of the Second Annual Conference, 58-63. Cambridge, MA: The MIT Press.

Fernandez, T., and Evett, M. 1997. Training Period Size and Evolved Trading Systems. In Genetic Programming: Proceedings of the Second Annual Conference, 95. San Francisco, CA: Morgan Kaufmann.

Fogel, D., and Fogel, L. 1996. Preliminary Experiments on Discriminating between Chaotic Signals and Noise Using Evolutionary Programming. In Genetic Programming: Proceedings of the First Annual Conference, 512-520. Cambridge, MA: The MIT Press.

Grassberger, P., and Procaccia, I. 1983. Measuring the Strangeness of Strange Attractors. *Physica* D 9: 189-208.

Hiden, H.; Willis, M.; McKay, B.; and Montague, G. 1997. Non-linear and Direction Dependent Dynamic Modelling Using Genetic Programming. In Genetic Programming: Proceedings of the Second Annual Conference, 168-173. San Francisco, CA: Morgan Kaufmann.

Hsieh, D. 1989. Testing for Nonlinear Dependence in Daily Foreign Exchange Rates. *Journal of Business* 62: 339-368.

Jonsson, P., and Barklund, J. 1996. Characterizing Signal Behaviour Using Genetic Programming. In Evolutionary Computing, Lecture Notes in Computer Science 1143, 62-73. AISB Workshop, Berlin: Springer.

Kaboudan, M. 1998. Statistical Properties of Time-Series-Complexity Measure Applied to Stock Returns. Forthcoming in *Computational Economics*.

Koza, J. 1992. *Genetic Programming*, Cambridge, Massachusetts, The MIT Press.

Mulloy, B.; Riolo, R.; and Savit, R. 1996. Dynamics of Genetic Programming and Chaotic Time Series Prediction, In Genetic Programming: Proceedings of the First Annual Conference, 166-174. Cambridge, MA: The MIT Press.

Oakley, H. (1996). Genetic Programming, the Reflection of Chaos, and the Bootstrap: Toward a Useful Test for Chaos. In Genetic Programming: Proceedings of the First Annual Conference, 175-181. Cambridge, MA: The MIT Press.

Oakley, H 1992. Two Scientific Applications of Genetic Programming: Stack Filters and Non-Linear Equation Fitting to Chaotic Data. In K. Kinnear, Jr. *Advances in Genetic Programming*, 367-389. Cambridge, MA, The MIT Press.

Tong, H. 1990. *Non-linear Time Series: A Dynamical System Approach.* Oxford, Oxford University Press.