Predicting Financial Time Series by Genetic Programming with Trigonometric Functions and High-Order Statistics

Roy Schwaerzel
Department of Computer Science
The University of Texas at San Antonio
San Antonio, TX 78249-0667
1-210-212-5600
rschwaer@cs.utsa.edu

Tom Bylander
Department of Computer Science
The University of Texas at San Antonio
San Antonio, TX 78249-0667
1-210-458-5693
bylander@cs.utsa.edu

Abstract
This paper describes an extension of the traditional application of Genetic Programming in the domain of the prediction of daily currency exchange rates. In combination with trigonometric operators, we introduce a new set of high-order statistical functions in a unique representation and analyze their performance using daily returns of the British Pound and Japanese Yen. In addition, the same experimental design and analysis is applied to ten other financial time series from two different domains. We will demonstrate that the introduction of high-order statistical functions in combination with trigonometric functions will outperform other traditional models such as ARMA models and Genetic Programming with the basic function set. We utilize the Akaike Information Criterion for the selection of the best ARMA model for our benchmark testing. Performance will be measured on hit percentage, average percentage change, and profit. The t-test is applied to derive confidence intervals and to evaluate the significance of our results.

Program Track
Evolutionary Computing, Time Series Prediction.

Keywords
Genetic Programming, Financial Time Series, Prediction and Forecasting, Time Series Analysis, Trigonometric Function Set, High-Order Statistical Function Set.

1. Introduction
Genetic Programming (GP) is an optimization technique that has shown to outperform other traditional techniques in regression [1], classification [2], and optimization [3]. GP has proven to be a valid approach to find suitable models to describe financial time series [4, 5]. Although GP has shown to be amenable for a wide variety of optimization problems, the impact of using high-order statistical functions as building blocks for time series analysis has not been effectively explored. Most current GP implementations [6, 7] are restricted to a basic set of operators (e.g., addition, subtraction, multiplication, protected division[1], etc.) to model the data. For time series analysis, these operators can be enhanced and extended by incorporating a class of statistical measures, defined as the High-Order Statistical Function Set (HSFS). The HSFS includes building blocks that characterize the statistical behavior and nature of financial time series and will include statistical moments (i.e., mean, standard deviation, skewness, and kurtosis). In addition, because financial time series can be periodic, trigonometric functions can characterize this aspect. This research will analyze the performance improvement of the HSFS by itself and in conjunction with trigonometric functions. Our evaluation shows that the function sets, separate and combined, outperform ARMA models and Basic Genetic Programming.

2. Methodology
2.1 GP Background
Genetic Programming can be seen as an extension of the Genetic Algorithm approach replacing the fixed length bit-strings with a more dynamic representation structure of LISP-like symbolic expressions. In GP, the individual solutions of a problem are represented as program trees composed of nodes and arcs. Nodes without outgoing arcs (leaf nodes) are terminals. Nodes with one or more outgoing arcs are functions that act on terminals or other functions. The terminal and function sets are important components of a genetic program since they make up the structure (alphabet) of the genetic program.

The terminal set for our time series modeling consist of a number of variables and constants, where the variables are previous data points of the time series and the constants are real numbers. The function set consists of various mathematical functions, such as addition, subtraction, multiplication, protected division, and potentially other more complex functions. The size, shape, and structure of the solution as a genetic program are left unspecified and are found by using the genetic programming operators (i.e., crossover and mutation). Solving a problem therefore becomes a search through all possible combinations of symbolic expressions defined by the programmer.

2.2 GP Representation
The structure of a genetic program consisting of a set of symbolic expressions is best shown as a parse tree. For example, the symbolic expression (+ (* 6 3) (- 4 (* 2 5))) can be represented by the parse tree in Figure 2.1. The functional alphabet of the genetic program is {+, -, *} and the terminal alphabet is {2, 3, 4, 5, 6}.

---

[1] Protected division is a modified function where the division by zero is defined. The division operator can be rewritten as protected division: ***int division(int x, int y) {if (y==0) return 0; else return (x/y);}***

Though this alphabet is correct for the genetic program shown, this is only a small subset of the available terminal symbols. The selection of an appropriate functional and terminal alphabet is crucial for the convergence of the GP.

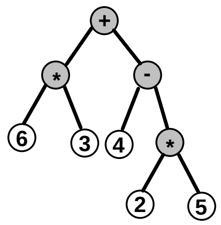


Figure 2.1. Simple Parse Tree representing a symbolic expression of (+ (* 6 3) (- 4 (* 2 5))). The filled nodes represent the functional alphabet and the unfilled nodes represent the terminal alphabet.

The initial set of functions generally used in GP models is the Basic Function Set (BFS). As proposed by Koza [8], the BFS contains elementary mathematical functions such as addition (+), subtraction (-), multiplication (*), and protected division (/). When applied to time series predictions, the BFS should be able to implement regression functions such as those as derived from traditional statistical models.

The BFS may be suitable for simple time series predictions; however, complex systems such as financial time series are often characterized by periodicity or seasonality. These time series might be modeled much better by a series of waves, such as those generated by trigonometric functions. Although they can be constructed from the BFS, basic trigonometric functions should be added to the function set because using the BFS operations of +, -, *, or / to discover them may be difficult and computationally expensive for a genetic program.

We will also expand the original function set used in GP, the Basic and Extended Function Sets, with a set of operations yielding higher order statistical measures (e.g., mean, standard deviation, skewness, and kurtosis). We call this set of operators the High-Order Statistical Function Set (HSFS). The introduction of the HSFS will add aggregated information about the trend and distribution of time series.

2.3 Statistical Features of Time Series

A time series is a sequence of data values $x_1, x_2, ..., x_n$ that are separated by equal time intervals. Here, x_t is the notation for the value of the time series at time t. The analysis of time series is primarily the art of estimating the most likely stochastic process that could have generated an observed time series. Although low order statistical features (e.g. mean and standard deviation) have been widely used for summarizing data, high order statistical features (e.g., skewness and kurtosis) are not often used outside the statistical community, especially for time series modeling.

2.3.1 Mean

The mean is a measure for describing the central tendency for a set of observations. The sample mean $\overline{X}$ is the sum of the individual observations x_i divided by the number of observations n,

$$\overline{x} = \frac{1}{n}\sum_{t=1}^{n} x_t \ .$$

A further distinction needs to be made between the mean of a set of observations and the mean of the population. The sample mean $\overline{X}$ is an estimate of the population mean μ, $\overline{X}$ becomes a better estimate of μ if the number of observations n is large.

2.3.2 Standard Deviation

The standard deviation is a measure that describes the dispersion or variability of a population, the tendency for observations to depart from the central value. The variance s^2 of a set of n observations $x_1, x_2, ..., x_n$ is calculated as

$$s^2 = \frac{1}{n-1}\sum_{t=1}^{n}(x_t - \overline{x})^2 \ ,$$

where $\overline{X}$ is the sample mean as previously defined. The index s^2 is called the sample variance and its positive square root s is called the sample standard deviation. A distinction must be drawn between the population standard deviation σ and the sample standard deviation s. As the number of samples approaches the size of the population, the sample standard deviation s becomes a better estimate of σ.

2.3.3 Skewness

Given that the distribution of a given time series will probably not be completely symmetrical, the degree of asymmetry, referred to as skewness, can be calculated as

$$b = \frac{1}{n-1}\sum_{t=1}^{n}\left(\frac{(x_t - \overline{x})^3}{s^3}\right).$$

A distribution has a positive skewness if the long tail is on the side of the high values of x. Similarly, the skewness is negative if the long tail is on the side of the low values of x. A symmetric distribution has a skewness of zero.

2.3.4 Kurtosis

The degree of "peakedness" in a distribution is expressed by the kurtosis which is defined as

$$k = \frac{1}{n-1}\sum_{t=1}^{n}\left(\frac{(x_t - \overline{x})^4}{s^4}\right).$$

The kurtosis is equal to 3 for a Normal distribution. If the kurtosis is greater than 3, the distribution is more sharply peaked than the Normal distribution (i.e. if it has long "tails"). If the kurtosis is less than 3, the distribution is flatter than the Normal distribution, with shorter tails.

2.4 GP Implementation

A number of genetic programming tools have been made available for the researcher community. We selected the GPC++ - Genetic Programming C++ Class Library [10], which provided us with a basic tool to implement our extended function set. We modified the code to allow the selection of the previous 20 data points to be part of the new terminal set, added the Extended Function Set (sin, cos, tan, log, and exp), and our High-Order Statistical Function Set (mean, standard deviation, skewness, and kurtosis).

Although the implementation of the trigonometric functions were straight forward, only allowing for one operand per function, the implementation of the HSFS required information about the data points these statistics should be calculated on. Therefore, we defined two parameters, the **LAG** and the **LENGTH**. The LAG parameter is a value between 1 and 20 specifying how many time steps back from the prediction this statistics should be calculated on. For example, LAG=1 would include the previous data point, LAG=5 would include the data point one week prior the prediction. The LENGTH parameter specifies the number of data points to include for the statistics, starting from the LAG value backwards. Thus, Mean(1,10) would calculate the average of the last 10 trading days.

3. Experimental Setup

3.1 Data Sets

The daily rates and returns of two major currencies - the British Pound and the Japanese Yen – against the US Dollar for the period from January 1, 1990 to September 16, 2005 were selected for our experiments and analysis. The source of the data was the Federal Reserve Bank of New York. The above time interval contains 4,100 weekdays with 3,953 actual trading days. Both time series were divided into the training and the testing data sets as specified in Table 3.1. In the case that a weekday was not a trading day, the closing rate of the previous trading day is used as the closing rate for that particular day.

Table 3.1. Training and Testing Data Sets Summary for the British Pound (GBP) and the Japanese Yen (JPY) Exchange Rates

	Training Data Set		Testing Data Set	
	GBP	JPY	GBP	JPY
Time period	1/1/1990 - 3/29/1995		3/30/1995 - 9/16/2005	
Weekdays	1,368 days		2,732 days	
Trading days	1,318 days		2,635 days	
No Change days	15 days	23 days	25 days	14 days

This analysis focuses on daily percentage returns rather than on daily rates, because the daily rates are nonstationary and will be pre-processed as discussed in next section "Data Transformation". Figures 3.1 to 3.4 show the plots for the daily currency exchange rates and returns of both analyzed currencies. The statistical properties for the daily exchange rates are summarized in Table 3.2, the statistical properties for the daily returns can be found in Table 3.3.

Table 3.2. Statistics of daily rates for selected currencies

	British Pound	Japanese Yen
Trading days	3,953	
Lowest rate	0.499127	81.120
Highest rate	0.728332	159.900
Average	0.616905	117.296

Table 3.3. Statistics of daily returns for selected currencies

	British Pound	Japanese Yen
Arithmetic Mean	-0.000011697	-0.000043239
Variance	0.000031738	0.000046351
Standard Deviation	0.005633678	0.006808183
Skewness	0.288380279	-0.462796449
Kurtosis	2.511965074	4.100380570
Minimum	-0.028478964	-0.054746474
Maximum	0.033406957	0.032929599
Range	0.061885922	0.087676072

3.2 Data Transformation

The prediction of time series is based on the assumption that the behavior of that time series is stationary. It does not mean that the time series itself must remain fairly stable, but that some function of the time series has to remain stable. A process that generates data with the same mean, variance, and correlation at time lag ***k*** over time is known as a stationary process.

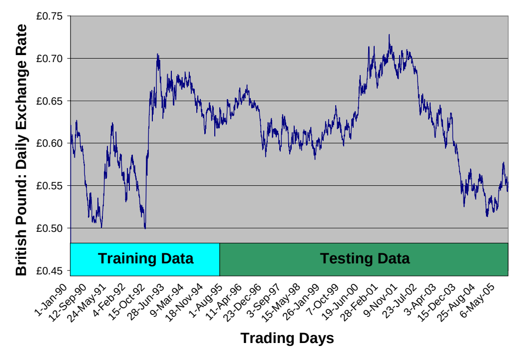


Figure 3.1. Daily Exchange Rates of British Pound per US Dollar (non-stationary time series)

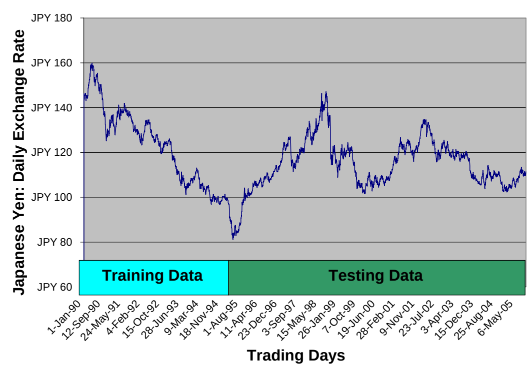


Figure 3.2. Daily Exchange Rates of Japanese Yen per US Dollar (non-stationary time series)

Figures 3.1.and 3.2 are good examples for nonstationary time series, as they demonstrate that the mean of smaller periods are not constant over time. In fact, most financial time series are characterized as nonstationary. The statistical models discussed are restricted to stationary time series. Fortunately, many nonstationary time series can be transformed into stationary ones. Thus, even statistical models used to analyze stationary time series can be used to analyze nonstationary data. The most common transformation on nonstationary data to stationary data is the differencing method, as used by the ARIMA model. Differencing is a simple operation that involves calculating successive changes in the values of a data series. The following example, adapted from Pankratz [11], describes the differencing process in more detail. To difference a data series, a new variable y_t is defined, which is the change in the observed time series data x_t:

$$y_t = x_t - x_{t-1}, \quad t = 2,3,\ldots,n$$

The new series y_t is called the first differences of x_t. In order to incorporate the magnitude of the change, we transformed the first differences to a new transformed time series, the daily percentage returns, or short daily returns, using the following simple transformation:

$$y_t = \frac{x_t - x_{t-1}}{x_{t-1}}, \quad t = 2,3,\ldots,n$$

Figures 3.3 and 3.4 show the plots of the transformed time series from Figures 3.1 and 3.2 after above transformation (daily percentage return) has been applied. Now, both transformed time series exhibit a stationary process that can be used for the statistical modeling and for the genetic program training.

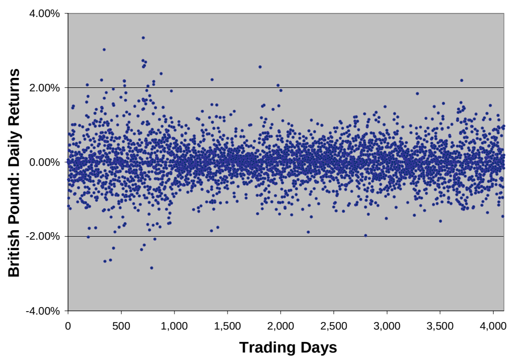


Figure 3.3. Daily Returns of British Pound per US Dollar (stationary time series)

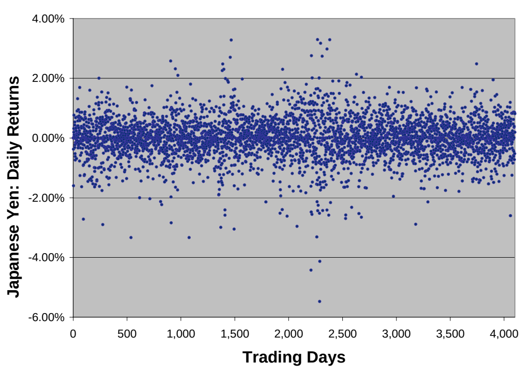


Figure 3.4. Daily Returns of Japanese Yen per US Dollar (stationary time series)

3.3 Performance Measures

The **Mean Square Error (MSE)** is used by the GP system to determine the individuals that best fit the training data of the time series. The MSE is a well known statistical measure and is defined as follows:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - f(x_i))^2,$$

where y_i is the outcome of the ***i-th*** data point, $f(x_i)$ is the calculated outcome using the ***i-th*** data point, and N is the number of the data points in the data set. Thus, the MSE measures the "misfit" of the regression estimates (calculated predictions of the GP system) and its desired output. It is a simple performance metric. The MSE is an artificial measure of performance, because the mean square error is just a quantitative expression of the current system error and it does not measure the profitability of the system.

The **Hit Percentage (HIT)** shows how accurately the trend directions have been tracked by the model [12]:

$$HIT = \frac{N_{up\_up} + N_{down\_down}}{N},$$

where N_{up_up} is the number of time when the estimated and the desired outcome exhibit both upward raising tendency, and N_{down_down} is the number of times when the estimated and the given outcome exhibit both falling tendency. N is the total number of cases where the original series had a change from its previous data point, excluding all non-trading days and days without any change (see Table 3.1 for data sets summary). The ***HIT*** matrix gives a first estimate of the profitability. However, a HIT rate of more than 50% still doesn't guarantee that the system is generating a profit, because the system can be correct for trades with small returns and incorrect for trades with larger returns. To capture all these effects, a more realistic market metric is necessary.

The **Average Percentage Change (APC)** is a performance measure that is independent from any historic performance and is calculated as the average of all returns

$$APC = \frac{1}{N}\sum_{i=1}^{N}\begin{cases} +\left(\dfrac{|y_i - y_{i-1}|}{y_{i-1}}\right) & \begin{array}{l} if \quad (y_i > 0 \; and \; f(x_i) > 0) \\ \; or \; (y_i < 0 \; and \; f(x_i) < 0) \end{array} \\ -\left(\dfrac{|y_i - y_{i-1}|}{y_{i-1}}\right) & \begin{array}{l} if \quad (y_i > 0 \; and \; f(x_i) < 0) \\ \; or \; (y_i < 0 \; and \; f(x_i) > 0) \end{array} \\ 0 & if \;\; y_i = 0 \;\; or \;\; f(x_i) = 0 \end{cases}$$

where y_i is the outcome of the ***i-th*** data point, y_{i-1} is the outcome of the previous data point, $f(x_i)$ is the calculated outcome, and N is the number of the data points in the data set. In contrast to the Hit Percentage, the APC takes the magnitude of a correct or incorrect prediction into account. For example, an incorrect prediction on a day with a large daily rate change will be penalized more than the incorrect prediction with a small daily rate change.

The ultimate performance metric is a **Profit** analysis (real market simulation) with and without including trading cost. This analysis simulates the real cash flow of investment in the financial market. In the simplest implementation, a foreign currency is bought or held whenever a positive return is predicted, and is sold, when the predicted return is negative. Simulation without including trading cost will be used for the performance comparison of our GP systems and benchmark models.

3.4 Benchmark Models

Besides the comparison of the different Genetic Programming models, we have included the widely used Buy & Hold strategy and the more complex statistical ARMA model. The class of ARMA models is formed by combining the Autoregressive Model (AR) and the Moving Average Model (MA). Autoregressive models and Moving Averages models are special cases of a broader class of time series models known as autoregressive integrated moving average (ARIMA) models. The ARIMA models include models that can be represented as follows:

$$Y_t = \overbrace{\alpha_0}^{Constant} + \underbrace{\alpha_1 Y_{t-1} + \alpha_2 Y_{t-2} + \cdots + \alpha_p Y_{t-p}}_{Autoregressive\ terms} + \overbrace{\varepsilon_t}^{Noise\ term} - \underbrace{\beta_1 \varepsilon_{t-1} - \beta_2 \varepsilon_{t-2} - \cdots - \beta_q \varepsilon_{t-q}}_{Moving\ average\ terms}$$

where $\boldsymbol{\varepsilon_t}$ are independent.

The ARIMA models that we consider are referred to as ***ARIMA(p,d,q)*** models. The notion of ***p*** refers to the number of autoregressive terms in the model. The ARIMA model is able to transform nonstationary time series to stationary time series using a process called differencing. If a time series is not stationary, then changes in the successive time series values or the first differences, $\boldsymbol{Y_t - Y_{t-1}}$, are often stationary. If the first differences are not stationary, then the differences of the differences, or the second differences, of the time series values are often stationary. The notion of ***d*** refers to the degree of differencing specified to obtain a stationary series. The notion of ***q*** refers to the number of lagged error terms that make up the moving average part of the model.

ARIMA models are often estimated by using an iterative nonlinear least square estimation procedure. We will be using "EasyReg International", an academic software package developed by J. Bierens [13], to provide us with these estimates. Since we already transformed the original time series data (daily rates) to a stationary time series (daily returns), we will use the ***ARIMA(p,0,q)*** model, also known as the ***ARMA(p,q)*** model. We used the Akaike Information Criterion (AIC) for the selection of the optimal model. Akaike's Information Criterion is a criterion for selecting among nested econometric models. The ***AIC*** is a number associated with each model:

$$AIC = \ln\left(s_m^2\right) + \frac{2m}{T} \quad ,$$

where ***m*** is the number of parameters in the model, and $\boldsymbol{s_m^2}$ is (in an *AR(m)* example) the estimated residual variance: $\boldsymbol{s_m^2}$ = (sum of squared residuals for model ***m***)/***T***. That is, the average squared residual for model ***m***.

The criterion may be minimized over choices of ***m*** to form a tradeoff between the fit of the model (which lowers the sum of squared residuals) and the model's complexity, which is measured by ***m***. Thus an ***AR(m)*** model versus an ***AR(m+1)*** can be compared by this criterion for a given batch of data.

4. Experiments and Results for two Major Currency Exchange Rates

GP with four different function set selections were used to model two time series. All other parameters (e.g., population size, number of generations) were kept constant for all experiments.

GP configurations:

1. **Basic GP:** Traditional GP approach using the Basic Function Set (addition, subtraction, multiplication, and protected division)
2. **GP+EFS:** Traditional GP approach using the Basic and the Extended Function Set (sin, cos, tan, log, and exp)
3. **GP+HSFS:** Traditional GP approach using the Basic and the High-Order Statistical Function Set (mean, standard deviation, skewness, and kurtosis)
4. **GP+EFS+HSFS:** Traditional GP approach using the Basic, the Extended, and the High-Order Statistical Function Set

Primary Currency Exchange Rates:

1. British Pound (GBP) per US Dollar
2. Japanese Yen (JPY) per US Dollar

GP with four different function set selections were used to model both selected time series**.** All other parameters (e.g., population size, number of generations) were kept constant for all GP models. Thirty independent runs were performed on each of above experiments using a population size of 1,000 individuals over 100 generations. The performance on the training data is measured using MSE, HIT, APC, and Profit.

4.1 Performance on Testing Data

In this section, we compare the performance of the GP models with the lowest MSE to each other and to the ARMA and Buy & Hold models. Tables 4.1 and 4.2 summarize the performance of the GP models, the ARMA model selected by the Akaike criterion, and the simple Buy & Hold Strategy. We want to point out that all GP models demonstrated a significantly better performance using any of the previously defined performance measures.

Table 4.1. Performance Matrix for British Pound

Method	HIT	APC	Profit
GP+EFS+HSFS	**55.25%**	**0.042591%**	**88.78%**
GP+EFS	53.91%	0.037322%	75.61%
GP+HSFS	53.72%	0.024652%	47.99%
Basic GP	52.38%	0.020227%	38.71%
Buy & Hold	50.19%	0.002996%	12.15%
Akaike: AR(1)	48.93%	0.002679%	9.54%

Table 4.2. Performance Matrix for Japanese Yen

Method	HIT	APC	Profit
GP+EFS+HSFS	**53.00%**	**0.051799%**	**79.78%**
GP+EFS	52.88%	0.048613%	72.03%
GP+HSFS	52.84%	0.033564%	40.05%
Basic GP	52.61%	0.027045%	26.78%
Akaike: AR(1)	48.61%	-0.009940%	-19.33%
Buy & Hold	48.65%	-0.011033%	-20.71%

Figure 4.1 displays the profit performance of the GP models vs. the performance of the other benchmark models on the British Pound series. This figure compares the GP models that demonstrated the best MSE performance on the training data; although we note that the average performance of the GP models also clearly outperform the other models. Figure 4.2 displays the profit performance on the Japanese Yen series which exhibits a very similar tendency as the British Pound series. We added a plot of the corresponding currency exchange rate to the lower part of each performance chart to view the model performance in its context to the market behavior.

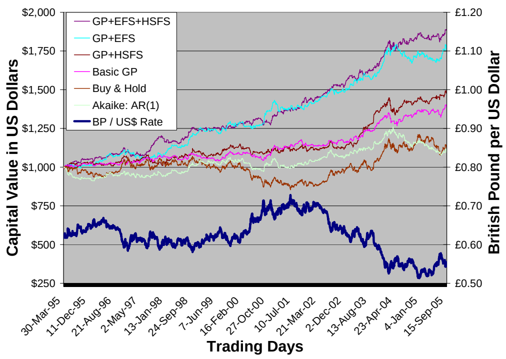


Figure 4.1. Profit performance of models for trading of British Pound; Startup capital of $1,000; the order of models in the key corresponds to the final order on the right; bottom series is a plot of the corresponding British Pound Exchange rate.

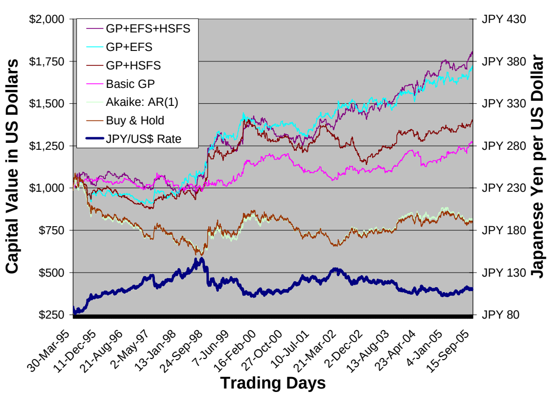


Figure 4.2. Profit performance of models for trading of Japanese Yen; Startup capital of $1,000; the order of models in the key corresponds to the final order on the right; bottom series is a plot of the corresponding Japanese Yen Exchange rate.

4.2 Statistical Results on Testing Data

In this section, we show the statistical results over the 30 runs comparing the GP models to each other and to the Buy & Hold and ARMA models. Tables 4.3 to 4.5 summarize the average performance, including the 95% confidence intervals over the 30 runs, of the various GP configurations and performance metrics.

Table 4.3. Average HIT for British Pound and Japanese Yen including the 95% confidence interval ranges (over 30 runs) on testing data set.

Method	HIT – BP	HIT – JY
GP+EFS+HSFS	53.13% ± 1.27%	52.08% ± 0.56%
GP+EFS	52.64% ± 0.39%	51.73% ± 0.89%
GP+HSFS	51.96% ± 1.18%	51.46% ± 0.51%
Basic GP	50.57% ± 0.31%	51.43% ± 0.75%

Table 4.4. Average APC for British Pound and Japanese Yen including the 95% confidence interval ranges (over 30 runs) on testing data set.

Method	APC – BP	APC – JY
GP+EFS+HSFS	0.035240% ± 0.013894%	0.028637% ± 0.011097%
GP+EFS	0.028039% ± 0.012222%	0.019347% ± 0.014748%
GP+HSFS	0.013006% ± 0.007788%	0.009974% ± 0.010606%
Basic GP	0.005009% ± 0.007248%	0.010717% ± 0.008492%

Table 4.5. Average Profit for British Pound and Japanese Yen including the 95% confidence interval ranges (over 30 runs) on testing data set.

Method	Profit – BP	Profit – JY
GP+EFS+HSFS	73.30% ± 29.15%	44.11% ± 17.09%
GP+EFS	56.44% ± 24.95%	28.77% ± 21.98%
GP+HSFS	25.32% ± 15.16%	11.90% ± 12.66%
Basic GP	12.23% ± 12.82%	10.53% ± 8.34%

We utilized the difference of means t-Test on the HIT averages over those 30 runs to determine if the performance improvement of the various models is significant to a certain level. The results are summarized in Tables 4.6 to 4.11. The analysis of the British Pound HIT performance, summarized in Table 4.6, underlines that the GP+HSFS is significantly better than the Basic GP, that the GP+EFS is significantly better than the GP+HSFS, and that the GP+EFS+HSFS is significantly better than any of the other GP models at a 99.9% and 80% confidence level. The analysis of the Japanese Yen HIT performance shows similar results. Although the HIT performance of the GP+HSFS was not significantly better than the Basic GP, the GP+EFS models showed a significant performance improvement over the Basic GP and the GP+HSFS models at a confidence level of 99.9% and 95%. The GP+EFS+HSFS is the best performing model with 99.9% and 90% confidence intervals as shown in Table 4.7. All GP models showed a significantly better HIT performance than the simple Buy & Hold Strategy and the ARMA model for both currencies.

Table 4.6. Difference of Means Test on HIT Performance for BP

Model	Akaike	Basic GP	GP+HSFS	GP+EFS	GP+EFS+HSFS
Buy & Hold	n/a	**95%**	**99.9%**	**99.9%**	**99.9%**
Akaike		**99.9%**	**99.9%**	**99.9%**	**99.9%**
Basic GP			**99.9%**	**99.9%**	**99.9%**
GP+HSFS				**95%**	**99.9%**
GP+EFS					80%

Table 4.7. Difference of Means Test on HIT Performance for JY

Model	Akaike	Basic GP	GP+HSFS	GP+EFS	GP+EFS+HSFS
Buy & Hold	n/a	**99.9%**	**99.9%**	**99.9%**	**99.9%**
Akaike		**99.9%**	**99.9%**	**99.9%**	**99.9%**
Basic GP			Failed	80%	**99.9%**
GP+HSFS				80%	**99.9%**
GP+EFS					90%

The statistical results for the APC performance of the British Pound and the Japanese Yen, shown in Tables 4.8 to 4.9, are similar to the statistical results for the HIT performance of both currencies. Although all GP models outperform the Buy & Hold and the ARMA models, the performance of GP+HSFS model was not significantly better than the Basic GP. The best performing model based on APC was the GP+HSFS, followed by the GP+EFS and the G+HSFS models for both currencies.

Table 4.8. Difference of Means Test on APC Performance for BP

Model	Akaike	Basic GP	GP+HSFS	GP+EFS	GP+EFS+HSFS
Buy & Hold	n/a	80%	**99.9%**	**99.9%**	**99.9%**
Akaike		90%	**99.9%**	**99.9%**	**99.9%**
Basic GP			**99.9%**	**99.9%**	**99.9%**
GP+HSFS				**99.9%**	**99.9%**
GP+EFS					**95%**

Table 4.9. Difference of Means Test on APC Performance for JY

Model	Akaike	Basic GP	GP+HSFS	GP+EFS	GP+EFS+HSFS
Buy & Hold	n/a	**99.9%**	**99.9%**	**99.9%**	**99.9%**
Akaike		**99.9%**	**99.9%**	**99.9%**	**99.9%**
Basic GP			Failed	**99%**	**99.9%**
GP+HSFS				**99%**	**99.9%**
GP+EFS					**99%**

The statistical results for the Profit performance, summarized in Tables 4.10 and 4.11, are similar to the statistical results of the APC performance. The main difference is that the Basic GP models for the British Pound did not provide significantly better results than the Buy & Hold Strategy and the ARMA model.

Table 4.10. Difference of Means Test on Profit Performance for BP

Model	Akaike	Basic GP	GP+HSFS	GP+EFS	GP+EFS+HSFS
Buy & Hold	n/a	Failed	**99%**	**99.9%**	**99.9%**
Akaike		Failed	**99.9%**	**99.9%**	**99.9%**
Basic GP			**99.9%**	**99.9%**	**99.9%**
GP+HSFS				**99.9%**	**99.9%**
GP+EFS					**98%**

Table 4.11. Difference of Means Test on Profit Performance for JY

Model	Akaike	Basic GP	GP+HSFS	GP+EFS	GP+EFS+HSFS
Buy & Hold	n/a	**99.9%**	**99.9%**	**99.9%**	**99.9%**
Akaike		**99.9%**	**99.9%**	**99.9%**	**99.9%**
Basic GP			Failed	**99.9%**	**99.9%**
GP+HSFS				**99.9%**	**99.9%**
GP+EFS					**99%**

5. Experiments and Results for Additional Financial Time Series

In this section, we show that the results for the prediction of the British Pound and Japanese Yen currency exchange rates can be applied to other major currencies or to other financial time series, in general. We selected the same time periods for the training and testing data sets as for the British Pound and the Japanese Yen currency exchange rates from the previous section. The source for the currency exchange rates was the Federal Reserve Bank of New York. The source for the major stock market indices was Yahoo!, who obtained the data from Commodity Systems, Inc. (CSI). In case that a weekday was not a trading day, the closing rate of the previous trading day was used as the closing rate for that non-trading day.

Additional Financial Time Series:

Major Currency Exchange Rates:

1. Australian Dollar (AUD) per US Dollar
2. Canadian Dollar (CAD) per US Dollar
3. Swiss Frank (CHF) per US Dollar
4. Taiwan Dollar (TWD) per US Dollar
5. Thai Baht (THB) per US Dollar

Major Stock Market Indices:

6. FTSA 100 (British Stock Market Index)
7. NIKKEY 225 (Japanese Stock Market Index)
8. Dow Jones Industrial Average (US Stock Market Index)
9. NASDAQ (US Stock Market Index)
10. S&P 500 (US Stock Market Index)

5.1 Performance on Testing Data

Tables 5.1 to 5.10 summarize the performance of the GP models, the ARMA model (selected by Akaike criterion), and the simple Buy & Hold Strategy. A plot of the profit performance of each analyzed time series can be found in Figures 5.1 to 5.10. In general, the GP models utilizing the EFS and the HSFS were performing better than the Basic GP, the Buy & Hold Strategy, and even the Akaike selected ARMA models. Unlike the experiment in the previous section, only one run was performed for each of the GP models on each of the time series. Again, we used a population size of 1,000 individuals over 100 generations.

Table 5.1. Performance Matrix for Australian Dollar

Method	HIT	APC	Profit
GP+EFS+HSFS	52.16%	**0.033644%**	**62.47%**
GP+EFS	51.74%	0.033030%	60.42%
GP+HSFS	52.43%	0.028535%	51.32%
Basic GP	52.51%	0.028801%	50.70%
Akaike: MA(4)	**52.66%**	0.028682%	49.12%
Buy & Hold	50.96%	-0.000092%	5.90%

Table 5.2. Performance Matrix for Canadian Dollar

Method	HIT	APC	Profit
GP+HSFS	**51.81%**	0.011540%	**27.31%**
Basic GP	51.73%	**0.011612%**	27.17%
GP+EFS+HSFS	50.04%	0.008080%	21.46%
Buy & Hold	49.88%	0.005487%	18.70%
GP+EFS	49.65%	-0.002541%	5.03%
Akaike: MA(1)	50.15%	-0.005645%	0.12%

Table 5.3. Performance Matrix for Swiss Franc

Method	HIT	APC	Profit
GP+EFS+HSFS	53.46%	**0.043130%**	**69.86%**
GP+HSFS	**53.54%**	0.030136%	42.14%
GP+EFS	50.63%	0.025882%	34.04%
Basic GP	52.35%	0.021407%	25.19%
Buy & Hold	48.03%	-0.006167%	-10.21%
Akaike: MA(1)	46.88%	-0.005548%	-11.16%

Table 5.4. Performance Matrix for Taiwan Dollar

Method	HIT	APC	Profit
GP+EFS+HSFS	53.01%	**0.017105%**	**12.12%**
GP+EFS	**53.21%**	0.015889%	10.38%
GP+HSFS	52.14%	0.012058%	4.56%
Basic GP	52.72%	-0.000167%	-10.95%
Buy & Hold	49.95%	-0.009189%	-20.81%
Akaike: AR(1)	41.98%	-0.036482%	-21.94%

Table 5.5. Performance Matrix for Thai Baht

Method	HIT	APC	Profit
GP+HSFS	51.30%	**0.047978%**	**33.78%**
GP+EFS	**51.75%**	0.038585%	29.09%
GP+EFS+HSFS	50.45%	0.038648%	28.04%
Basic GP	50.98%	0.011415%	-10.93%
Buy & Hold	49.63%	-0.021940%	-39.86%
Akaike: MA(1)	50.08%	-0.020219%	-41.11%

Table 5.6. Performance Matrix for FTSE 100

Method	HIT	APC	Profit
GP+EFS+HSFS	51.71%	0.024299%	**80.15%**
GP+EFS	50.95%	0.021832%	73.83%
Buy & Hold	**52.81%**	**0.025828%**	72.10%
GP+HSFS	51.59%	0.017581%	63.49%
Basic GP	50.95%	0.017322%	60.27%
Ak./HQ: MA(7)	50.95%	0.013454%	55.39%

Table 5.7. Performance Matrix for NIKKEI 225

Method	HIT	APC	Profit
GP+EFS	50.82%	**0.041565%**	**53.93%**
GP+EFS+HSFS	50.04%	0.030128%	31.65%
GP+HSFS	50.04%	0.029670%	30.42%
Ak./HQ: AR(2)	**51.01%**	0.007757%	2.21%
Basic GP	49.61%	0.007646%	-6.56%
Buy & Hold	49.77%	0.001160%	-21.28%

Table 5.8. Performance Matrix for DJIA

Method	HIT	APC	Profit
GP+HSFS	51.43%	0.038895%	**171.48%**
GP+EFS	51.35%	0.038251%	165.40%
Buy & Hold	**52.30%**	**0.040263%**	155.77%
GP+EFS+HSFS	52.03%	0.034194%	151.33%
Basic GP	51.50%	0.034984%	150.31%
Akaike: AR(7)	51.65%	0.035188%	149.09%

Table 5.9. Performance Matrix for NASDAQ

Method	HIT	APC	Profit
GP+HSFS	52.90%	0.056618%	**251.81%**
GP+EFS+HSFS	52.67%	**0.058503%**	249.74%
Ak./HQ: MA(1)	52.75%	0.047539%	203.06%
Basic GP	53.09%	0.045296%	181.19%
GP+EFS	52.83%	0.043051%	178.68%
Buy & Hold	**54.11%**	0.051082%	163.73%

Table 5.10. Performance Matrix for S&P 500

Method	HIT	APC	Profit
GP+EFS	52.41%	**0.047345%**	**194.30%**
Ak./HQ: AR(7)	52.26%	0.046348%	183.06%
GP+EFS+HSFS	52.33%	0.037209%	156.98%
GP+HSFS	52.49%	0.035589%	155.37%
Buy & Hold	**52.71%**	0.039139%	146.05%
Basic GP	52.41%	0.029736%	128.26%

5.2 Statistical Results on Testing Data

We utilized the Paired Difference t-Test to determine if the performance improvement of the various models is significant to a certain level. We selected the HIT, the APC, and the Profit performance measures. Table 5.11 shows the results of the paired difference t-test on the HIT performance metric, indicating that no statistically significant performance improvement can be established using this metric. However, HIT by itself is an incomplete performance metric, because it doesn't include the impact of a correct or wrong prediction.

Table 5.11. Paired Difference t-Test on HIT Performance

Model	Akaike	Basic GP	GP+EFS	GP+HSFS	GP+EFS+HSFS
Buy & Hold	Failed	Failed	Failed	**80%**	Failed
Akaike		**80%**	Failed	**80%**	**80%**
Basic GP			Failed	Failed	Failed
GP+EFS				Failed	Failed
GP+HSFS					Failed

The APC is a profit-oriented performance measure that is independent from any historic performance. Table 5.12 summarizes the results of the Paired Difference t-Test over all ten analyzed financial time series based on the APC. The GP models with the added function sets clearly outperform the simple Buy & Hold Strategy, and the Akaike model with confidence levels of 95% to 99%. The APC can only establish an 80% confidence that the Basic GP models are outperforming the Simple Buy & Hold and the Akaike selected ARMA models. Although the average APC performance of the GP+HSFS and the GP+EFS+HSFS is higher than the average profits of the GP+EFS, the t-test cannot establish a significance of those differences.

Table 5.12. Paired Difference t-Test on APC Performance

Model	Akaike	Basic GP	GP+EFS	GP+HSFS	GP+EFS+HSFS
Buy & Hold	Failed	80%	**95%**	**95%**	**95%**
Akaike		80%	**95%**	**95%**	**98%**
Basic GP			90%	**95%**	**99%**
GP+EFS				Failed	Failed
GP+HSFS					Failed

The Profit is the ultimate performance metric, because it simulates the real cash flow in a selected time period. Table 5.13 summarizes the results of the paired difference t-test based on the realized profits. The results for the Profit show a very similar behavior as the results for the APC. Again, the results indicate that the GP models with the added function sets clearly outperform the simple Buy & Hold Strategy, the Akaike model, and the Basic GP models with confidence levels of 95% to 99%. Similarly to the APC, the average Profit of the GP+HSFS and the GP+EFS+HSFS are higher than the average Profit of the GP+EFS, but the results cannot establish a significance of those differences.

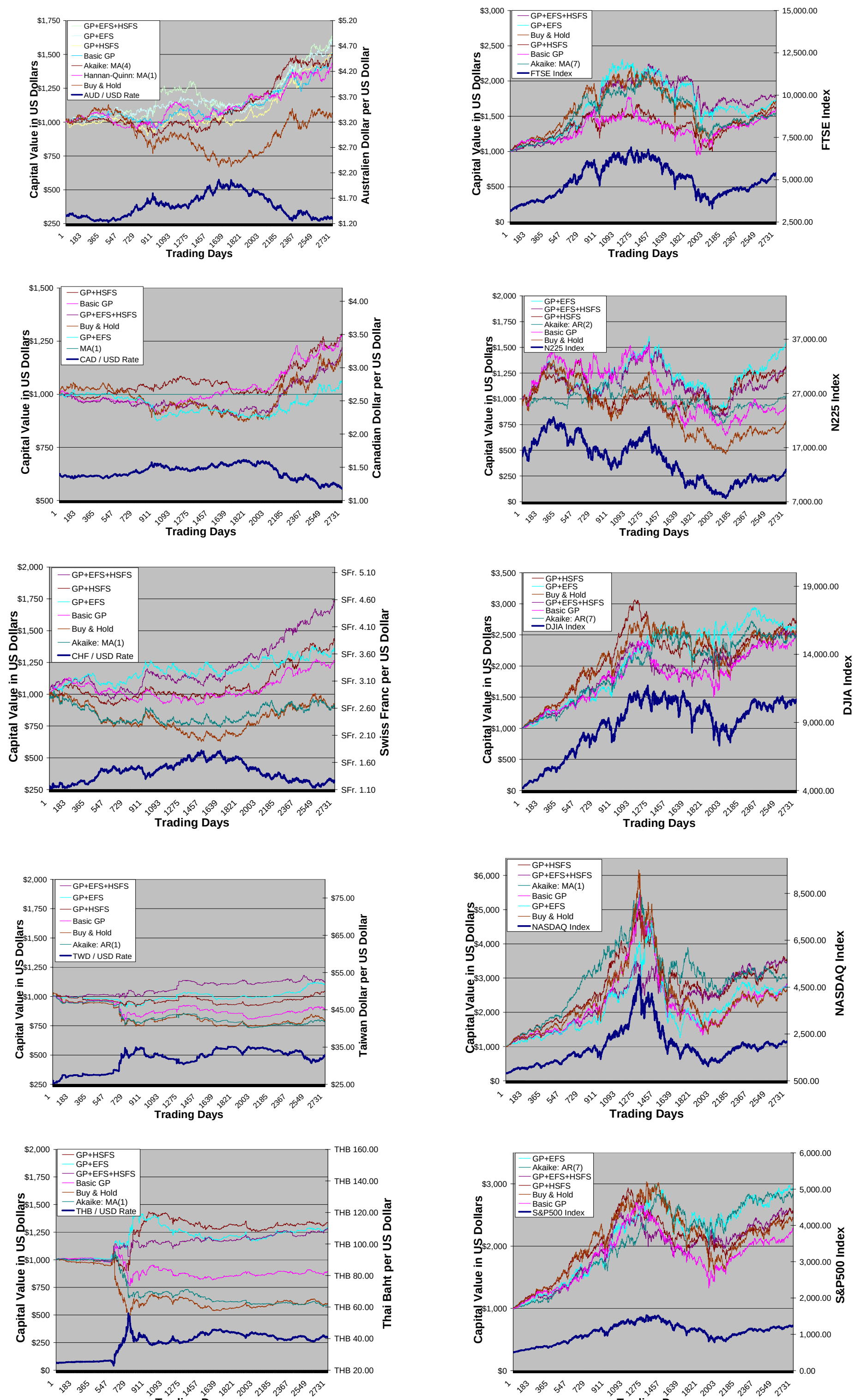
Table 5.13. Paired Difference t-Test on Profit Performance

Model	Akaike	Basic GP	GP+EFS	GP+HSFS	GP+EFS+HSFS
Buy & Hold	Failed	90%	**99%**	**99%**	**99%**
Akaike		Failed	**95%**	**98%**	**98%**
Basic GP			**95%**	**99%**	**99%**
GP+EFS				Failed	Failed
GP+HSFS					Failed

6. Discussion and Conclusions

We have shown that the Genetic Programming models outperformed the Akaike selected ARMA model and the simple Buy & Hold strategy using the HIT, the APC, and the Profit performance measures. A more detailed analysis of the performance over the 10-year testing period for the two major currencies shows that the addition of the trigonometric functions was adding better prediction power than the proposed high-order statistical functions. The combination of the trigonometric and statistical functions provided significantly better predictions than either individual function set.

One interesting aspect is that the British Pound exchange rate was in an overall upwards trend, whereas the Japanese Yen exchange rate was in a downward trend. Investing in the British Pound using a simple Buy & Hold strategy produced a 12.15% profit over the analyzed 10-year period. The Genetic Programming model utilizing both function sets was able to produce a profit of 88.78% over the same testing period. However, investing in the Japanese Yen using the simple Buy & Hold Strategy resulted in a loss of 20.71%. Our Genetic Programming approach with the combined function sets was still able to realize a calculated profit of 79.78% over the same testing period. We realize that the 79.78% profit is a rather low return for a 10-year period. However, the focus of this research has been the analysis of the impact of various function sets in the predictive power of a model, rather than integrating a larger number of other optimization approaches.



Figures 5.1. - 5.10. Profit performance of models for trading of Currencies and Stock Market Indices; left column top to bottom: Australian Dollar, Canadian Dollar, Swiss Franc, Taiwan Dollar, and Thai Baht; right column top to bottom: FTSE 100, NIKKEY 225, Dow Jones Industrial Average, NASDAQ, and S&P 500; Startup capital of $1,000; the order of models in the key corresponds to the final order on the right; bottom series is a plot of the corresponding financial time series.

The analysis of the performance for the additional financial time series demonstrates that the GP models with the added function sets clearly outperformed the Basic GP model, the Buy & Hold Strategy, and the Akaike selected ARMA models. The significance of the performance improvement of the various models was shown utilizing the Paired Difference t-Test.

Our results suggest that the implementation of trigonometric functions and our high-order statistical functions may be a valuable add-on for other genetic programming applications in this or other domains. The incorporation of additional statistical measures (e.g., Exponential Moving Average) may further improve the performance of our approach. We intend to perform further research in this area.

7. References

[1] Blicke, T. Evolving compact solutions in genetic programming: A case study. *Tik-report.* TIK Institute für Technische Informatik and Kommunikationsnetze, Computer Engineering and Networks Laboratory, ETH, Swiss Federal Institute of Technology, Switzerland, 1996.

[2] Koza, J.R. A genetic approach to truck backer upper problem and the inter-twined spiral. In *Proceedings of the IJCNN International Joint Conference on Neural Networks Volume IV,* IEEE Press, New York, NY, 1992, 310-318.

[3] Koza, J.R. and Kaene, M.A. Genetic breeding of non-linear optimal control strategies for broom balancing. In *Proceedings of the Ninth International Conference on Analysis and Optimization of Systems,* Antibes, France, June, 1990. Springer-Verlag, 1990, 47-56.

[4] Koza, J.R. Genetic Programming for Economic Modeling. In S. Goonatilaje and P. Treleaven (eds.), *Intelligent Systems for Finance and Business*, London: John Wiley & Sons, 1995, 251-269.

[5] Iba H. and Nikolaev H. Genetic Programming Polynomial Models of Financial Data Series. In *Proceedings of CEC 2000*, IEEE Press, 2000, 1459-1466.

[6] Santini, M. and Tettamazzi A. Genetic Programming for Financial Time Series Prediction. In P.L. Lanzi et al. (eds.), *Proceeding of EuroGP 2001*, LNCS 2038. Berlin: Spinger, 2001, 361-370.

[7] Chen, Shu-Heng (ed.) Genetic Algorithms and Genetic Programming in Computational Finance, Boston, MA, Kluwer, 2002.

[8] Koza, J.R. Genetic Programming: On the Programming of Computers by Means of Natural Selection. Cambridge, MA: MIT Press, 1992.

[9] Jain, R. The Art Of Computer System Performance Analysis: Techniques for Experimental Design, Measurement, Simulationm and Modeling. John Wiley & Sons, New York, NY, 1991.

[10] Weinbrenner, T. *"GPC++ - Genetic Programming C++ Class Library"*, Institute for Mechatronics, Technical University of Darmstadt, Germany, 1997.

[11] Pankratz, A. Forecasting With Univariate Box-Jenkins Models: Concepts and Cases. John Wiley & Sons, New York, NY, 1983.

[12] Iba, H. and Sasaki, T. Using Genetic Programming to Predict Financial Data. In *Proceedings of the 1999 Congress on Evolutionary Computation, CEC99,* IEEE Press, Piscataway, NY, 1999, 244-251.

[13] Bierens, H. J. *"EasyReg International"*, Department of Economics, Pennsylvania State University, University Park, PA, 2005.