APPLICATIONS OF ARTIFICIAL INTELLIGENCE IN FINANCE AND ECONOMICS
ADVANCES IN ECONOMETRICS

Series Editors: T. Fomby and R. Carter Hill
CONTENTS

LIST OF CONTRIBUTORS vii

STATISTICAL ANALYSIS OF GENETIC ALGORITHMS IN DISCOVERING TECHNICAL TRADING STRATEGIES
Chueh-Yung Tsao and Shu-Heng Chen 1

A GENETIC PROGRAMMING APPROACH TO MODEL INTERNATIONAL SHORT-TERM CAPITAL FLOW
Tina Yu, Shu-Heng Chen and Tzu-Wen Kuo 45

TOOLS FOR NON-LINEAR TIME SERIES FORECASTING IN ECONOMICS – AN EMPIRICAL COMPARISON OF REGIME SWITCHING VECTOR AUTOREGRESSIVE MODELS AND RECURRENT NEURAL NETWORKS
Jane M. Binner, Thomas Elger, Birger Nilsson and Jonathan A. Tepper 71

USING NON-PARAMETRIC SEARCH ALGORITHMS TO FORECAST DAILY EXCESS STOCK RETURNS
Nathan Lael Joseph, David S. Brée and Efstathios Kalyvas 93

CO-EVOLVING NEURAL NETWORKS WITH EVOLUTIONARY STRATEGIES: A NEW APPLICATION TO DIVISIA MONEY
Jane M. Binner, Graham Kendall and Alicia Gazely 127

FORECASTING THE EMU INFLATION RATE: LINEAR ECONOMETRIC VS. NON-LINEAR COMPUTATIONAL MODELS USING GENETIC NEURAL FUZZY SYSTEMS
Stefan Kooths, Timo Mitze and Eric Ringhut 145
FINDING OR NOT FINDING RULES IN TIME SERIES
Jessica Lin and Eamonn Keogh 175

A COMPARISON OF VAR AND NEURAL NETWORKS WITH GENETIC ALGORITHM IN FORECASTING PRICE OF OIL
Sam Mirmirani and Hsi Cheng Li 203

SEARCHING FOR DIVISIA/INFLATION RELATIONSHIPS WITH THE AGGREGATE FEEDFORWARD NEURAL NETWORK
Vincent A. Schmidt and Jane M. Binner 225

PREDICTING HOUSING VALUE: GENETIC ALGORITHM ATTRIBUTE SELECTION AND DEPENDENCE MODELLING UTILISING THE GAMMA TEST
Ian D. Wilson, Antonia J. Jones, David H. Jenkins and J. A. Ware 243
LIST OF CONTRIBUTORS

Jane M. Binner  
Department of Strategic Management, Aston Business School, Aston Triangle, Aston University, Birmingham, UK

David S. Brée  
Computer Science Department, University of Manchester, UK

Shu-Heng Chen  
AI-ECON Research Center, Department of Economics, National Chengchi University, Taiwan

Thomas Elger  
Department of Economics, Lund University, Sweden

Alicia Gazely  
Department of Information Systems and Management, Nottingham Trent University, UK

David. H. Jenkins  
School of Technology, University of Glamorgan, UK

Antonia J. Jones  
Department of Computer Science, Cardiff University, UK

Nathan Lael Joseph  
Aston Business School, Aston University, Aston Triangle, Birmingham, UK

Efstathios Kalyvas  
Computer Science Department, University of Manchester, UK

Graham Kendall  
School of Computer Science and Information Technology, University of Nottingham, UK

Eamonn Keogh  
Computer Science & Engineering Department, University of California – Riverside, USA

Stefan Kooths  
Muenster Institute for Computational Economics, Germany
Tzu-Wen Kuo  AI-ECON Research Center, Department of Economics, National Chengchi University, Taiwan

Hsi Cheng Li  Bryant University, Smithfield, USA

Jessica Lin  Computer Science & Engineering Department, University of California – Riverside, USA

Sam Mirmirani  Bryant University, Smithfield, USA

Timo Mitze  Gesellschaft fuer Finanz- und Regionalanalysen (GEFRA), Germany

Birger Nilsson  Department of Economics, Lund University, Sweden

Eric Ringhut  Muenster Institute for Computational Economics, Germany

Vincent A. Schmidt  Wright State University, Dayton, OH, USA

Jonathan A. Tepper  School of Computing and Technology, Faculty of Construction, The Nottingham Trent University, Nottingham, UK

Chueh-Yung Tsao  Department of Business Administration, Chang-Gung University, Taiwan

Andrew Ware  School of Computing, University of Glamorgan, UK

Ian D. Wilson  School of Computing, University of Glamorgan, UK

Tina Yu  ChevronTexaco Information Technology Company, San Ramon, CA, USA
INTRODUCTION

Artificial intelligence is a consortium of data-driven methodologies which includes artificial neural networks, genetic algorithms, fuzzy logic, probabilistic belief networks and machine learning as its components. We have witnessed a phenomenal impact of this data-driven consortium of methodologies in many areas of studies, the economic and financial fields being of no exception. In particular, this volume of collected works will give examples of its impact on the field of economics and finance. This volume is the result of the selection of high-quality papers presented at a special session entitled “Applications of Artificial Intelligence in Economics and Finance” at the “2003 International Conference on Artificial Intelligence” (IC-AI ’03) held at the Monte Carlo Resort, Las Vegas, NV, USA, June 23–26 2003. The special session, organised by Jane Binner, Graham Kendall and Shu-Heng Chen, was presented in order to draw attention to the tremendous diversity and richness of the applications of artificial intelligence to problems in Economics and Finance. This volume should appeal to economists interested in adopting an interdisciplinary approach to the study of economic problems, computer scientists who are looking for potential applications of artificial intelligence and practitioners who are looking for new perspectives on how to build models for everyday operations.

The structure of this volume is as follows; in the first chapter by Shu-Heng Chen and Chueh-Yung Tsao a statistical approach to testing the performance of GA-based trading strategies is proposed. The paper asks the question, what are the statistical properties which distinguish a successful application of GA from an unsuccessful one? The performance of ordinal GA-based trading strategies is evaluated under six classes of time series model, namely, the linear ARMA model, the bilinear model, the ARCH model, the GARCH model, the threshold model and the chaotic model. The performance criteria employed are the winning probability, accumulated returns, Sharpe ratio and luck coefficient. Asymptotic test statistics for these criteria are derived. The hypothesis as to the superiority of GA over a benchmark, say, buy-and-hold, is then be tested using Monte Carlo simulation. From this rigorously-established evaluation process, simple genetic algorithms are found to work very well in linear stochastic environments, and they are also found to work very well in nonlinear deterministic (chaotic) environments. However, they may perform much worse in pure nonlinear stochastic cases. These results shed
light on the superior performance of GA when it is applied to the two tick-by-tick time series of foreign exchange rates: EUR/USD and USD/JPY.

The second chapter by Tina Yu, Shu Heng Chen and Tzu-Wen Kuo models international short-term capital flow by identifying technical trading rules in short-term capital markets. Through the simulation, they investigate if there exist any trading strategies that are capable of predicting the capital inflow and out-flow, hence making investment profitable. The modelling and simulation were conducted using Genetic Programming (GP), a novel approach for this task. The simulation results suggest that the international short-term markets were quite efficient during the period of 1997–2002, with most GP generated trading strategies recommending buy-and-hold on one or two assets. The out-of-sample performance of GP trading strategies varies from year to year. However, many of the strategies are able to forecast Taiwan stock market down time and avoid making futile investment. Investigation of Automatically Defined Functions shows that they do not give advantages or disadvantages to the GP results.

The third chapter by Binner, Elger, Nilsson and Tepper contrasts the forecasting performance of two non-linear models, a regime-switching (RS) vector autoregressive model (VAR) and a recurrent neural network (RNN), to that of a linear benchmark VAR model. These models belong to different classes of non-linear models that are both econometrically challenging and therefore rarely compared. Evidence suggests that the RNN model and RS-VAR model outperform the VAR model for both monthly and annual forecast horizons. The RS-VAR and the RNN perform approximately on par over both forecast horizons. For the RS-VAR model, findings suggest that imposing the restriction that only the intercept is allowed to vary across regimes provides the best forecasts. For the RNN-model, the forecasting performance depends on the number of hidden units and thus free parameters included.

Nathan Joseph, David Brée and Efstathios Kalyvas ask the question, “Are the learning procedures of genetic algorithms (GAs) able to generate optimal architectures for artificial neural networks (ANNs) in high frequency data?” in paper four. The approach is in some respects similar in spirit to the use of bootstrapping to select suitable ANN structures. The architectures of the ANNs are also evaluated, both in- and out-of-sample, against a set of naïve RW models and the mean absolute error (MAE). The use of in-sample forecasts facilitates an assessment of the suitability of the chosen ANN configuration prior to implementation, while the use of RW models serves to evaluate the contribution of the ANNs to forecasting accuracy. No ANN architectures were able to outperform a random walk, despite the finding of non-linearity in the excess returns. This failure is attributed to the absence of suitable ANN structures and further implies that researchers need to be cautious when making inferences from ANN results that use high frequency data.
Chapter 5 uses the Artificial Intelligence techniques of evolutionary strategies and neural networks to evaluate the performance of simple sum monetary aggregates vis-à-vis their Divisia index counterparts in a simple inflation forecasting experiment. Jane Binner, Graham Kendall and Alicia Gazely find that that superior tracking of inflation is possible for models that employ a Divisia M2 measure of money that has been adjusted to incorporate a learning mechanism to allow individuals to gradually alter their perceptions of the increased productivity of money. Divisia measures of money outperform their simple sum counterparts as macroeconomic indicators. Co-evolution is found to compete very favourably with neural networks and has the potential to beat neural networks in terms of superior predictive performance when used to evolve neural networks. Artificial Intelligence techniques in general and co-evolution in particular are highly effective tools for predicting future movements in inflation and the paper concludes that there is tremendous scope for further research into the development of these methods as new macroeconomic forecasting models.

Stefan Kooths, Timo Mitze and Eric Ringhut seek to determine whether the predictive power of dynamic single-equation, linear econometric models outperform models based on a novel computational approach using genetic-neural fuzzy rule-bases when forecasting the EMU inflation rate in paper six. Evidence for the superiority of the computational approach based on genetic-neural fuzzy rule-bases (GENEFER) is found according to various different evaluation criteria. Especially striking is the ability of GENEFER models to predict turning points reliably. GENEFER models perform best within a 2-stage approach, where the disequilibrium (error-correction) terms from cointegration analysis are used as input variables. This result proposes a combination of econometric and computational techniques and calls for further research.

Given the recent explosion of interest in streaming data and online algorithms, clustering of time series subsequences has received much attention. In the seventh chapter, Jessica Lin and Eamonn Keogh make a surprising claim. Clustering of time series subsequences is completely meaningless. More concretely, clusters extracted from these time series are forced to obey a certain constraint that is pathologically unlikely to be satisfied by any dataset, and because of this, the clusters extracted by any clustering algorithm are essentially random. While this constraint can be intuitively demonstrated with a simple illustration and is simple to prove, it has never appeared in the literature. Jessica and Eamonn can justify calling their claim surprising, since it invalidates the contribution of dozens of previously published papers. They justify their claim with a theorem, illustrative examples, and a comprehensive set of experiments on reimplementations of previous work.
Sam Mirmirani and Hsi-Cheng Li apply VAR and ANN techniques to make ex-post forecast of U.S. oil price movements in the eighth chapter. The VAR-based forecast uses three endogenous variables: lagged oil price, lagged oil supply and lagged energy consumption. However, the VAR model suggests that the impacts of oil supply and energy consumption has limited impacts on oil price movement. The forecast of the genetic algorithm-based ANN model is made by using oil supply, energy consumption, and money supply (M1). Root mean squared error and mean absolute error have been used as the evaluation criteria. Their analysis suggests that the back-propagation network-GA model noticeably outperforms the VAR model.

Vincent Schmidt and Jane Binner demonstrate how neural network models (such as the Aggregate Feedforward Neural Network) provide beneficial information in the domain of discovering and describing the money-price relationship using Divisia component data in the ninth chapter. The AFFNN is demonstrated as being straightforward to design and use with encoded Divisia component and inflation data, and the model is able to effectively learn the relationships within the dataset. A simple decompositional rule extraction technique examines the learned knowledge and automatically generates a collection of if-then rules in terms of the original attribute values. These Divisia rules are suitable for examination by subject-matter experts (specifically, econometricians). The rules potentially provide interesting and useful insight into monetary aggregation theory, particularly when exploring the relationships between various monetary assets and the corresponding growth rate of prices. As an additional advantage, the resulting rules are expressed in well-documented computer code, capable of being executed for validation or used for forecasting purposes.

Ian Wilson, Antonia Jones, David Jenkins and Andrew Ware show, by means of an example of its application to the problem of house price forecasting, an approach to attribute selection and dependence modelling utilising the Gamma Test (GT), a non-linear analysis algorithm that is described. The GT is employed in a two-stage process: first the GT drives a Genetic Algorithm (GA) to select a useful subset of features from a large dataset that is developed from eight economic statistical series of historical measures that may impact upon house price movement. Next a predictive model is generated utilising an Artificial Neural Network (ANN) trained to the Mean Squared Error (MSE) estimated by the GT, which accurately forecasts changes in the House Price Index (HPI). They present a background to the problem domain and demonstrate, based on results of this methodology, that the GT was of great utility in facilitating a GA based approach to extracting a sound predictive model from a large number of inputs in a data-point sparse real-world application.

There are still many important Artificial Intelligence disciplines yet to be covered. Among them are the methodologies of independent component analysis,
reinforcement learning, inductive logical programming, classifier systems and Bayesian networks, not to mention many ongoing and highly fascinating hybrid systems. A way to make up for their omission is to visit this subject again later. We certainly hope that we can do so in the near future with another volume of “Applications of Artificial Intelligence in Economics and Finance.”
STATISTICAL ANALYSIS OF GENETIC ALGORITHMS IN DISCOVERING TECHNICAL TRADING STRATEGIES

Chueh-Yung Tsao and Shu-Heng Chen

ABSTRACT

In this study, the performance of ordinal GA-based trading strategies is evaluated under six classes of time series model, namely, the linear ARMA model, the bilinear model, the ARCH model, the GARCH model, the threshold model and the chaotic model. The performance criteria employed are the winning probability, accumulated returns, Sharpe ratio and luck coefficient. Asymptotic test statistics for these criteria are derived. The hypothesis as to the superiority of GA over a benchmark, say, buy-and-hold, can then be tested using Monte Carlo simulation. From this rigorously-established evaluation process, we find that simple genetic algorithms can work very well in linear stochastic environments, and that they also work very well in nonlinear deterministic (chaotic) environments. However, they may perform much worse in pure nonlinear stochastic cases. These results shed light on the superior performance of GA when it is applied to the two tick-by-tick time series of foreign exchange rates: EUR/USD and USD/JPY.

1. INTRODUCTION

Genetic algorithms (GAs) have been developed by Holland (1975) to mimic some of the processes observed in natural evolution. They are based on the
genetic processes of natural selection which have become widely known as the “survival of the fittest” since Darwin’s celebrated work. In recent years, GAs have been successfully applied to find good solutions to real-world problems whose search space is complex, such as the traveling salesman problem, the knapsack problem, large scheduling problems, graph partitioning problems, and engineering problems, too.¹

In finance, Bauer (1994) provides the first application of GAs to discover trading strategies. Since then, GAs have gradually become a standard tool for enhancing investment decisions.² While many studies have supported the effectiveness of GAs in investment decisions; however, the foundation of these applications has not been well established. The thing that concerns us, therefore, is the robustness of these empirical results. For example, if GAs are effective for the investment in one market at one time, would the same result apply to the same market or different markets at different times? It is for the purpose of pursuing this generality, that we see the necessity of building a solid foundation upon which a rigorous evaluation can be made.

In this paper, a statistical approach to testing the performance of GA-based trading strategies is proposed. Instead of testing the performance of GAs in specific markets as a number of conventional studies already have, we are interested in a market-independence issue: what makes GAs successful and what makes them not? Since the data to which GAs are applied consist of financial time series, the question can be rephrased as follows: what are the statistical properties which distinguish a successful application of GA from an unsuccessful one? One way to think of the question is to consider two markets following different stochastic processes. One market follows stochastic process A, and the other stochastic process B. If GAs can work well with stochastic process A, but not B, then the successful experience of GAs in the first market is certainly not anticipated in the second market.

Having said that, this paper follows the following research methodology. First, some financially-related stochastic processes are singled out as the standard scenarios (testbeds) to test the performance of GA. Second, appropriate performance criteria are used to evaluate the performance of the GA over these testbeds. Third, the associated asymptotic statistical tests are applied to examine whether the GAs perform significantly differently as opposed to a familiar benchmark. By this procedure, we may be able to distinguish the processes in which the GA has competence from others in which it does not. Once the critical properties are grasped, we can then apply the GA to the financial time series whose stochastic properties are well-known, and test whether the GA behaves consistently with what we have learned from the previous statistical analysis.

By means of the procedure established in this paper, we hope to push forward the current applications of GAs or, more generally, computational intelligence (CI),
toward a more mature status. After all, whether GA will work has been asked too intensely in the literature. The very mixed results seem to suggest that we look at the same question at a finer level and start to inquire why it works or why it doesn’t. We believe that there are other ways to do something similar to what we propose in this paper. We do not exclude these possibilities. In fact, little by little, these efforts will eventually enable GA or CI tools to rid themselves of their notoriety for being blackboxes.

The rest of the paper is organized as follows. Section 2 introduces a specific version of GA, referred as to the ordinary GA (OGA), used in this paper. Section 3 will detail the classes of stochastic processes considered in this paper and the reasons for this choice. Section 4 reviews the four performance criteria and establishes their associated asymptotic test. Section 5 sets up the Monte Carlo simulation procedure. Section 6 summarizes and discusses the actual performance of the GA over the artificial data, whereas the counterpart over the real data is given in Section 7. Section 8 concludes this paper.

2. TRADING WITH GAS

A trading strategy $g$ can be formally defined as a mapping:

$$g: \Omega \rightarrow \{0, 1\}.$$ (1)

In this paper, $\Omega$ is assumed to be a collection of finite-length binary strings. This simplification can be justified by the data-preprocessing procedure which transforms the raw data into binary strings. The range of the mapping $g$ is simplified as a 0–1 action space. In terms of simple market-timing strategy, “1” means to “act” and “0” means to “wait.” Here, for simplicity, we are only interested in day trading. So, “act” means to buy it at the opening time and sell it at the closing time.

Like all financial applications of GA, the start-off question is the representation issue. In our case, it is about how to effectively characterize the mapping $g$ by a finite-length binary string, also known as a chromosome in GA. Research on this issue is very much motivated by the format of existing trading strategies, and there are generally two approaches to this issue. The first approach, called the decision tree approach, was pioneered by Bauer (1994). In this approach each trading strategy is represented by a decision tree. Bauer used bit strings to encode these decision trees, and generated and evolved them with genetic algorithms. The second approach, called the combinatoric approach, was first seen in Palmer et al. (1994). The combinatoric approach treats each trading strategy as one realization from $\binom{n}{k}$ combinations, where $l \leq k \leq n$, and $n$ is the total number of given trading rules. Using GAs, one can encode the inclusion or exclusion of a
specific trading rule as a bit and the whole trading strategy as a bit string (chromosome).

Both approaches have very limited expression power. While various enhancements are possible, they all lead to non-standard GAs in the sense that their representations are not based on finite-length binary strings. Since the main focus of this paper is to illustrate a statistical foundation of the GA, we try to avoid all unnecessary complications, including the use of those non-standard representations. In other words, at this initial stage, we only make the illustration with the ordinary genetic algorithm (OGA), and, for that reason, Bauer’s simple decision-tree representation is employed. However, it is clear that the statistical foundation presented in this paper is also applicable to GAs with different representations.

Bauer’s decision-tree representation corresponds to the following general form of trading strategies

\[
\text{IF (CONDS)} \quad \text{THEN (BUY AND SELL [DAY TRADING])} \quad \text{ELSE (WAIT))}.
\]

The CONDS appearing in the trading strategy is a predicate. CONDS itself is a logical composition of several primitive predicates. In this paper, all CONDSs are composed of three primitive predicates. Each primitive predicate can be represented as:

\[
\text{Cond}(Z) = \begin{cases} 
1(\text{True}), & \text{if } Z \oplus a, \\
0(\text{False}), & \text{if } Z \ominus a
\end{cases}
\]  

(2)

where \( Z \), in our application, can be considered as a time series of returns indexed by \( t \), e.g. \( r_{t-1}, r_{t-2}, \text{etc.} \), and \( a \) can be regarded as a threshold or critical value \((a \in \mathbb{N}, \text{a set of integers})\).\( \oplus \in \{\geq, <\} \) and \( \ominus = \{\geq, <\} - \oplus \). An example of CONDS with three primitive predicates is

\[
\text{CONDS}(r_{t-1}, r_{t-2}, r_{t-3}) = \text{Cond}(r_{t-1}) \lor (\text{Cond}(r_{t-2}) \land \text{Cond}(r_{t-3})),
\]  

(3)

where “\( \lor \)” refers to the logic operator “OR,” and “\( \land \)” refers to “AND.”

Following Bauer, we use a 21-bit string to encode a trading strategy of this kind. Details can be found in the Appendix (Section A.1). Let \( G \) be the collection of all trading strategies encoded as above. Then the cardinality of \( G \) is \( 2^{21} \) \((\#(G) = 2^{21})\), which is more than 2 million. The search over the space \( G \) can be interpreted as a numerical algorithm as well as a machine learning algorithm for solving a mathematical optimization problem. Without losing generality, consider
the trading strategy with only one primitive predicate,

\[ \text{Cond}(Z) = \begin{cases} 1(\text{True}), & \text{if } r_{t-1} \geq a, \\ 0(\text{False}), & \text{if } r_{t-1} < a. \end{cases} \] (4)

Suppose the stochastic process of \( r_t \) is strictly stationary and denote the joint density of \( r_{t-1} \) and \( r_t \) by \( f(r_{t-1}, r_t) \). In this simplest case, a trading strategy is parameterized by a single parameter \( a \). Denote it by \( g_a \). Then the optimal strategy \( g_{a^*} \) can be regarded as a solution to the optimization problem

\[ \max_a E(\ln(\pi_n)), \] (5)

where

\[ \pi_n = \prod_{t=1}^{n} (1 + r_t) \] (6)

is the accumulated returns of \( g_a \) over \( n \) consecutive periods. It can be shown that the solution to the problem (5) is

\[ a^* = F^{-1}(0), \quad \text{if } F^{-1}(0) \text{ exists.} \] (7)

where

\[ F(a) = \int_{-\infty}^{\infty} \ln(1 + r_t)f(a, r_t) \, dr_t \] (8)

To solve Eq. (7), one has to know the density function of \( f(r_{t-1}, r_t) \), which can only be inferred from the historical data. In this case, GAs are used as a machine learning tool to obtain an estimate of this joint density. Also, to arrive at a value for \( a^* \), we have to know the inverse function of \( F(a) \), which in general can only be solved numerically. In this case, GAs are used as a numerical technique to solve this problem. Therefore, in the trading-strategy problem, GAs are used simultaneously as a numerical technique and a machine learning tool to determine the critical parameter \( a^* \). In the general case when CONDS has more than one predicate, the mathematical formulation of the problem can become very complicated, but the dual role of GAs remains unchanged. This interpretation justifies the mathematical significance of using GAs to discover the trading strategies.

The GA employed in this paper is a very basic version, which we shall call the ordinary genetic algorithm (OGA). In this study, we only focus on the OGA. Nonetheless, in a further study, it will be interesting to see whether a better result can be expected from advanced versions of GAs. The technical details of the OGA are given in the Appendix (Section A.2).