A genetic programming model to generate risk-adjusted technical trading rules in stock markets

Akbar Esfahanipour *, Somayeh Mousavi

Industrial Engineering Department, Amirkabir University of Technology, P.O. Box 15875-4413, Tehran, Iran

ARTICLE INFO

Keywords:
Genetic programming
Technical trading rules
Risk-adjusted measures
Conditional Sharpe ratio
Tehran Stock Exchange (TSE)

ABSTRACT

Technical trading rules can be generated from historical data for decision making in stock markets. Genetic programming (GP) as an artificial intelligence technique is a valuable method to automatically generate such technical trading rules. In this paper, GP has been applied for generating risk-adjusted trading rules on individual stocks. Among many risk measures in the literature, conditional Sharpe ratio has been selected for this study because it uses conditional value at risk (CVaR) as an optimal coherent risk measure. In our proposed GP model, binary trading rules have been also extended to more realistic rules which are called trinary rules using three signals of buy, sell and no trade. Additionally we have included transaction costs, dividend and splits in our GP model for calculating more accurate returns in the generated rules. Our proposed model has been applied for 10 Iranian companies listed in Tehran Stock Exchange (TSE). The numerical results showed that our extended GP model could generate profitable trading rules in comparison with buy and hold strategy especially in the case of risk adjusted basis.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

Technical analysis is a broad collection of methods and strategies which attempt to exploit the short term fluctuations in the stock markets. In this approach trading rules are generated from historical data, to determine the right time for buying and selling of the securities. Traders use these trading rules to profit from active and frequent trades in stock markets. This approach is called buy and sell strategy. An alternative approach which is known as buy and hold strategy is a more passive investment strategy. In this approach, investors buy and hold the assets for a long period of time, regardless of small fluctuations. Traders believe that they can earn more profit than investors. However, the profitability of technical analysis has been criticized by two hypotheses namely "random walk hypothesis" and "efficient market hypothesis". These hypotheses state that there should not be any discernable and exploitable pattern in the financial data. According to these hypotheses, traders could not profit from technical trading rules. Although the first studies in 1960s and 1970s supported these hypotheses (Alexander, 1964; Fama, 1970; Fama & Blume, 1966; Jensen & Bennington, 1970), however investors were reluctant to abandon their charts and rules. According to Taylor reports, up to 90% of traders use some sort of technical analysis in their trading decision makings (Taylor, 2000).

These hypotheses are also rejected by most of the academic researches which demonstrated technical trading rules could be profitable (Bessembinder & Chan, 1995; Brock, Lakonishok, & LeBaron, 1992; Pruitt & White, 1988). They compared the return of technical trading rules with the return of buy and hold strategy and found that positive excess returns can be achieved using technical trading rules. By positive excess returns, we mean that the buy and sell strategy is superior to the buy and hold strategy in terms of resulted returns.

Above mentioned studies applied classical techniques for generating trading rules such as moving average and trading range break. New technologies such as artificial intelligent systems look more promising, because they allow a system to automatically generate and adapt profitable trading rules. For instance, genetic algorithms, genetic programming and neural networks are very successful in technical analysis (Bauer, 1994; Chou, Hsu, Yang, & Lai, 1997).

Bauer (1994) used genetic algorithm to exploit technical trading rules in the US exchange market. These trading rules led to positive excess returns. However genetic programming (GP) seems to be more appropriate in extracting trading rules from historical data because of its structure (Potvin et al., 1994). In GP approach for rule discovery, rules are extracted in the form of decision trees from the past data. Allen and Karjalainen (1999) used genetic programming to generate technical trading rules on S&P500 data. They found that the transaction cost adjusted returns failed to obtain positive excess returns. Although the first studies such Allen and Karjalainen could not reject the EMH, but later studies demonstrated otherwise. Neely, Weller, and Dittmar

* Corresponding author. Tel: +98 21 6454 5369; fax: +98 21 6695 4569.
 E-mail addresses: esfahaa@aut.ac.ir (A. Esfahanipour), s_moosavi@aut.ac.ir (S. Mousavi).

Table 1
Position of this study among the related studies in the literature.

Authors (year)	Outperformed buy and hold	Transaction cost	Applied risk measure	Dividend and splits	Case study
Allen and Karjalainen (1999)		☐			S&P500 index
Neely et al. (1997)	☐	☐			6 EMS exchange rates
Neely and Weller (1999)	☐	☐			4 EMS exchange rates
Neely (2001)		☐	Sharpe ratio, Jensen's alpha, etc.		S&P500 index
Potvin et al. (2004)					14 Canadian companies
Fyfe et al. (2005)		☐	Sharpe ratio		S&P indices
Mallick et al. (2008)	☐	☐			30 DJIA companies
Esfahanipour et al. (2009)	☐	☐			9 Iranian companies
This study	☐	☐	Conditional Sharpe ratio	☐	10 Iranian companies

(1997) used genetic programming to find technical trading rules for the main six currencies on foreign exchange market. Also Neely and Weller (1999) found the same results for three of four European monetary system (EMS) exchange rates. Another considerable work of Neely (2001) was in equity markets on S&P500. Despite that Neely considered risk adjusted excess returns, he could not reject the EMH which was inconsistent with Allen and Karjalainen (1999)'s claim. In another work, technical trading rules outperform the buy and hold strategy in risk unadjusted basis, but underperform when risk is considered in the case of three S&P indices (Fyfe, Marney, & Tarbert, 2005).

All of the above-mentioned researches considered global market indices (e.g., Dow Jones, S&P500) and generated trading rules for these indices. They investigated the profitability of technical analysis and two critic hypotheses. Another promising approach is to consider stocks offered by individual companies. This approach looks more applicable since each stock is investigated separately and each rule is generated for one stock. Potvin, Soriano, and Vall'ee (2004) applied this approach for 14 Canadian companies listed on Toronto stock exchange. Their results showed that trading rules are generally beneficial when the market falls or when it is stable. Despite the attractiveness of this work, they did not consider transaction cost for evaluating fitness of generated rules. Transaction cost is brokerage fees that are payable for each trade carried out. Since many trades take place in the buy and sell strategy, transaction cost would affect the profitability of trading rules. Hence, Mallick, Lee, and Ong (2008) considered transaction cost in their GP model and applied it for thirty component stocks of Dow Jones Industrial Average index (DJIA). Their statistical results confirm that the GP based trading rules generate a positive excess return over the simple buy and hold strategy, under all market conditions whether rising or falling. Also genetic programming technique has been applied to generate transaction cost adjusted trading rules for nine Iranian companies (Esfahanipour, Karimi, & Mousavi, 2009). The later study showed that GP could generate profitable trading rules in comparison with the buy and hold strategy especially for companies having frequent trades in the market.

Although risk is an important factor in trading decisions, however it is not considered in Potvin et al. (2004), Mallick et al. (2008), Esfahanipour et al. (2009). In these studies raw excess returns are evaluated rather than their risk adjusted excess returns. Also all of the above-mentioned studies generated trading rules using the historical data of stock prices and/or trading volumes. They did not consider other effective parameters on return such as dividends and splits. Investigating the structure of GP trading rules in previous studies, we found they generate trading rules with two buy and sell signals. In fact, they assumed that trading should be practiced every day. Since sometimes no trade is the best decision in stock trading, a useful extension of trading rules is to include "no trade" signal as well. In this approach the structure of trading rules should change to carry out three signals as buy, sell and no trade.

Therefore, our goal here is to explore the application of GP for generating three signals technical trading rules on individual stocks in the case of risk adjusted measures and transaction cost. It also includes all effective factors such as dividends and splits. Table 1 summarizes previous works which used GP to generate trading rules in comparison with this study.

The reminder of this paper has been organized as follows. In the next section the GP algorithm is introduced. Then risk adjusted measures are investigated and one measure is selected for our GP model. In Section 4 a structure is proposed to extend GP based trading rules with three signals. A GP model is presented to generate risk adjusted trading rules in Section 5. Then our extended GP model is implemented on 10 Iranian companies and computational results are reported. The paper closes with our conclusion.

2. Genetic programming

Genetic programming as an artificial intelligence technique has recently been used successfully to extract knowledge in the form of IF–THEN rules and has been utilized in various fields particularly in finance and technical analysis (Chou et al., 1997; Engelbrecht & Schoeman, 2002). Koza (1992) developed this technique for the first time as an extension of genetic algorithm (GA) (Holland, 1975). The main difference between GP and GA is the representation of the solution. In GP, the individual population members are not fixed length character strings that encode possible solutions to the problem at hand, they are programs that, when executed, are the candidate solutions to the problem. These programs are expressed in genetic programming as parse trees, rather than as lines of code (Abraham, Nedjah, & Mourelle, 2006).

The basic steps in a GP system are as follows (Poli, Langdon, & McPhee, 2008):

1. Randomly create an initial population of programs from the available primitives.
2. Repeat
   2.1. Execute each program and ascertain its fitness.
   2.2. Select one or two program(s) from the population with a probability based on fitness to participate in genetic operations.
   2.3. Create new individual program(s) by applying genetic operators with specified probabilities.
3. Until an acceptable solution is found or some other stopping condition is met (e.g., a maximum number of generations is reached).
4. Return the best-so-far individual.

After introducing risk adjusted measures and trinary trading rules, each step of GP is explained and extended for generating trading rules in the following sections.