Introduction to Artificial Intelligence Writing 5

YiFan Li
ZeYuan Yang

December 2, 2019

1   Introduction

Predicting the future can only happen in science fictions, but predicting the future trend of some certain time series is feasible. Time series analysis has a long history, and multiple approaches have been studied.

The most traditional methods are statistical ones such as the well-known ARIMA model. Although the model is simple, it is still used in recent researches such as [18]. Some delicately designed machine learning approaches are also introduced in these years, varying from relatively simple support vector machine [14] to complex neural network [6].

The one that receives the most attention among all time series is probably the stock price. Firstly, as an essential part of the financial market, the stock market is the apple of investors' eyes. Not only financial researchers but also individuals and securities companies have tried their best to analyze the stock price time series and predict its behavior in the future, to make a fortune. Secondly, unlike most time series prediction tasks where the output must be exact future values, only knowing whether the future price will go up or down is already sufficient to make a trading decision. Therefore, the problem can be simplified thus enabling more ideas and approaches to be applicable. Thirdly, sufficient financial indexes and trading rules can serve as domain knowledge to be combined with computer algorithms.

Nowadays, there are numerous financial indexes, and over two hundred different trading rules were built upon them to suggest buying, selling or holding decisions, based on historical stock price time series [9]. Each rule has its rationality and will indicate some features of the target stock. However, the movement of a stock price is the combined result of a massive number of microeconomic and macroeconomic factors, which makes it impossible to predict the trend with any index or rule solely. As Korczak [9] pointed out, there is no rule consistently outperforming the others. Therefore, finding the most appropriate way to utilize them becomes the key to the problem.

Although some scholars argue that stock prices follow a random walk pattern, most experts in this field believe that the prices can be predicted with more than 50 percent accuracy, with eligible trading rules and reasonable combination [16]. Plenty of approaches has been examined and we will go through some of them in the literature review. Noticeably, the genetic algorithm is widely used in related researches.

Genetic algorithm is an adaptive heuristic search method, inspired by the natural process of genetic evolution. It can quickly generate high-quality solutions to optimization problems, such as scheduling and shortest path, and modeling system where randomness is involved, such as the stock market, by simulating the selection, crossover and mutation procedures in nature [10].

In our paper, we will apply the genetic algorithm to find the best combination of a set of trading rules, which can help with the buy and sell decisions. We will dig deeper into a more comprehensive and applicable solution based on the previous works.

2   Related Work

We will go through related works about the genetic algorithms first, then look into its application on the stock time series prediction.

2.1   Genetic Algorithm

The genetic algorithm was invented in the early 1970s by John Holland [10]. It is based on the mechanics of natural genetic selection. To conduct GA, a "chromosome" must be defined to represent a solution to the problem. A chromosome is a sequence of alleles, such as a bit. The genetic features of such representation enable crossover and mutation, in order to exchange information with each other [12]. A fitness function, which can evaluate the solution, has to be defined as well. It will be used to select those chromosomes that survive.

The general steps of the genetic algorithm are as follows:

1. Initialization. Some chromosomes are randomly generated to form a population.
2. Selection. During each generation, a proportion of the existing population will be selected through a process, where fitter solutions, measured by the fitness function, are more likely to be selected. They are used to generate the population for the next generation.
3. Reproduction. A second generation population is bred by the previous generation. There are typically two genetic operators involved in this step:
   (a) Crossover. Parent chromosomes are selected from the survivors and their genes are combined in some certain way to create a new chromosome.
   (b) Mutation. Some genes in some chromosomes are randomly chosen and changed.

   Generally, the population size is preserved throughout each generation, so the selection and reproduction process should counterbalance each other's effect on the number of chromosomes.
4. Termination. The selection and reproduction are repeated several times until a termination condition has been reached. Common termination conditions can be:
   (a) A solution is found which satisfies a minimum fitness criteria
   (b) A fixed number of iteration is reached
   (c) The fitness value of the best chromosomes in several successive generations is not improving

As the genetic algorithm keeps developing, several variants are proposed. The first one is the chromosome representation. Traditionally, the chromosome consists of a series of binary bits. However, in some problem settings, the gene can be a floating-point number [8], or even complex data structures like list or map [15]. The second one is elitism [2], which requires the selection and reproduction to be carefully designed so that the several best chromosome must be preserved to the next generation. The third one is the adaptive genetic algorithm [17]. Unlike the traditional ones whose strategy of selection and reproduction is constant, the possibility of selection, crossover, and mutation will vary in adaptive GA. The adjustment of these possibilities is related to the current fitness value. All these variants improved some aspect of the genetic algorithm, and we will implement some of them in our experiment.

2.2   Application of GA to Stock Price Prediction

Given the stock price time series as the original data, Stephen Leung et al. [3] modeled it as a fuzzy time series by converting the percentage of the price change into a fuzzy value. The fuzzy logic relationship is extracted and classified to form fuzzy logic relationship groups (FLRGs). FLRGs will then be normalized to get a weight matrix. Each gene in the chromosome corresponds to a fuzzy value and the whole chromosome will output a prediction. The fitness function used in the genetic algorithm is:

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^{n} \left(\text{actual}(t) - \text{predicted}(t)\right)^2}{n}}$$

One technique in their approach is worth mentioning. The tournament method is used to select survivors in the previous generation. Several "tournaments" among a few chromosomes chosen at random from the population are conducted. The winner of each tournament (the one with the best fitness) is selected for crossover. Comparing with the traditional approach of directly selecting the good ones from the whole population, it can improve gene diversity thus prevent GA from being trapped in local optimum [7].

Applying the fuzzy logic as well, Kuo, Ren Jie and his team [11] further choose to combine genetic and neural networks. They developed a stock trading system, using GFNN (genetic-algorithm-based fuzzy neural networks) to build the qualitative model. The basic model is an FNN (fuzzy neural network), and the genetic algorithm is used to provide the initial weights and select hyperparameters. This research and some similar ones utilized the optimization feature of the genetic algorithm, but put more emphasis on other techniques.

In addition to those combining multiple methodologies to build a complex model without the help of domain knowledge, more studies use only GA as the algorithm and take the financial indexes and trading rules into account. They no longer produce exact value for future time series but provide trading decisions. As we mentioned before, there are numerous financial indexes, each summarizing one feature of the stock price time series. They can be used to build trading rules, which will indicate a decision for buying, selling or holding. It is the strategy to utilize them that is optimized in those researches. However, the way they model the problem setting may be slightly different.

Lin Li et al. [13] arbitrarily picked one trading rule, "the filter rule", and optimize the choice of its parameters. The fitness function is the profit of trading using its decision. Comparing with brute force grid search, genetic algorithm saved a lot of running time, but loses very little precision. Nevertheless, only using one trading rule limits the model's performance, hence the profit is still significantly lower than that of a human trader. Most studies are focused on the way of combining multiple financial indexes.

Fuentu et al. [5] selected three financial indexes, RSI, MACD, and Stochastic Index, and five of their variants are coded in the algorithm. The goal is to find the best time to buy stock, indicated by the values of the five index variants. Each gene in the chromosome represents an interval of the value of a financial index, and the fitness function is calculated as below:

1. If there is a period that could match all the financial indexes represented by the chromosome, the chromosome's fitness value is the profit of making the buying decision at the start of that period.
2. If there is no such period, the chromosome will not be selected into the next generation.

In the future, when the five financial indexes have the value represented by the best chromosome, it can be concluded to be the best time to buy a stock.

In some other researches [1] [9], the suggestions of trading rules are integrated to give a final trading decision and such integration is optimized with the genetic algorithm.

Firstly, a set of representative trading rules are selected to examine, such as Moving Average and Relative Strength Index. Each gene in the chromosome corresponds to one rule, with only 1 and 0 as their possible values representing whether the rule should be considered in the final decision. The final decision is the voting result of all the rules considered. On each day in the stock time series, the chromosome will give such a final decision, and the fitness function is the profit made by these decisions. After that, a general genetic algorithm can be applied to find the best chromosome.

This approach is intuitive and relatively easy to implement. Rohit and Kumkum [4] take advantage of it and further uses support vector machine to determine the weight of each rule in the final decision.

Two concerns about this approach are raised, though. Firstly, the previous experiments are always run on one stock. Whether this methodology is suitable for more stocks in the market remains a problem [1]. Secondly, whether the resulting chromosome, i.e. combination of trading rules, remains useful as time goes by is unknown. Some experts argue that the influence of time is not obvious, but it still requires examination [9]. Our work will be based on the approach introduced above, try to address these two issues and hopefully find other insights.

References

[1] F. A. Badawy, H. Y. Abdelazim, and M. G. Darwish. Genetic algorithms for predicting the egyptian stock market. In *2005 International Conference on Information and Communication Technology*, pages 109–122, Dec 2005.

[2] S. Baluja and R. Caruana. Removing the genetics from the standard genetic algorithm. In *Machine Learning Proceedings 1995*, pages 38–46. Elsevier, 1995.

[3] Q. Cai, D. Zhang, B. Wu, and S. C. Leung. A novel stock forecasting model based on fuzzy time series and genetic algorithm. *Procedia Computer Science*, 18:1155–1162, 2013.

[4] R. Choudhry and K. Garg. A hybrid machine learning system for stock market forecasting. *World Academy of Science, Engineering and Technology*, 39, 01 2008.

[5] D. de la Fuente, A. Garrido, J. Laviada, and A. Gómez. Genetic algorithms to optimise the time to make stock market investment. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pages 1857–1858. ACM, 2006.

[6] R. J. Frank, N. Davey, and S. P. Hunt. Time series prediction and neural networks. *Journal of Intelligent and Robotic Systems*, 31(1):91–103, May 2001.

[7] D. E. Goldberg et al. A note on boltzmann tournament selection for genetic algorithms and population-oriented simulated annealing. *Complex Systems*, 4(4):445–460, 1990.

[8] C. Z. Janikow and Z. Michalewicz. An experimental comparison of binary and floating point representations in genetic algorithms. In *ICGA*, pages 31–36, 1991.

[9] J. Korczak and P. Roger. Stock timing using genetic algorithms. *Applied Stochastic Models in Business and Industry*, 18(2):121–134, 2002.

[10] M. Kumar, M. Husian, N. Upreti, and D. Gupta. Genetic algorithm: Review and application. *International Journal of Information Technology and Knowledge Management*, 2(2):451–454, 2010.

[11] R. J. Kuo, C. Chen, and Y. Hwang. An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network. *Fuzzy sets and systems*, 118(1):21–45, 2001.

[12] L. Lemarchand, A. Plantec, B. Pottier, and S. Zanati. An object-oriented environment for specification and concurrent execution of genetic algorithms. In *ACM SIGPLAN OOPS Messenger*, volume 4, pages 163–165. ACM, 1992.

[13] L. Lin, L. Cao, J. Wang, and C. Zhang. The applications of genetic algorithms in stock market data mining optimisation. *Management Information Systems*, 2004.

[14] K.-R. Müller, A. Smola, G. Rätsch, B. Schölkopf, J. Kohlmorgen, and V. Vapnik. Using support vector machines for time series prediction. *Advances in kernel methods—support vector learning*, pages 243–254, 1999.

[15] M. Patrascu, A. F. Stancu, and F. Pop. Helga: a heterogeneous encoding lifelike genetic algorithm for population evolution modeling and simulation. *Soft Computing*, 18(12):2565–2576, Dec 2014.

[16] B. Qian and K. Rasheed. Stock market prediction with multiple classifiers. *Applied Intelligence*, 26(1):25–33, Feb 2007.

[17] M. Srinivas and L. M. Patnaik. Adaptive probabilities of crossover and mutation in genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics*, 24(4):656–667, 1994.

[18] W. Suwardo, M. Napiah, and I. Kamaruddin. Arima models for bus travel time prediction. *Journal of the institute of engineers Malaysia*, pages 49–58, 2010.