Date: 05/12/2019
Name: David Goerig (djg53)
School: School of Computing
Programme of Study: Advanced Computer Science (Computational Intelligence)
Module name: C0837 Natural Computation
E-mail: djg53@kent.ac.uk
Academic Adviser: DR P Kenny (P.G.Kenny@kent.ac.uk)

Coursework 2: Short essay
Applications of Evolutionary Algorithms in Finance (focus on stock market trading rules)

1. Abstract

The area of finance contains a lot of applications of Evolutionary Algorithms like portfolio selection [1], stock market data mining optimisation [2], etc. The use of evolutionary algorithms started around 1980s, but with the stock exchange, deterministic and analytical manner was longer preferred[3]. So we will focus on stock market algorithms, and we will take as an example an evolutionary algorithms on the Madrid Stock Exchange General Index (IGBM), where we will try to find the best trading rules.

2. Context and technical trading rules

In the stock market, analyst / users need tools for doing their research, and take decisions to buy / sell shares. Overs the years, tons of technical trading rules have been created, with various effectiveness. The first one was certainly made by Alexander [5]:

“A trader need to buy if the price rises by a fixed percentage (5%, say) and sell if the price declines by the same percentage.”

This conclusion was also supported years after by Flama and Blume [5]. In a more recent study, they found that the best way to know which period to be in the market, is by calculating the volatility of the period [9]. Nowadays, these rules can still be applied, but without the same impact as before. Indeed, there are many more stock markets, with new inputs every second, the variability of them are too high.

Evolutionary algorithms are a good solution to find adapted trading rules, it is a good option to obtain results in a timely manner [4]. Human alone cannot handle the vast amount of data that is constantly produced and transmitted.

3. Stock market algorithms individuals

To generate meaningful technical trading rule parameters, we first need to create individuals with meaningful parameters, computed as a vector (equal to the chromosome).

3.1 Stock parameters

We are interested in 2 parameters: the prices of the stock, every X-times, and the rate of it over a chosen past period.

In the example (IGBM) [4], prices of the stock are taken every end of the day (daily closing price). The rate is taken over the 3 past months. The chosen data set cover the 2 January 1972 to 15 November 1997 period (4376 observations).

3.2 Moving average (MA)

In the market transactions, the main statistical datas are the MAs. It's an indicator that helps smooth out price action by filtering out the "noise" caused by random short-term price fluctuations [6]. It's an average of current and past prices over a chosen period of time. We will speak about short and long MA, where short and long refer to the period size.

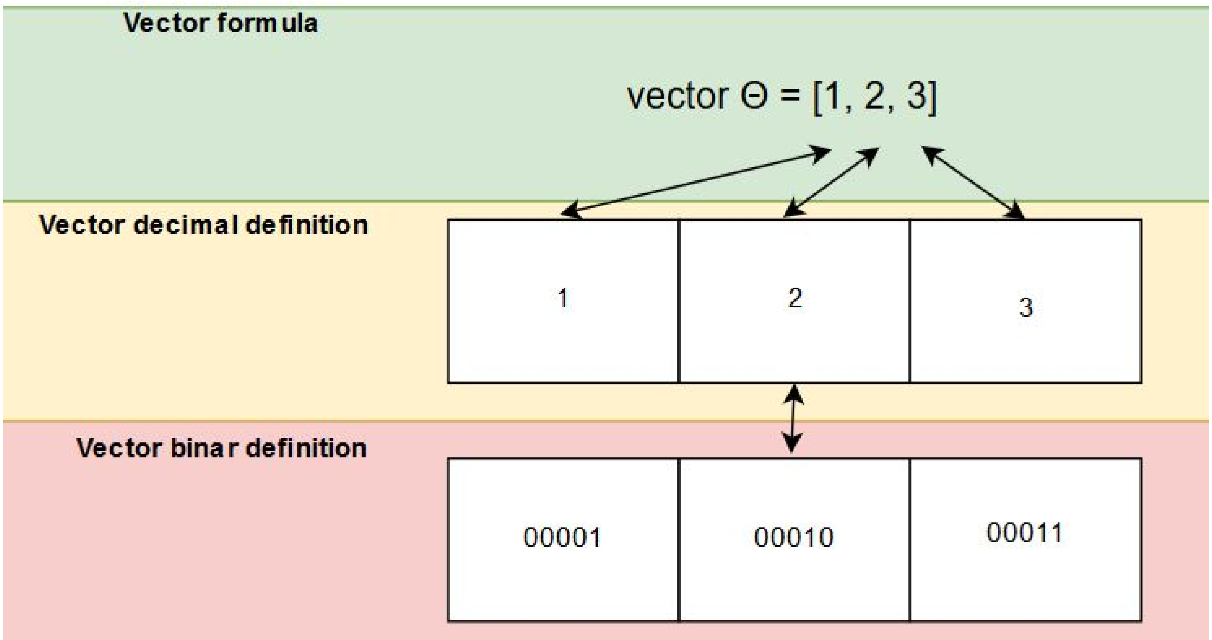
In our example "short" MA will be the past week and "long" MA will be in the 3 last months.

3.3 Stock parameters to vectors

These parameters need to be computed to chromosomes ([4] they speak about vectors), which are made of three parameters the short MA, long MA and the filter parameter that prevents premature buy and sell signals.

For example, in the vector Θ = [θ1,θ2,θ3], θ1 is the length of a short MA and θ2 is the length of the long MA, representing the number of days used to calculate the MAs, while θ3 is the filter parameter.

Layout of the vector creation:



4. Selection and fitness function

The fitness function determines how close an input is to the optimum solution of the desired problem. Here it rates how each vector are a good investment, depending on the in / out period, and the benefits with it. What result in the comparison over the risk rates of each vector, corresponding of the risk rate on each investment.

4.1 Selection method

The algorithm used in the example [6] is the Genitor selection (Whitley, 1989). It is a steady state selection method. Each time one individual is chosen according to linear ranking and then the worst individual in the population is replaced [8]. What result in rank all individuals according to performance. With this rank the algorithm will replace by copy the poorly performing individuals, by the most performing ones.

In addition to that, single point crossover using random breakpoint, and random mutations are applied over candidates surviving the selection process. This occurs with a very low probability in order not to destroy promising areas of search space [See part 5 for the parameters used in the example].

4.2 Fitness function, according to trading rules

The trading rule represented in the study is based on a market timing strategy, consisting of rating the security risk of each investment, and investing the total fund in the risk free stock markets.

The trading strategy specifies the position to take the following day of an investment, depending on the "buy" or "sell" signals.

The rule to determine these signals can be expressed in the formula [8]:

$$S(\Theta)_t = MA(\theta_1)_t - (1 + (1 - 2S_{t-1})\theta_3)\, MA(\theta_2)_t$$

Where Θ = [θ1,θ2,θ3] is the created vector. MA(θ) is the MA indicator, defined as :

$$MA_t(\theta) = \frac{1}{\theta}\sum_{i=0}^{\theta-1} P_{t-i}, \qquad t = \theta, \theta+1, \ldots, N$$

To rate the security of an investment over a period, each period is classified in two categories:

- "in": the investor is holding shares in the market. If the price of the market decrease in the future, $S(\Theta)_t$ will be "sell", shares are sold and this investment will earn a risk-free rate of return rf(t).

- “out”: if the price on the market increase in the future, $S(\Theta)_t$ will be “buy”, investment will be done. This investment will earn a risk-free rate of return rm(t)

Finally, the function to find the risk-free rate of an investment, used to find the best investment (corresponding to the fitness function) is:

$$r_{tr} = \sum_{t=1}^{N} S_{t-1} rm_t + \sum_{t=1}^{N} (1 - S_{t-1}) rf_t - T * c$$

where c denoting a transactional cost of 0.10% and T is the number of transactions.

5. Algorithm parameters

The settings that have been used in this particular algorithm result of another study [12], where they multiplied the experience in order to find the best parameters:

Crossover probability	0.6%
Mutation probability	0.005%
Maximum number of iteration	200
Initial population of chromosomes	200

6. Weak/ Strong points

Analytical and deterministic algorithms do take considerable time, due to the multitude of possible solutions, and because of the number of updates. We don't have this problem with an evolutionary algorithm, so it is one of the most valuable choices in finance algorithmic.

Another good point is the “randomness” (due to the mutation, crossover, etc.) of the algorithm. It adapts perfectly to the randomness on the stock market, with some fluctuation.

But this part of randomness in the algorithm result in a sort of “black box’’ [10]. We can’t really make prediction on what will happen during the iterations of the algorithms. And this result in a problem of optimization. If we don’t know where is the lack of precision in the algorithm, it is pretty hard to improve it.

7. Future research direction

Multiobjective evolutionary algorithms (MOEAs) have attracted a lot of research effort during the last 20 years, and they are still one of the hottest research areas in the field of evolutionary computation. MOEAs permit to optimize more than one objective function simultaneously, what is the main improvement we can do on the stock market.

<u>Reference:</u>

[1] Antonin Ponsich; Antonio Lopez Jaimes; Carlos A. Coello Coello. *Transactions on Evolutionary Computation.* Volume 17, Issue 3, pages 321-344. (Chapter name: *A Survey on Multiobjective Evolutionary Algorithms for the Solution of the Portfolio Optimization Problem and Other Finance and Economics Applications.*)

[2] Li Lin, Longbing Cao; Jiaqi Wang; Chengqi Zhang. T*he Applications of Genetic Algorithms in Stock Market Data Mining Optimisation*. Faculty of Information Technology, University of Technology, Sydney, NSW 2007, Australia Capital Market CRC, Sydney NSW 2000, Australia

[3] Ma. Guadalupe Castillo Tapia; Carlos A. Coello Coello. *Applications of multi-objective evolutionary algorithms in economics and finance: A survey*. Conference in Singapore, 25-28 Sept. 2007. Paper publisher: IEEE.

[4] Janko Strassburg; Christian Gonzalez-Martel; Vassil Aleksandrov. *Procedia Computer Science*. Volume 9, 2012, pages 1306 - 1313 (Chapter name: *Parallel genetic algorithms for stock market trading rules*)

[5] Eugene F. Fama and Marshall E. Blume. *Filter Rules and Stock-Market Trading. The Journal of Business.* Vol. 39, No. 1, Part 2: Supplement on Security Prices (Jan., 1966), pp. 226-241

[6] Franklin Allen; Risto Karjalainen. *Using genetic algorithms to find technical trading rules.* Journal of Financial Economics Volume 51, Issue 2, February 1999, Pages 245-271.

[7] S. W. Roberts. *Control Chart Tests Based on Geometric Moving Averages*. Pages 239-250. Publication date: 30 Apr 2012.

[8] Byoung-Tak ZHANG; Jung-Jib KIM. *Evolutionary Optimization .An International Journal on the Internet.* Volume 2, Number 1, 2000, p.55-70.

[9] William Brock; Josef Lakonishok; Blake LeBaron. *Simple Technical Trading Rules and the Stochastic Properties of Stock Returns.*

[10] Hassan Ismkhan. *Black box optimization using evolutionary algorithm with novel selection and replacement strategies based on similarity between solutions. Applied Soft Computing.* Volume 64, March 2018, Pages 260-271.

[11] Frank Schlottmann; Detlef Seese. *Swarm and Evolutionary Computation* December 2019, Volume 51.
[12] F. Ferńandez-Rodrıguez; C. Gonzalez-Martel; S. Sosvilla-Rivero. *Optimization of technical rules by genetic algorithms: evidence from the madrid stock market, Applied Financial Economics* 15 (11) (2005) 773–775.