Mining in-depth patterns in stock market

Li Lin* and Longbing Cao

Faculty of Information Technology, Sydney,
University of Technology,
Capital Market CRC,
Sydney, Australia
E-mail: linli@it.uts.edu.au
E-mail: lbcao@it.uts.edu.au
*Corresponding author

Abstract: Stock trading plays an important role for supporting profitable stock investment. In particular, more and more data mining-based technical trading rules have been developed and used in stock trading systems to assist investors with their smart trading decisions. However, many mined trading rules are of no interest to traders and brokers because they are discovered based on statistical significance without checking traders’ interestingness concerns. To this end, this paper proposes in-depth data mining technologies to overcome the disadvantages of current data mining methods. We implement a decision support in-depth trading pattern discovery system with Robust Genetic Algorithms (RGA). The system integrates expert knowledge and considers domain constraints into the trading rule development. We further utilise this technique to mine actionable stock-rule pairs targeting behaviour with high return at low risk. The proposed approaches are tested in real stock orderbook data with varying investment strategies.

Keywords: in-depth pattern; data mining; stock-rule pair; Robust Genetic Algorithm (RGA); technical trading rule; domain knowledge; constraint.


Biographical notes: Li Lin has been awarded PhD in Computer Science at University of Technology, Sydney in 2006. Currently, his research interest is stock data mining.

Longbing Cao’s research interest includes data mining and multi-agent technology.

1 Introduction

In stock (share) markets or any other similar financial markets, one important target is making profit. Since the profit is total earnings minus expenses, the key issue to reach this target is a trader (or an investor) to buy a stock at a low price and to sell it at a high price. However, sometimes, it is difficult to confirm whether a price is low or high. Technical trading rules are developed by some researchers to overcome these difficulties (Acar and Satchell, 1997; Allen and Karjalainen, 1999; Bessembinder and
Chan, 1997, 1998; Robert, 1999; Sullivan et al., 1999). A trading rule is a mathematical formula with some parameters, once the parameters are set properly (pattern), it can find the low and high price and give a recommendation to buy or sell (trading signal or signal in brief). Currently, many trading support platforms pay enough attention to incorporate trading rules to assist their users in searching for more profitable signals on the Australian Stock Exchange (ASX). However, there are still four limitations of current trading rules. Firstly, trading rules discarded the integration of expert experiences, business constraints and domain knowledge, which are necessary in real market trading. For example, a trading rule generated only five signals during one-year trading for getting the best Sharpe Ratio (SR) (Lin et al., 2004). This five-signal pattern is unreasonable and unprofitable in future trading. Secondly, the output of a trading rule was only one single value for each parameter. Sometimes, it was a noise and unreasonable. Thirdly, trading rules did not match and rank stock-rule pairs. If there is only one stock-rule pair, trading rules could help to generate the best signals, but, it could not help to choose a stock-rule pair with high profit. Finally, after we found the pairs, trading rules did not consider the relationship between various investments and the number of pairs. The different investments may have a different return for the same pairs, but, trading rules did not consider the investments when generating signals.

In this paper, we propose a computer-aided system extracting in-depth trading rules (Cao et al., 2005) based on Robust Genetic Algorithms (RGA). Domain driven in-depth pattern discovery methodology (Cao and Zhang, 2006a,b) and particular stock in-depth rule mining technologies (Lin, 2006), for instance, RGA, optimal subdomain and stock-rule pair are developed and integrated with our proposed system to improve the performance out-of-sample.

Firstly, special domain constraints are useful in real market applications, otherwise the result cannot be used in real markets. For example, it is unreasonable if the number of one-year trading signals is fewer than five. In this paper, we propose a RGA based on Genetic algorithm and the trading rules to integrate domain constraints. Since RGA considers efficiency and domain constraints, the result is more applicable in real stock markets. The result is the best one (more than 90% of enumerating result), but, the execution time is only 1% of enumerating algorithms (Lin et al., 2004).

Secondly, some traders have their own successful experience and they want to integrate their knowledge into a system to improve the trading rules performance, and the traders also want to microtune parameters to fit their own special knowledge. For example, length of in-sample data is more than 30 days. In our system, we can integrate expert knowledge. Meanwhile, we also output an optimised subdomain for traders. Traders can microtune in the subdomain in which almost every value can help them to make a positive profit because the subdomain is evaluated for ‘continuous optimal’. The subdomain is a small area, in which all values are near-optimal.

Thirdly, investments are various in stock markets. Traders and investors need to consider and trade different numbers of stocks for different investments in order to make the highest return. The relationship between investments and the number of pairs is non-linear because stock signals interact with each other. For example, if we invest less money, we have more chance to have not enough money in hand to buy a signal because there are more than one stock-rule pairs. We drew the graph about investments and the number of pairs and we find maximal points for different investments.

Finally, current traders and investors can select a stock from every market around the world through internet trading system. The problem is how to find the best stock and best
trading rule pairs. It is beyond technical trading rules. In this paper, we rank stock-rule pair to help the users (traders and investors) to choose the best one and experiments show the result is better than that of market return (see Figure 11).

This paper is organised as follows: in Section 2, technical trading rule concepts are introduced briefly. In Section 3, mining in-depth pattern technical methods and some applications are presented, such as, RGA, finding the relationship between the investment and number of stocks and ranking stock-rule pairs. Finally, we summarise the related work and conclusions in Sections 4 and 5, respectively.

2 Technical trading rules in stock market

Technical trading rules are defined by mathematical formulas, which are still widely used in many trading platforms and can be easily implemented by computer programs as well (Olson, 2004). We give a brief introduction and an example in this section.

2.1 Technical trading rules

The technical trading rules (in brief ‘rules’, also called trading strategy) are widely used in stock markets to help the traders to make smart trading decisions, such that they can make higher profit with lesser risk. Herein, an example is given: Moving Average (MA) cross-over rules.

MA cross-over rules are one of the most popular rules discussed in the technical analysis literature (Sullivan et al., 1999). The standard MA rule, which utilises the price line and the MA of price, generates signals in an uptrend, while long commitments are retained as long as the price trend remains above the MA. Thus, when the price trend reaches a top and turns downward, the downside penetration of the MA is regarded as a sell signal. Similarly, in a downtrend, short positions are held as long as the price trend remains below the MA. Thus, when the price trend reaches a bottom, and turns upward, the upside penetration of the MA is regarded as a buy signal.

In practice, there are numerous variations and modifications of the above rule. We examine several of these. For example, more than one MA can be used to generate trading signals. Buy and sell signals can be generated by cross-over of a slow MA (long run, L.R. in brief) by a fast MA (short run, S.R. in brief), where a slow MA is calculated over a greater number of days than the fast MA. The MA for a particular day is calculated as the arithmetic average of prices over the previous \( n \) days, including the current day. Thus, a fast MA has a smaller value of \( n \) than a slow MA. There are two types of ‘filters’ we will impose on the MA rules. The filters can assist in filtering out false trading alarms (i.e. those signals that would result in losses). The fixed percentage band filter requires that the buy or sell signal exceeds the MA by a fixed multiplicative amount. The time delay filter requires that the buy or sell signal remains valid for a prespecified number of days before action is taken. Note that only one filter will be imposed at a given time. Once again, we consider holding a given long or short position for a prespecified number of days (see Figure 1, CommSec,http://www.commsec.com.au).

In case short run average changes from lower to higher than long run average, then generating a ‘BUY’ alert; else, when short run average changes from higher to lower than long run average, then generating a ‘SELL’ alert.
The workflow rules are

1. the whole order book data is divided into two parts: in-sample set and out-of-sample set
2. in in-sample set, we try to find the best combination of parameters to make the highest return through training and keep the parameters in out-of-sample set to verify and
3. in out-of-sample set, we evaluate the result to see whether it is still the best or nearly the best one.

Because the parameter is derived from the in-sample data and test result is obtained from the out-of-sample data, it has the predictability in real stock market trading.

2.2 Data and evaluation metric

In stock market trading, most traders and investors want to get profit or higher return with lesser risk, (risk means the probability of losing some or all of original investments). The profit and return metrics do not consider risk, so, we undertake SR (Dowd, 2000) as the evaluation metrics (www.investopedia.com), which is defined by \( \frac{(R_p - R_f)}{\sigma_p} \), where \( R_p \) is the expected portfolio return, \( R_f \) is the risk free rate and \( \sigma_p \) is the portfolio standard deviation. When SR is higher, it means the higher return with lower risk.

In this paper, we test the following trading rules: Enhanced MA, Filter rules, Channel-break-out, Simple MA, Filtered MA and Support and Resistance, they are coded as Rule 1–6, respectively. Totally 27 stocks are randomly selected from ASX includes AGL, AMC, AMP, ANZ, AXA, BIL, BLD, CCL, CML, CSL, GPT, IAG, MAY, MBL, MGR, MIG, ORG, PBL, QAN, QBE, SGP, SUN, TEL, TLS, WES, WFT, WOW and WSF. They are numbered from Stock 1 to Stock 27, respectively. The data set for training and testing is from 20000101 (1 January 2000) to 20021231 (31 December 2002), we use intraday orderbook on-market data.
3 Mining in-depth patterns

Currently, there are many profitable rules used successfully in stock markets. However, there are still many problems that need to be solved, such as, to consider the domain knowledge, to optimise subdomain for trading rule parameters, to determine the number of trading stocks for different investment amounts and to rank the best stock-rule pairs.

In the following sections, the solutions of these problems are introduced respectively. Basically, we develop a kind of RGA and use it to mine actionable stock-rule pairs for different investment portfolio analysis.

3.1 Domain-knowledge integration

When we use traditional trading rule methods to get the best parameters, we need only to get the best SR for a selected rule and stock. However, sometimes, the best mathematical results are not actually workable in real stock market trading (see Figure 2). So, we incorporate some filters to remove the unreasonable results in order to get the reasonable and applicable results. These filters are domain knowledge, experience, domain constraints, etc.

Figure 2 The best SR corresponding to the signals. (SR is 28.25. Stock is WSF, rule is MA, in-sample data is one-month from 1 January 2000 to 31 January 2000 and out-of-sample data is one month continuously after in-sample data. The ordinate is price in Australian dollar, the horizon is time. The following figures are the same data set except mentioned differently.)

The determination of filters is better based on the domain-constraints and expert knowledge. In this system, we build a knowledge-database to store the domain-constraints and expert knowledge. For instance, trading signals should be between 10 and 100 annually; SR is between −5 and 5, otherwise it might be a noise. Further, a sell signal should not be generated in the same day of a buy signal. These constraints are related to domain knowledge and business requirements, which is through the cooperation between domain expert and mining system. Moreover, the knowledge base should accumulate by learning results and expert feedbacks automatically as well.
Based on the above strategies, the new filtered result is shown in Figure 3. Compared with the situation in Figure 2, this result is more applicable and reasonable than that without filters. The size of in-sample set and out-of-sample set are set by experience. It can be set from one month to one year (Robert, 1999).

**Figure 3**  The signal of robust result. The short and long term is reasonable and can get a better result in the out-of-sample data (Sharpe Ration is 0.685)

Another application of RGA is to find a subdomain of parameters. Since the best value of a parameter maybe a noise, it is better to give an optimal subdomain than an exact value.

We develop a new GA-based algorithm to find a near optimal subdomain rather than the best value. In this subdomain, almost every value can help traders to make a positive profit even if it is not the best one. Therefore, the decision support is more robust and actionable.

Figure 4 represents the randomly distributed SR of original GA-based trading rule results. In this figure, although we can find the best value, it is randomly distributed. In practice, this result is not flexible. If there is a little shift of the value, the profit could move to negative scope. However, after we optimise SR by outputting a small domain, the new results are all positive in this domain (see Figure 5). It is steady and less possibility to be a noise.

**Figure 4**  SR of all combination of the parameters. The best one in the circle. ('+' positive, 'o' zero, '' negative value)
Figure 5 The optimal subdomain, in which all value can make a positive profit prediction.

![Image of Figure 5](image1.png)

Figure 6 shows the human-machine interface in our system. The parameter values can be set an exact value or a subdomain before execution and can be microtuned after execution. The subdomain is optimised in which almost every combination is profitable because RGA is looking for a better ‘group’ instead of a better single value.

Figure 6 The subdomain parameters.

![Image of Figure 6](image2.png)

Furthermore, we can demonstrate the in-depth pattern mining in terms of finding links between stock and rule. In the following subsections, we introduce RGA to generate more reasonable signals for a rule and a stock pair and the relationship between investments and stock-rule pairs.
3.2 To determine investments

For different investments, returns should be different even if the pairs are the same. When the investment is small, we can get high return by considering a small number of stocks. When the investment is large, we need to trade more stocks to make high profit. However, the relationship is not clear and there is no solution for this problem in current trading rules. In this section, we give an in-depth pattern mining application to find the relationship between the investment and the number of pairs.

Users (traders and investors) can fix the amount they want to invest, such as AUD 1000 or AUD 10 million. We provide the users with the best return for the number of pairs traded. We also provide the users each stock-rule pair and alert signals.

Figure 7 shows that when the number of pairs is fewer, the return is higher. It means better pairs are found and traded firstly. When the number of pairs increases, the return decreases because many worse signals are triggered and many better signals are missed.

**Figure 7** The average return to the number of stock-rule pairs for different investment

In this paper, we presented an experimental proof of the relationship between investment amounts and number of stock-rule pairs (stocks).

The method presented in this section is as follows.

Firstly, we get the top pairs, which are ranked by SR – the higher SR, the higher return.

Secondly, we sort all trading signals by timing (see Figure 8). Figure 8 shows the result of the signals sorted by timing. All the signals are generated by our proposed system. It includes serial number, trading rule code, stock code, date, time, trade price, volume and signal type (buy/sell).

Thirdly, we implement trades as a real market. If a buy signal is generated and we have enough money in hand, we buy the current stock as much volume as possible under considering the order book volume and available money. If a sell signal is generated, we sell the current stock at the volume as large as possible.

Finally, we calculate return and SR and decide the best investment.

The following Figures 9 and 10 show the result. Figure 9 shows the return and Figure 10 shows the profit.
Mining in-depth patterns in stock market

Figure 8 The signals of the real-time best-pair trading alerts, when consider more stocks

![Figure 8](image)

Figure 9 Return of the different investment and the number of pairs

![Figure 9](image)

Figure 10 Profit of the different investment and the number of pairs

![Figure 10](image)

Figures 9 and 10 present the following information. When the investment is more, the profit is higher but the return becomes lesser. When the investment is less, the return is higher, but the profit is lesser. For different investments, a user needs to consider the number of stocks to be traded. Such as, for a bigger investment, the user needs to trade more stocks.
From Figures 9 and 10, we can see the result is consistent with real-stock markets. Further it represents the exact relationship between investments and the number of stocks.

Moreover, we identify that return decreases when investment increases. Profit increases when investment increases. From the graphs (Figures 9 and 10), we can find the best point for different investment.

Figure 11 and Table 1 show the market index return and the pair return. It shows that the total monthly return (1.511% when investment is AUD 1000 and 1.457% when investment is AUD 10000) of the pair is better than market average monthly return (0.228%). Seven monthly returns are better than market return. (Year 2001 and transaction is 0.25%).

**Figure 11**  The market index average return and the pair return

**Table 1**  The return comparison

<table>
<thead>
<tr>
<th>Annual return (%)</th>
<th>Average monthly return (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Index</td>
<td>2.747</td>
</tr>
<tr>
<td>A$1K</td>
<td>18.136</td>
</tr>
<tr>
<td>A$10K</td>
<td>17.7</td>
</tr>
</tbody>
</table>

### 3.3 To rank stock-rule pairs

For a stock and a rule, we can find a robust result by RGA as we present in the earlier section. In this section, we present ranked stock-rule pairs and activate trading signals as well.

Firstly, we build a two-dimensional stock-rule pair performance matrix table in terms of SR. Secondly, we sort this two-dimensional table into a one-dimensional list by SR. The list is the ranked stock-rule pairs by SR.

Simulated on these pairs, Figure 12 shows the average return per pair, which is total return divided by the number of pairs. It shows that the top pairs are better for making profit with less risk than the others. When the number of pairs increases, SR decreases that means the pairs are ranked properly.
Figure 12  The average monthly return to the top percent pairs. (The red trend line shows the average return is decreasing)

The results obtained from this method show that the ranked pairs are more profitable than the previous work without ranking and more profitable compared with market index return, which can be regarded as the average market return.

Figures 13 and 14 show the part of the ranked pairs both in-sample and out-of-sample by SR, respectively.

Figure 13  The ranked pairs in-sample set

Figure 14  The ranked pairs out-of-sample set
The rules are recommended by some finance experts (Sullivan et al., 1999) and we consider in-depth pattern data mining methods to improve the best parameters and rules. The results show that stock-pair methods provide higher profit with lesser execution time (Lin, 2006).

4 Related work

A number of researchers and research projects have already taken advantage of trading rules and optimisation techniques in data mining field (Sullivan et al., 1999). However, most of the projects concentrated only on trading rules. We focus on mining in-depth pattern methods based on the trading rules.

Acar and Satchell (1997) and Sullivan et al. (1999) summarised trading rules against profit. They also gave experiments that the trading rules can help traders to get profitable signals. Allen and Karjalainen (1999) performed GA to find trading rules. Neely et al. (1996) proved some evidence that the trading rules had predictability. Olsen (2004) tested whether MA trading rule profits declined over the period from 1971 to 2000. Some previous studies have reported mixed results regarding the success of technical trading rules in currency markets. However, Olsen’s optimised rules for successive 5-year in-sample periods from 1971 to 1995 and tested over subsequent 5-year out-of-sample periods. Results show that risk-adjusted trading rule profits have declined over time from an average of over 3% in the late 1970s and early 1980s to about zero in the 1990s. Thus, market inefficiencies reported in previous studies may have been only temporary inefficiencies. Leigh et al. (2002) implemented a recogniser for two variations of the "bull flag" technical charting heuristic and used that recogniser to discover trading rule on the NYSE (www.nyse.com) composite index out-of-sample results indicates that these rules are effective. All the above works are a little different from ours since our work is based on trading rules instead of seeking a new trading rule. Prinzie and Van den Poel (2005) have done the research about data mining models with practical constraints or thresholds. It can improve model performance as the model is optimised for the given implementation environment, if the implementation constraints/thresholds are known in advance. They illustrated the relevance of this constrained optimisation of data mining models on a direct-marketing case only. Lam and Lam (2000) presented a method to find trading signals that is similar to seeking a new trading rule.

5 Conclusions and future work

In this paper, we presented some methods and applications on mining in-depth patterns in stock markets based on technical trading rules. We applied RGA to get a result with domain constraints so that the result is more applicable and practicable. Moreover, we also offered a human–machine interface to integrate domain knowledge (expert experience and domain constraints) and traders can tune parameters by them. Moreover, we ranked stock-rule pairs with SR to find the best stocks and rules to form a pair. Finally, we drew the relationship graph between the number of stock-rule pairs and investments. All of these works were subjected for making a better performance (more profit or higher return, but less risk).
In future, we propose to overcome the following four problems.

Firstly, the problem is how to search the best stock-rule pairs. Currently, we build a stock-rule metrics and sort them by the performance. We can imagine that stocks can be divided into the same subset with same special patterns. In every subset, the stocks should be having same patterns and they can combine with the same trading rules.

Secondly, the problem to be considered is how to consider investments when ranking stock-rule pairs. Currently, when we rank the stock-rule pairs, we do not consider investments and we just consider a fixed investment, such as 1000 $. For different investments, the better pair maybe becomes a worse one. So the best way is when ranking pairs, to consider investments at the same time. The expected result is: when investment is 1000 $, we get a sorted stock-rule pair list; when investment changes to 1,000,000 $, we get another maybe different sorted stock-rule pair list.

Thirdly, the problem is to find a top best pair group. Currently, when we want to select the best top percentage pairs, the method is we sort all stock-rule pairs and select the top percentage pairs. The problem is the total return of a group is not equal to the sum of single pair’s return. Because signals of different pairs may influence each other and the volume and money in hand are different, too, so the integral result is not guaranteed to be the best even if each pair is the best one. For example, one pair generating a buy signal may make another pair’s buy signal unsuccessfully traded because there is not enough money in hand. So, our future work is to get the best combination pairs considering signals distribution and volume, but the member pairs may be not the best one when they are sorted pair by pair.

Finally, we use genetic algorithms to solve finance problems and make genetic algorithms more efficiently and effectively. We have implemented genetic algorithms and RGA into our system, but, we do not make any improvement for genetic algorithms (traditional genetic algorithm and added domain constraints). In future, we shall consider to upgrade genetic algorithms and make them more efficiently and effectively, for example, adding some new operations or changing the probability of crossover and mutation.

Acknowledgement

We would like to thank Prof. Chengqi Zhang and Dr. Shichao Zhang for their helpful discussion and support and Capital Market Cooperated Research Centre (CMCRC) (http://www.cmcrc.com), SIRCA (http://www.sirca.org.au) and AC3 (http://www.ac3.com) for funding this project and providing large realistic stock order book data for this research. All supports and feedbacks from Prof. Mike Aitken in the University of New South Wales, Prof. Alex Frino in the University of Sydney and others are much appreciated.

References


