

An Exploration of Simple Optimized Technical Trading Strategies

Ben G. Charoenwong*

Abstract

This paper studies the behavior and statistical properties of three simple trading strategies. Technical trading strategies can be viewed as a form of information gathering. But are they worth the computational cost? I compare the profitability and trading accuracy for three strategies with different information gathering techniques and parametric dimensions. The trading rules were a filter strategy, moving average strategy, and an arithmetic and harmonic mean difference strategy. Using an out of sample evaluation for both predictability and profitability as criteria, I find that added complexity does not translate into better performance.

1. Introduction

Technical analysis has been around nearly as long as the stock market. However, real study and widespread activity in the area began accruing around the period of extensive and fully disclosed financial information. The new availability of information allowed traders to look at more attributes of common stocks and other financial instruments, fostering the practice of fundamental analysis. Traders have tried to implement trading models using historical public information in hopes of finding patterns in the stock market movement. Moreover, major brokerage firms still publish technical commentary on the stock market and some individual securities compiled by “experts”. The continual existence of large technical analysis departments in large financial institutions is consistent with the belief that technical analysis is empirically useful. Moreover, there has been literature applying different technical trading rules in different countries’ stock markets¹. Results show that despite the variation in different stock markets, technical analysis manages to find excess returns consistently.

¹Isakov and Hollistein (1999) apply rules based on moving averages on Swiss stock prices, while Ratner and Leal (1999) study the variable length moving average for equities in 10 emerging countries in Latin America and Asia. Fernandez-Rodriguez, Martel, and Rivero (2000) use artificial neural networks in the Madrid stock market. Allen and Karjalainen (1993) use genetic algorithms to evolve basic building blocks

The advent of the efficient market hypothesis proposed by Fama (1965) was followed with a flurry of papers claiming that technical analysis is not profitable. Later, Samuelson (1965) and Fama (1970) stated that simulated trading results are in a sense a test of market efficiency. The hypothesis states that the price of stocks is a representation of all current information, so any movement cannot be predicted systematically. However, another group of studies related to this work show evidence of excess returns in strategies derived from past returns.

Research in trading strategies was popular from the 1960s and then again in early 2000s. Various papers found profitable trading strategies, attributing possible reasons to the non-linear semi-structured nature of the stock market, information asymmetry, and investor psychology. Brock, Lakonishok, and LeBaron (1992) claim that perhaps the excess returns over the buy and hold strategy to the simplistic and possibly inaccurate measure of volatility as the standard deviation of the return and lack of an accurate asset pricing model. In other words, if there were a better asset pricing model or measure of risk, the “excess” returns may disappear accordingly.

An investor, seeking to make a profit in the market, should consider between a random walk model and a more complex model a degree of dependence. Fama and Blume (1966) present the idea that in a random-walk market with or without a positive drift, no technical trading rule applied to a single security will consistently outperform the buy and hold strategy. Developing alternative models to the fair market hypothesis involves dedicating a fair amount of resources. Therefore, if the actual degree of dependence cannot be systematically optimized to generate excess returns over the buy and hold strategy, the investor should stick with the buy and hold policy.

of technical analysis into more complex algorithms applied to the S&P Composite Index.

*Charoenwong worked under the supervision of Professor Edward Rothman of the Statistics Department in the University of Michigan.

Technical trading strategies are algorithms that take inputs regarding the stock market, and outputs a decision, whether to buy or sell a stock for a given time period. Academic interest in testing technical analysis dates back to the 1960s. Early studies focused primarily on simple trading rules. There is an abundance of literature finding profitability in technical trading strategies using complex statistical tools and machine learning techniques. JS Liao and PY Chen (2001) develop a learning classifier system to adapt to changing market environments under the assumption that the stock market is semi-structured, non-linear and non-stationary. Potvin, Soriano and Vallee (2004) propose genetic programming as a means to automatically generate short term trading rules to exploit short term fluctuations in price, and O'Neill, Brabazon, Ryan and Collins (2001) introduce grammatical evolution as an improvement over works that used genetic algorithms. As more financial data becomes readily available, these techniques can be implemented to try to extract any meaningful information from the stock market. Though the machine learning techniques may not offer a theoretical explanation to the behavior of the stock market, the existence of systematic profits or losses may point out interesting patterns to be explored in financial theory. The techniques for discovering possibly hidden relationships in stock returns range from extremely simple to quite elaborate.

Another perspective is that technical trading strategies could also be considered as information gathering. Grossman and Stiglitz (1980) suggested that the traditional interpretation of market efficiency provided by Fama (1965) is flawed. If prices fully reflected information in the market, then investors who expend resources to gather information should be making a loss exactly equal to the cost of gathering the information. However, if nobody gathers costly information, then it cannot be reflected in prices. Therefore, there must be an award of sorts for expending the resources in the first place. Since the cost of information gathering is not

accounted for model of fair returns, there will seem to be excess returns. If the cost of acquiring the additional information, whatever form it may be in, is accounted for, then the excess returns should disappear. The excess returns should be equal to the cost of acquiring information through technical analysis. In this view, the excess returns first shown by Brock, Lakonishok and LeBaron (1992) and later on by many others are consistent with market efficiency. However, if this claim were true, as data become cheaper to acquire, store and distribute and computers become more powerful, the cost of obtaining technical information should decrease. Since the cost of acquiring information decreases, the excess returns should also decrease. This study does not pay particular attention to this hypothesis. Though not rigorously tested in this study, an expected trend should emerge.

If excess returns persist through time despite the availability of data, it may be more likely that other factors are accountable for the apparent inefficiency of the stock market. A trading strategy that produces a consistent profit (or loss) may contain predictive power. The strategies are optimized for profits initially through both the Newton-Rhapson algorithm using numerical approximations to the gradient and hessian, and the one dimensional algorithm native to the statistical program R. All strategies are in comparison to the buy and hold strategy dictated by the efficient market hypothesis. The strategies that were tested are a modified filter strategy, a moving average strategy, and a comparison of arithmetic and harmonic means for prices. The first two are momentum based strategies and work based on positive correlation between the stock price and its first lag.

Using daily data allows for more variation in the stock price. If there are more fluctuations in the data, there are more potential optimal times to buy and sell stocks. Though the stock market may have shown a persistent long term growth trend, in the short term the price

behavior of stocks is very noisy. Therefore, active strategies should be more profitable in the short term with more variation than the long term, since there is more possibility that the stock may be ‘mispriced’ according to the criteria for each strategy.

The technical trading strategies used in this study are both a combination of filter and trend based. Filter strategies indicate a buy and sell when the price falls above or below a specific percentage of a combination of past prices. An example of this strategy would be to buy a stock if it has increased by 3% or more in the past day. Trend based produce a buy and sell signal as a result of the cross of current prices and past prices. An example of this strategy would be to sell a stock if it has dropped below the 3 day low and moving average.

The paper will discuss the data used in the study and then go over the methodology. After that, all of the strategies used are presented in their entirety, from their development to whether the strategies remain in use today and why. The strategies implemented in this study also allow the plausibility of small investors to use technical trading strategies for profit.

2. Data

Technical trading strategies can also be applied to any type of financial instrument. Due to the theoretical obscurities financial derivatives, this study only focuses on equities. Because of the complex supply and demand dynamics of different industries, this study narrows down on the S&P 500 Total Return Index. Also, the profitability of technical trading strategies in an index representative of the stock market are more readily interpreted in a macroeconomic condition.

Using a representation of the entire stock market does not subject the time series to a directional drift that may be present in an index segmented by market capitalization. The stock index attempts to create a representation of the entire stock market. A committee selects the stocks to be included, though it is not through a strict rules-based decision like the Russell 1000.

Moreover, only stocks of publicly traded companies and those with sufficient liquidity are included in the index. The S&P 500 Total Return Index also accounts for dividends paid out by the different companies held in the index.

Because dividends can be considered as a kind of returns on top of capital gains, the study generates strategies based on the adjusted close price of the S&P 500. The ex-dividend days' prices are adjusted by adding back the dividend. This helps to ensure that the price series does not drop periodically simply due to the existence of the dividends and generate misleading results. If the dividends were not factored in, then there would be a periodic drop in the price (theoretically the drop is exactly the amount of the dividend). Technical trading rules may pick up on this trend and attempt to generate profits by buy on ante-dividend days. In reality, there was no real shift in the value of the stocks since they were simply discounted by the dividend. Trading rules that act on this false signal would generate expected negative returns exactly equal to the trading cost.

The interday strategies are optimized for the S&P 500 index using the close of every trading day since January 3, 1950 to September 30, 2011 while for the Vanguard 500 the data ranged from March 27, 1987 to September 30, 2011. The dates included in this study are all the historical data readily available on Yahoo! Finance. The data contains multiple shocks and recession periods that should provide a large enough sample to generate robust strategies that produce long term excess profits as opposed to short term profits that do not necessarily exploit any possible trends in the stock prices.

3. Technical Trading Rules

3.1 Filter Strategies

Filter strategies are a set of straightforward rules based on price momentum that decide whether to buy or sell a stock after it has risen or dropped a certain percentage. An x percent filter strategy is defined as follows: If the percentage changes of price from time $t-1$

Condition	Decision
$\left(1 - \frac{P_t}{P_{t-1}}\right) > \frac{x}{100}$	Buy/Hold
$\left(1 - \frac{P_t}{P_{t-1}}\right) < \frac{-x}{100}$	Sell/Stay Out

to t is greater than x percent, buy and hold the security until it drops at least x percent. Typically, these benchmarks are simply the previous day's closing prices. Alexander (1961) formulated the filter strategy to test the hypothesis that the stock market adjusts gradually to new information. Alexander studied filters ranging from 5 percent to 50 percent for the periods 1897 to 1959 involving two indices: the Dow-Jones Industrials from 1897 to 1929 and the Standard and Poor's Industrials from 1929 to 1959. He found profits significantly greater than the simple buy and hold strategy. Extending his study, Fama and Blume (1966) studied filter strategies ranging from 0.5% to 50% in the Dow-Jones Industrial Average from different initial dates centering around the end of 1957 to September 26, 1962.

The belief is that there is a specific value that would consistently generate excess profit. This is equivalent to claiming that if the stock market rises x percent, it should raise by more than x percent until it decreases by x percent. The underlying notion of buying when the price increases by x percent is that there is a lag in investors' reactions to new information. However, because of the reasons provided in the motivation of this study, as information disseminates quickly, we should see the excess profits from this strategy decrease. It is important to point out

the both Alexander and Fama and Blume studied individual stocks rather than an index.

Theoretically, the variance of the stock index should be less than that of individual assets, so we may see proportionally less trades and activity. Fama and Blume find that even though some filter rules find positive profits above the buy and hold strategy, after factoring in commissions and transaction costs that even the floor trader cannot avoid, the overall strategy is inferior to the buy and hold strategy.

Indeed though an interesting exploration in the statistical properties of stock prices and an attempt to outperform the market and gather information from past prices, the filter rule has been unequivocally rejected as a strategy that provides consistent excess returns.

3.2 Moving Average Indicators

Moving averages are a series of partial mean of prices P over the previous k days and is a measure of stock price momentum. It can also be used as a means to smooth out price and volume fluctuations accordingly. The moving average at time t for k days is computed as

$MA_t^k = \frac{1}{k} \sum_{i=t-k}^t P_i$. As a trading strategy, if the indicator for an upward momentum is

triggered, the strategy would suggest a buy. Upward

momentum at time t is defined as a short-term average over s days crossing a longer term average of l days upwards. Typical numbers for the short term average is from 5 to 15 days, while

Condition	Decision
$MA_t^s > MA_t^l$	Buy/Hold
$MA_t^s < MA_t^l$	Sell/Stay Out

the longer term averages can range from 50 to 90 days. The strategy will hold onto the stocks until a downward momentum is signaled. Downward momentum is confirmed when the short-term average crosses below the long term average. Using the long term as a benchmark of long term growth, we buy and sell depending on the short term fluctuations in the short-term moving average.

The idea behind this is that if there is a hint of an upward momentum, investors should buy and hold the stock, believing that the short term growth is greater than long term growth for the time being, and selling when they are equal again.

The moving average strategy implemented here requires two parameters, one for the short term average, and one for the long term average. To increase generality of the strategy, the study allows both parameters to fluctuate freely with only a lower bound of 2 days and an upper bound of 252 days (the number of trading days in a calendar year).

Gunasekarage and Power (2001) study the effectiveness of moving average strategies in emerging markets in South Asia, paying attention on the implications of possible excess profits against the weak form of the fair market hypothesis. They reject the null hypothesis that the returns earned from studying the moving average values are equal to that from the buy and hold strategy and conclude that the employment of the techniques generate excess returns. However, the literature fails to take into account trading costs. Since computing moving averages may be interpreted as a kind of information gathering,

3.3 Harmonic and Arithmetic Mean Indicators

The motivation behind the mean difference strategy is based on the idea of average prices. An investor looking to buy a share starts with cash and converts them into stocks, while an investor looking to sell a share starts with stocks and converts them into cash. In considering the average price in a transaction, an investor looking to buy stocks should consider the simple arithmetic mean as the average share price. However, the investor looking to sell a share should consider the harmonic mean. In this zero-sum set up with no transaction costs, any profits for an agent must come from a loss in the counter party.

The arithmetic mean (AM) over k days is simply the moving average over k days. The harmonic mean (HM) is computed as

$$HM_t^k = \frac{1}{\frac{1}{k} \sum_{i=t-k}^t \frac{1}{P_i}}, \text{ in other words, it is}$$

Condition	Decision
$100 * \left(1 - \frac{HM_t^k}{MA_t^k}\right) > \frac{x}{100}$	Buy/Hold
$100 * \left(1 - \frac{HM_t^k}{MA_t^k}\right) < \frac{x}{100}$	Sell/Stay Out

the reciprocal of the mean of the reciprocals over k days. This strategy triggers a buy signal when the percentage difference between the arithmetic mean and harmonic mean reaches a certain threshold. However, because generally this difference is small, without loss of generality, the percentage difference is multiplied by 100 arbitrarily.

4. Methodology

The S&P 500 adjusted price data was downloaded from Yahoo! Finance. The stock prices are corrected for dividends to simplify the optimization process. The trading profits are compared against the perfect decision, derived using the ex post returns to recursively generate a matrix of correct trading decisions as a function of trading cost.

Profitability of trading strategies in back testing have an appalling number of local optima. In order to avoid get around this issue, the optimization was iterated with random initial starting values. All optimization is done numerically through R, exploiting the development and advancement of numerical optimization methods. The main concern for this optimization process is getting stuck in local optima. Because intuition for the space of stock prices is limited, it is conservative to assume that the profit function for each strategy is not convex. As a counter measure, the intuitive and reasonable solution would be to conduct a grid search in the p -dimensional space of the domain for each function. Since the filter and simple average strategies

is a function with a one dimensional domain, they are readily optimized through a grid search with boundary conditions.

However, because the arithmetic-harmonic mean difference strategy takes in two parameters, the grid search would be 2 dimensional. Also, since the harmonic and arithmetic means would both have to be calculated for different day parameters, it is computationally taxing. It would be beneficial to find a faster optimization process. Therefore, the study conducts the optimization using both the one dimensional process and the Nelder-Mead method repeated 20 times with random initial starting points for the arithmetic-harmonic mean different strategy. In optimizing this strategy, the number of days to average over was set, between 1 and 100, and optimal percentages were obtained.

The main algorithm used was the Nelder-Mead method, primarily due to its ease of use in the programming language. It is a kind of heuristic search method for twice differentiable functions. Since the profit function and its derivatives, as of current knowledge, cannot be defined in a closed form (due to autocorrelations and non-linearity), the derivatives are numerically approximated. The algorithm is effective for unimodal problems. The method is more effective than a simple grid search since it uses more information from the fitness function by approximating both the first and second derivatives. This is the native method in the optimization function in R. Though the algorithm used can be extrapolated to dimensions greater than 2, the maximum number of parameters in the trading strategies that were used in this study was 2.

5. Empirical Results

From Table 1 we see that maximizing profits was in line with maximizing accuracy. This suggests that the optimized strategies (even if they are only local and not global optima) are not overfitted to the data. It is reassuring in an intuitive sense to see that in order to make the most money, a strategy would have to make a correct through the stock position as opposed to simply getting the position right when it matters. This is consistent with the idea that in the long run, a profitable strategy is a strategy that predicts the market movement most accurately. This study is considered a long term study since the data ranges from 1950 to 2011.

Moreover, it seems that though all the strategies made positive annualized profits in relation to the buy and hold strategy, the simple filter strategy performed best. No claims can be made about robustness and profitability of simple trading strategies and their complexity. The filter strategy used the least information, only looking at the percentage change in the day to day stock price, while the arithmetic and harmonic mean difference strategy used the most information, having to compute the moving strategy in real time. All the computation complexity comes from the numerical maximization of each trading strategy.

Table 1: Results of Optimized Strategies

	No Trading Cost			\$8 Trading Cost		
	Profit (\$)	Annual Return	Accuracy (%)	Profit (\$)	Annual Return	Accuracy (%)
Ideal	49337.82	6956.6910%	100.0000%	30617.38	4317.0870%	100.0000%
Filter	49337.78	6956.6850%	99.6846%	32739.26	4616.2740%	44.9672%
Moving Average	8473.00	1194.8440%	55.7729%	2249.96	317.2470%	33.0995%
AM-HM Diff	1386.22	195.4587%	53.0635%	1114.72	157.1768%	6.8348%
Buy and Hold	1114.76	157.1825%	53.6105%	1114.76	157.1825%	5.9660%

Table 2: Out of Sample Results

	Training			Test		
	Profit (\$)	Annual Return	Accuracy (%)	Profit (\$)	Annual Return	Accuracy (%)
Ideal	4313.66	760.3134%	100.0000%	26303.72	236.6784%	100.0000%
Filter	4058.58	715.3537%	35.2615%	12824.12	115.3902%	45.1914%
Moving Average	1210.28	213.3205%	24.5535%	-467.96	-4.2107%	64.8762%
AM-HM Diff	1288.08	227.0333%	1.6251%	-198.48	-1.7859%	23.7697%
Benchmark	1288.1	227.0368%	1.5527%	-173.34	-1.5597%	23.6089%

The optimization process was computed twice for an environment with no trading costs, and an environment with an \$8 fee per transaction in table 1. Moreover, we evaluate the trading strategies out of sample. The results are in table 2. Using a simple heuristic, the initial 80% of the data was used as the training period, with the last 20% of the data being the test period. The training period consisted of 12430 days, while the test period was the latter 3108 days. We see that though all the strategies made large profits in the training period and overall periods, this may be due to overfitting. When tested out of sample, all the strategies except the filter strategy made a loss. However, an interesting phenomena is that the moving average trading strategy actually had a higher sign change accuracy than the filter strategy, which suggests that though most of the time the moving average advises in the right direction, when it doesn't the losses are greater. Another key point to realize is that the testing period also included the Financial Crisis of 2008-2009. This would may be claimed as unusual phenomena, and therefore will shift the results of possibly all the strategies studied.

The filter strategy implies an autoregressive lag order of 1 for the stock price. It is a well known fact that stock prices move very much like a random walk, and in general fails unit root tests. Even if the price is a stationary time series process, it has long memory. The other strategies imply that stock prices have an autoregressive lag order of more than one. However,

since we see that the out of sample performance of all the other trading strategies were bad, we may get the idea that the strategies attempt to use a kind of information from the historical price that is false information. That is, the stock price behavior may not rely on lags greater than 1, while the other strategies try to use past prices to predict the future and end up doing worse than the simple filter strategy. This idea is consistent with the stock price having the Markov property. That is, perhaps future prices are only reliant on current prices, and not past prices. Though we do not accept this proposal, we fail to reject it explicitly using the data at hand and the particular strategies included in the study.

Also, the strategies were optimized against changing trading costs. This is to check whether the strategies are highly sensitive to trading costs. Though all the strategies lost accuracy and returns, the filter strategy still performed best. Interestingly, the arithmetic and harmonic mean difference strategy was the most sensitive to trading cost.

Properties of each strategy are explored individually.

5.1 Filter Strategy

It is imperative that this strategy takes into account trading costs. Notice that if trading costs were not taken into account as $x \rightarrow 0$, the condition for buying would be

$$\left(1 - \frac{P_t}{P_{t-1}}\right) > \frac{x}{100} \rightarrow P_t - P_{t-1} > 0. \text{ That is, the strategy would trigger a buy signal}$$

whenever the price of the stock is greater than the previous period, and the conditions for selling would be $\left(1 - \frac{P_t}{P_{t-1}}\right) < \frac{x}{100} \rightarrow P_t - P_{t-1} < 0$, whenever the price of the stock at time t is lesser than the price at time $t-1$. This strategy will yield all the correct signals, and represents the case where the stock's holdings match the ideal holdings with trading costs equal to 0.

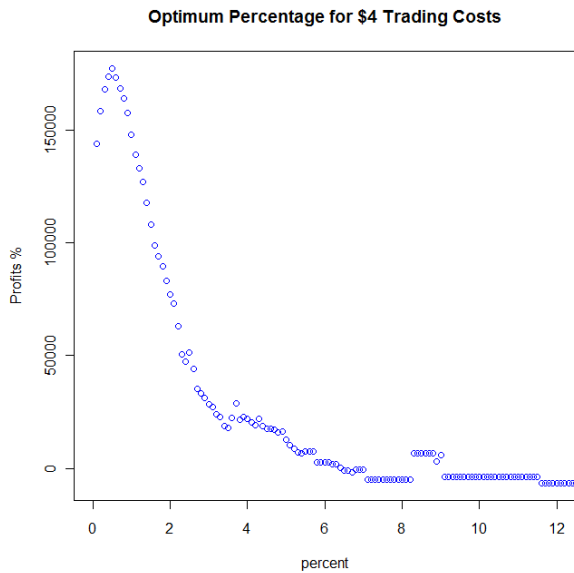


Figure 1.

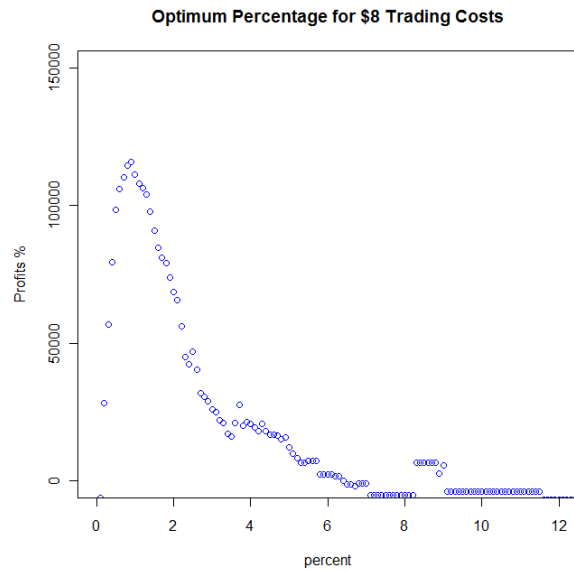


Figure 2.

Figure 1 and Figure 2 show that the returns percentage decreased dramatically from the maximum of over 150,000% to a little less than 125,000% when the trading cost doubled from \$4 to \$8. This strategy is sensitive to transaction costs, though the optimal profit seems to be achieved in the lower range of the percentage parameter. Another figure shows the sensitivity of the cumulative percentage returns as a function of trading cost. For when the trading cost was \$4, the optimal percentage was 0.5%, while when the trading cost was \$8, the optimal percentage was 0.9%. Since the trading cost penalizes the number of trades, we expect that as the trading cost increases, the optimal percentage will also increase. In other words, the strategy would pick the trades that tend to have more profitability.

There is an exponential decrease in the profit percentage as a function of trading cost. This can also be derived if we take into account trading cost in the buy and sell conditions, and solve as a function of trading cost.

The price of trading through a stock broker has decreased over the years. What was once attributable to the transaction cost may not be anymore. Companies such as Scottrade.com offer online trades for \$7, and Zecco has a cost as low as \$4.95 per trade. These companies were not available in the past, which may be why considering only a \$4 transaction cost throughout the

whole time period of available data generates a large profit. Therefore, this analysis of using \$4 and \$8 as benchmark trading costs is not valid for drawing general conclusions about the profitability of filter strategies. A more realistic approach would be to have decreasing transaction costs throughout time according to the market prices of stock brokerages. Though an interesting exploration of filter strategies on past stock price data, this strategy now does not allow for the generalization for or against the hypothesis that the amount of profit generated from this strategy is exactly equal that of the transaction cost.

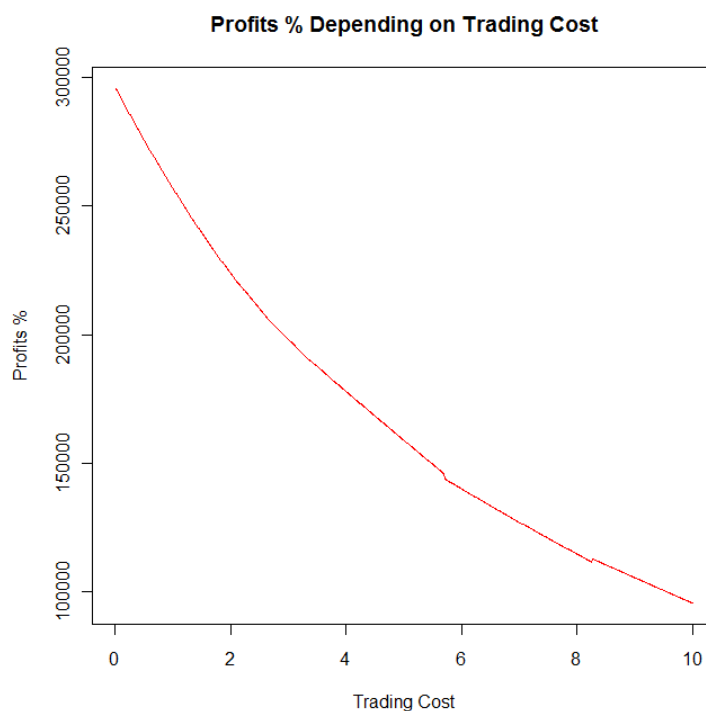


Figure 3.

However, if this were a kind of information gathering, then theoretically the excess profits should exactly equals the cost of computation and information retrieval. This conclusion cannot be reached within the scope of this study.

5.2. Simple Moving Average

The simple moving average strategy also generates excess returns, even given trading costs of \$4 and \$8 as benchmarks. Taking a look at the space of the 2-dimensional optimized parameters for short and long days, we see that there is no obvious relationship between these two parameters. However, we see that in general the number of days for which to evaluate the short term moving average is small. Also, getting optimal short parameters of around 35 suggest that perhaps apart from short and long term trends, there may be a middle term trend that could also possibly differ from the long term trend.

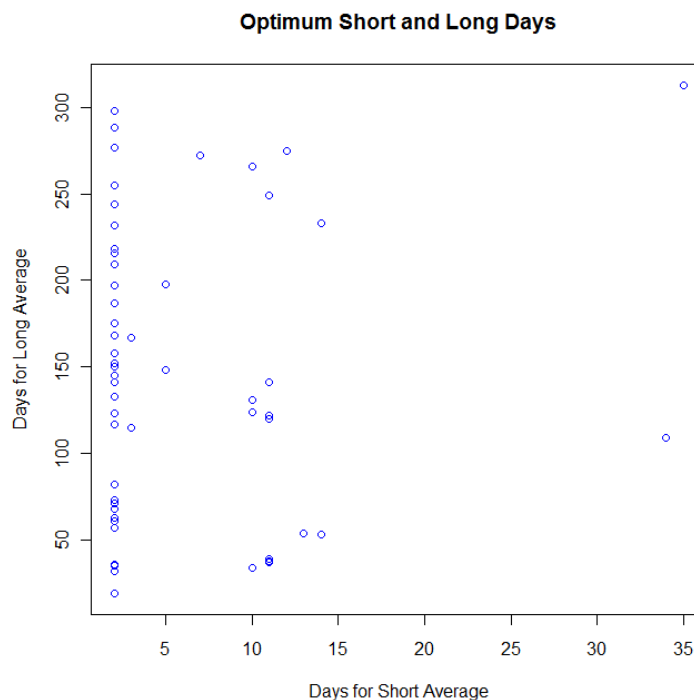


Figure 4.

Figure 4. above shows a scatterplot of the optimized strategy parameters. Notice that there is no clear relationship between the optimal parameters. Looking at each of the parameters in isolation, it seems that in general there is a decreasing trend for both parameters against percentage profits. Apart from the high leverage points in the plot of percentage profits versus

the short average parameter, in general the optimum parameter is less than 15 days, of which the values that generate the most profit are less than 5.

Another figure shows the percentage profits as a function of both short and long day parameters.

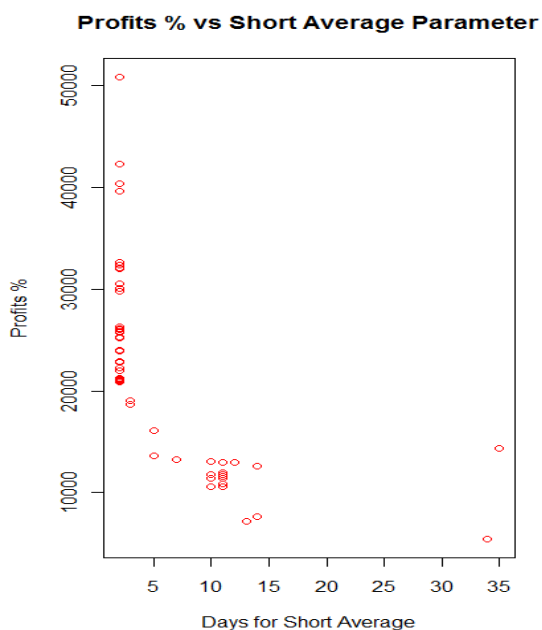


Figure 5.

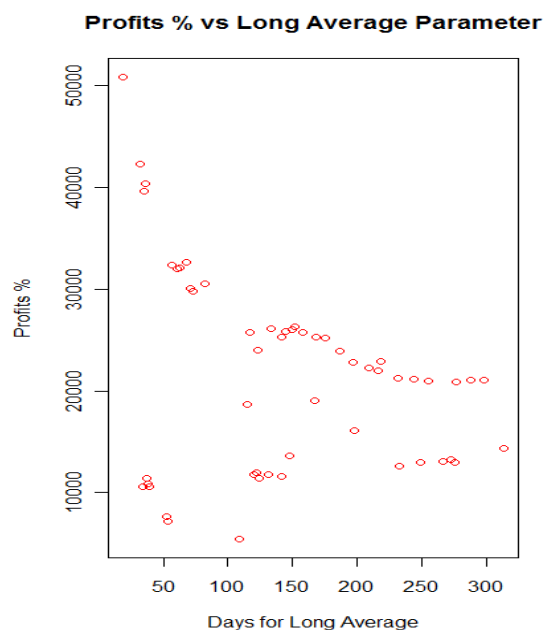


Figure 6.

5.3 Arithmetic-Harmonic Mean Difference Strategy

The arithmetic-harmonic mean different strategy generates positive returns. To get an intuition on the optimized parameters, we look at the optimal percentage difference as a function of the number of days to average over. However,

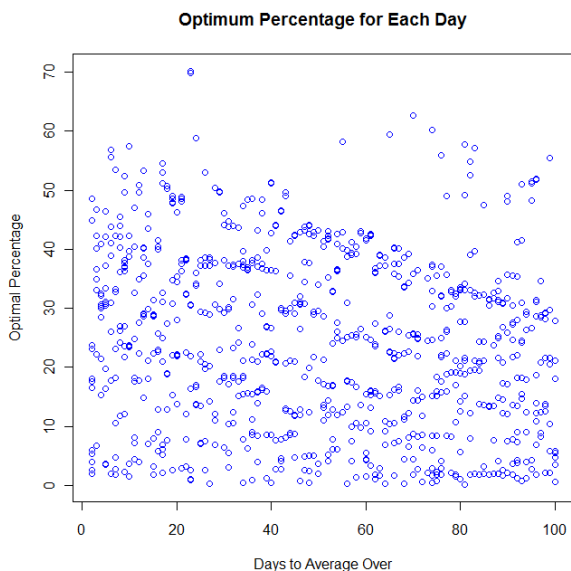


Figure 7.

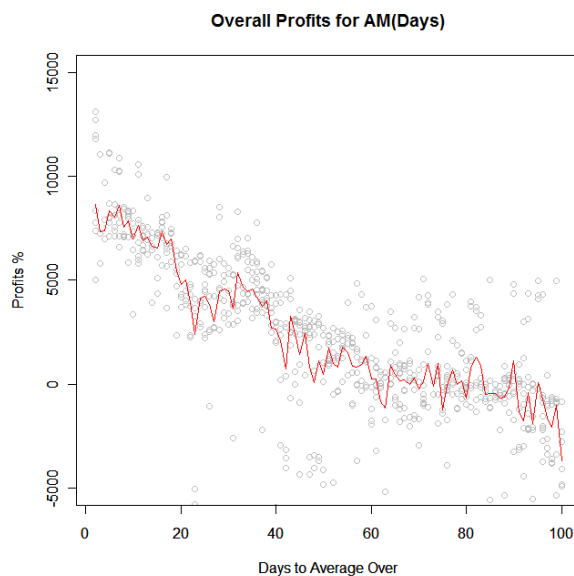


Figure 8.

The figure above reaffirms the idea of reiterating the optimization process for different starting points of days adds to the robustness of following findings. The optimized percentages show heteroskedasticity, their variance decrease as number of days to average over increases. This finding is consistent with the idea that the longer term averages are more stable. The otherwise random nature of this graph is consistent with the idea that the Nelder-Mead optimization converged to a multitude of local maxima. Because each point was run 20 times, we can study the overall profits as a function of the number of days to average over.

The computed percentage profits also seem quite noisy. In order to ‘smooth’ out the profits, we can compute the mean profits over each day and find the percentage return. Overall, profit seems to have a negative relationship with the number of days used to evaluate the arithmetic and harmonic means. This finding is consistent with the idea that since the arithmetic-harmonic mean is also one kind of momentum strategy, it is more applicable in the short term than in the long term. The exploitation of the relative mispricing of stocks seems to not last very long despite optimizing the percentage threshold over the different days. Though the existence of

positive profits contradicts the fair market hypothesis, the evanescence of these profits is consistent with the idea that as time goes on, the inefficiencies of the stock market die out.

Though the strategy offers insight on the behavior of the stock market, it does not seem feasible for small investors to implement. Investors with access to powerful computers may exploit this multiple-optimization process for short term evaluations of this strategy. However, since the optimization process is computationally intensive, the evaluation of the optimal percentage may not be in time to exploit this particular inefficiency. Since this strategy was not evaluated using out of sample performance tests, there is no claim to the robustness of any particular optimal parameter.

6. Discussion

6.1 Liquidity

An important assumption in the market is that the moment the strategy triggers a buy signal; we can buy for that day's exact closing price and sell at another day's exact closing price. This assumption is not feasible; since buyers tend to buy at a price slightly higher than the previous day's close, and sellers tend to sell at a price a little lower than the trigger price.

6.2 Breadth and Methods

This study shows only a brief exploration of some of some simple technical trading strategies. The strategies considered were: filter strategies, simple moving averages, and an arithmetic-harmonic mean difference strategy. As machine and statistical learning progresses, studying other more complex technical trading strategies may also show some insight on the stock market behavior, all while attempting to maximize profits. It also focuses on the practicality of the strategies and optimizes profits given different trading costs. Another fitness function that could be used is the forecastability of the strategies on making the correct trading

decision. Other applicable fitness functions may be generated by looking at some theoretical properties of stocks depending on a set of known variables. After describing the stock behavior using techniques such as a partial differential equation or a correlation study, a sort of criteria can be created using more parameters. The criteria can also then penalize or reward for the number of trades depending on whether the investor is active or passive, while also taking into account trading costs.

Other possible explorations of optimization techniques include genetic algorithms (Allen and Karjalainen 1993). Genetic algorithms allow for the cross-over and evolution of more complex trading strategies based on simple building blocks, much like the development and evolution of organisms through the building blocks of DNA. It simulates a population, then generates a second generation by mating patterns based on the parents. Random disruptions to the population can also be introduced, and the end result may well be a seemingly unintuitive but profitable trading strategy. Another viable candidate for generating a trading strategy is to use random forests (Breiman 2001, Ho 1995). The idea of random forests is an ensemble classifier that consists of decision trees based on different criteria. The final decision is then the mode of all the decision trees. A third possible method to generate trading strategies is to use neural networks (Fernandez-Rodriguez, Martel, and Rivero 2000). The main reason other optimization processes and strategies were not included in this study was due to both theoretical complexity and the implementation difficulty involved with coding such algorithms.

6.3 Additional Diagnostics

To gain even more insight on the stock market behavior and to evaluate explicitly evaluate the existence of excess profits against the baseline models, bootstrapping may be an appropriate method. The null models could be the simple random walk, a random walk with

drift, or a constant mean generalized autoregressive conditional heteroskedasticity model (GARCH).

References

- Alexander, S. (1961). "Price movements in speculative markets: Trends or random walks." *Industrial Management Review*. Vol 2, No. 2, pp. 7–26.
- Allen and Karjalainen (1993). "Using Genetic Algorithms to Find Technical Trading Rules." Rodney L. White Center for Financial Research. Wharton School of Business publications.
- Breiman, (2001). "Random Forests". *Machine Learning*. Vol 45, No. 1, pp 5-32.
- Brock, Lakonishok, and LeBaron (1992). "Simple technical trading rules and the stochastic properties of stock returns." *Journal of Finance*. Vol. 47, No. 5, pp 1731-1764.
- Fama (1970). "Efficient capital markets: A review of theory and empirical work." *Journal of Finance*. Vol 25, No. 2, pp 383-417.
- Fama, E., Blume, M. (1966). "Filter Rules and Stock-Market Trading." *The Journal of Business*. Vol. 39, No. 1, pp 226-241.
- Fernandez-Rodríguez, Martel, and Rivero (2000). "On the Profitability of Technical Trading Rules Based on Artificial Neural Networks: Evidence from the Madrid stock market." *Economic Letters*. Vol 69, No. 1, pp 89-94.
- Grossman and Stiglitz (1980). "On the Impossibility of Informationally Efficient Markets." *The American Economic Review*. Vol. 70, No. 3, pp 393-408.
- Gunasekarage and Power (2001). "The profitability of moving average trading rules in South Asian stock markets" *Emerging Markets Review*. Vol 2, No. 1, pp 17-33.
- Ho, T.K. (1995). "Random Decision Forests." *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*. Vol. 1, pp 278-282.
- Isakov and Hollistein (1999). "Application of simple technical trading rules to Swiss stock prices: Is it profitable?" *Social Science Research Network. Working Papers*.
- JS Liao and PY Chen (2001). "Dynamic trading strategy learning model using learning classifier systems" *Evolutionary Computing: Proceedings of the 2011 Congress on Evolutionary Computation*. Vol. 2, pp 783-789.
- Lukac and Brorsen (1989). "The Usefulness of Historical Data in Selecting Parameters for technical trading systems." *Journal of Futures Markets*. Vol 9, No. 1, pp 55-65.
- Nelder, J., Mead, R. (1965). "A Simplex Method for Function Minimization." *The Computer Journal*. Vol 7, No. 4, pp 308-313.

O'Neill, Brabazon, Ryan and Collins (2001). "Evolving Market Index Trading Rules Using Grammatical Evolution." *Applications of Evolutionary Computing*. Vol 2037/2001, pp 343-352.

Potvin, Soriano and Vallee (2004). "Generating trading rules on the stock markets with genetic programming" *Computers & Operations Research*. Vol 31, No. 7, pp 1033-1047.

Ratner and Leal (1999). "Tests of technical trading strategies in the emerging equity markets of Latin America and Asia" *Journal of Banking & Finance*. Vol 23, No. 12, pp 1887-1905.

Samuelson, P. (1965). "Proof that Properly Anticipated Prices Fluctuate Randomly." *Industrial Management Review*. Vol 6, No. 2, pp 41-49.

Appendix 1: R Code

```
#####
# Ben Charoenwong
# Technical Trading Strategies
#####

#setwd("M:\\Desktop\\Thesis");
#setwd("H:\\Desktop\\Thesis");
list.files(); #Check to see if it's there!

#Load the Data
price=as.matrix(read.csv("SP500.csv")[,7]);
dates=as.matrix(read.csv("SP500.csv")[,1]);
n=nrow(price);
returns=matrix(0,n,1);
returns[1]=0;
returns[2:n]=price[2:n]-price[1:(n-1)];

#Getting returns for a set of holdings:
getprofits=function(holdings)
{
  profits=0;
  #Be Careful of the start index!
  for(i in 1:nrow(holdings)) {
    if (holdings[i]>0) {profits=profits+returns[i];}
    else      {profits=profits-returns[i];}
  }
  return(profits);
}

getideal=function(cost) { #Ex post ideal holdings
  ideal=matrix(0,n);
  for(i in 2:n) {
    ideal[i]=ideal[i-1];
    if(returns[i]>=cost) {ideal[i]=1;}
    if(returns[i]<cost) {ideal[i]=0;}
  }
  return(ideal);
}

numTrades=function(holdings) {
  trades=0;
  for(i in 2:nrow(holdings)) {
    if(holdings[i]!=holdings[i-1])
      trades=trades+1;
  }
  return(trades);
}

#Function for checking predictability
```

```

getacc=function(holding,ideal)
{
  return(sum(holding==ideal)/n);
}

#Function for getting annualized returns
getret=function(profits,days)
{
  return(profits/price[1]/days*365);
}

#####STRATEGIES#####

#Benchmark: Buy and Hold Strategy
basic=function() {
  series=price;
  holdings=matrix(1,nrow=nrow(series));
  #return(getprofits(holdings));
  return(holdings);
}
#Fama and Blume tested 0.5 percent to 50% price change.

filter=function(percent,cost)
{
  series=price;
  n=nrow(series);
  holdings=matrix(0,n);
  for(i in 2:n) {
    holdings[i]=holdings[i-1];
    diff=series[i]/series[i-1]-1;
    if(diff>=percent/100) {holdings[i]=1;}
    if(diff<=-percent/100) {holdings[i]=0;}
  }
  #return(getprofits(holdings)-numTrades(holdings)*cost);
  return(holdings);
}

#Simple Moving Average: 1% filter
ma=function(days,cost)
{
  am=function(series,days)
  {
    m=nrow(series);
    a=matrix(0,nrow=m);
    for(i in days:m) {
      a[i]=mean(series[(i-days+1):i]);
    }
  }
  return(a);
}

short=floor(days[1]);long=floor(days[2]);

```

```

if(short<2 || long<2) {return(-99999)};
series=price;
sma=am(series,short);
lma=am(series,long);
holdings=matrix(0,n);
for(i in max(short,long):n) {
  holdings[i]=holdings[i-1];
  if(sma[i]>lma[i]*1.01) {holdings[i]=1;}
  if(sma[i]<lma[i]*1.01) {holdings[i]=0;}
}
#return(getprofits(holdings)-numTrades(holdings)*cost);
return(holdings);
}

#Arithmetic-Harmonic Mean Difference:
hmstrat=function(theta,cost) {
  am=function(series,days)
  {
    m=nrow(series);
    am=matrix(rep(0,m),nrow=m);
    for(i in days:m) {
      am[i]=mean(series[(i-days+1):i]);
    }
    return(am);
  }

  hm=function(series, days)
  {
    m=nrow(series);
    hm=matrix(rep(0,m),nrow=m);
    for(i in days:m) {
      hm[i]=1/mean(1/series[(i-days+1):i]);
    }
    return(hm);
  }
  days=floor(theta[2]);percent=theta[1];
  percent=percent/100;
  series=price;
  hm1=hm(series,days);
  am1=am(series,days);
  crit=(am1-hm1)/am1*100; #scaled up arbitrarily
  holdings=matrix(0,nrow(series));
  for(i in days:nrow(series)) #days >=2
  {
    holdings[i]=holdings[i-1];
    if(crit[i] > percent) {holdings[i]=1;}
    if(crit[i] < percent) {holdings[i]=0;}
  }
  #return(getprofits(holdings,returns)-cost*numTrades(holdings));
  return(holdings);
}

```

```
#####NO TRADING COST ENVIRONMENT#####
```

```
benchmark=basic();
profits.benchmark=getprofits(benchmark);
returns.benchmark=getret(profits.benchmark,n);
acc.benchmark=getacc(benchmark,ideal0);
```

```
ideal0=getideal(0);
profits.ideal0=getprofits(ideal0);
returns.ideal0=getret(profits.ideal0,n);
acc.ideal0=getacc(ideal0,ideal0);
```

```
#From past optimization: for 0 TC, parameter is: perc=0.0009765625
filter0=filter(0.0009765625,0);
profits.filter0=getprofits(filter0);
returns.filter0=getret(profits.filter0,n);
acc.filter0=getacc(filter0,ideal0);
```

```
#From past optimization: for 0 TC, parameter is: days=c(2,19)
ma0=ma(c(2,19),0);
profits.ma0=getprofits(ma0);
returns.ma0=getret(profits.ma0,n);
acc.ma0=getacc(ma0,ideal0);
```

```
#From past optimization: for 0 TC, parameter is: days=82, perc=1.835567046
hmstrat0=hmstrat(c(1.835567046,82),0);
profits.hmstrat0=getprofits(hmstrat0);
returns.hmstrat0=getret(profits.hmstrat0,n);
acc.hmstrat0=getacc(hmstrat0,ideal0);
```

```
#####$8 TRADING COST ENVIRONMENT#####
```

```
cost=8;
```

```
benchmark=basic();
profits.benchmark=getprofits(benchmark)-numTrades(benchmark)*cost;
returns.benchmark=getret(profits.benchmark,n);
acc.benchmark=getacc(benchmark,ideal8);
```

```
ideal8=getideal(cost);
profits.ideal8=getprofits(ideal8);
returns.ideal8=getret(profits.ideal8,n);
acc.ideal8=getacc(ideal8,ideal8);
```

```
#From past optimization: for 0 TC, parameter is: perc=0.918175814
filter8=filter(0.918175814,0);
profits.filter8=getprofits(filter8);
returns.filter8=getret(profits.filter8,n);
acc.filter8=getacc(filter8,ideal8);
```

```
#From past optimization: for 0 TC, parameter is: days=c(41,360)
ma8=ma(c(41,360),0);
profits.ma8=getprofits(ma8)-numTrades(ma8)*cost;
returns.ma8=getret(profits.ma8,n);
acc.ma8=getacc(ma8,ideal8);
```

```
#From past optimization: for 0 TC, parameter is: days=136, perc=-10.58583745
hmstrat8=hmstrat(c(-10.58583745,136),0);
profits.hmstrat8=getprofits(hmstrat8);
returns.hmstrat8=getret(profits.hmstrat8,n);
acc.hmstrat8=getacc(hmstrat8,ideal8);
```