Why You Simply Must Time The Market

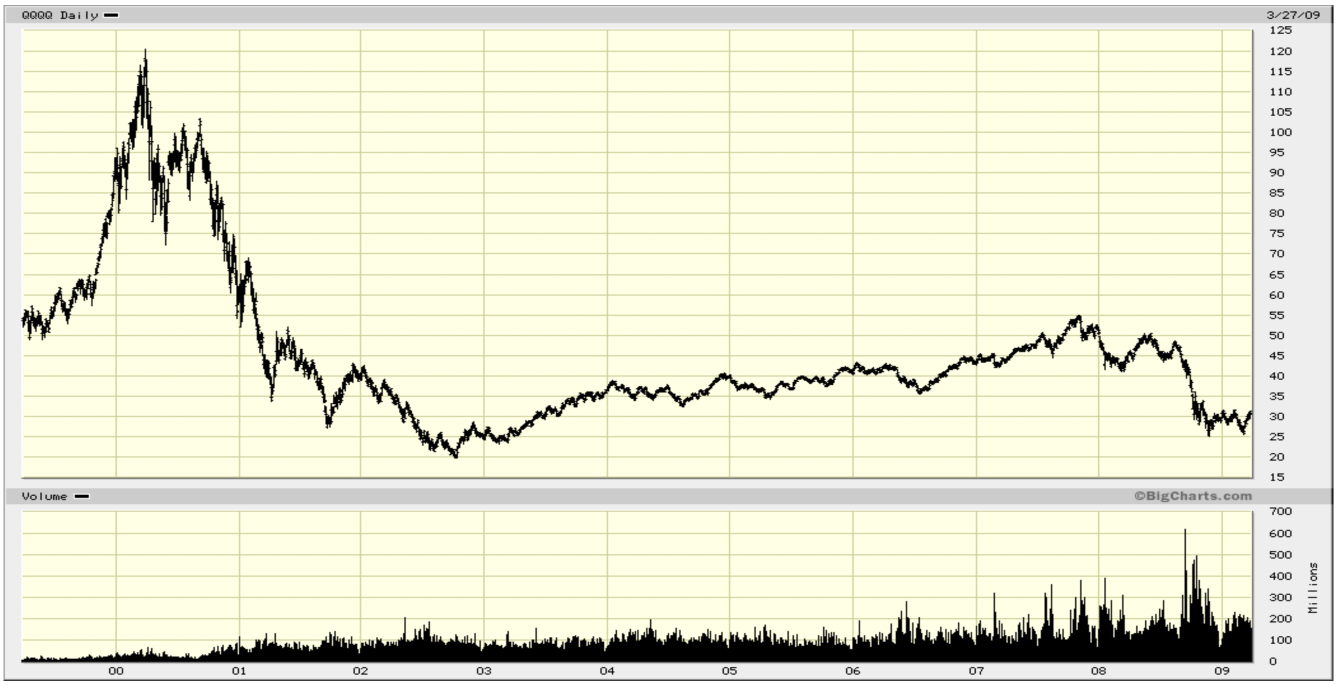
(And *How To Do It Using Artificial Neural Networks and Genetic Algorithms)*

Donn S. Fishbein, MD, PhD   Nquant.com

When repeated often enough and by increasing numbers, a supposition, however tenuous, may evolve into conventional wisdom or even consensus. This may occur bypassing the usual processes of the scientific method. Alchemy, geocentricity, vernalization, status thymicolymphaticus, eugenics and anthropogenic climate change are but a few past or current examples of claimed consensus which evolved without the strict discipline of the scientific method (Crichton 2005).

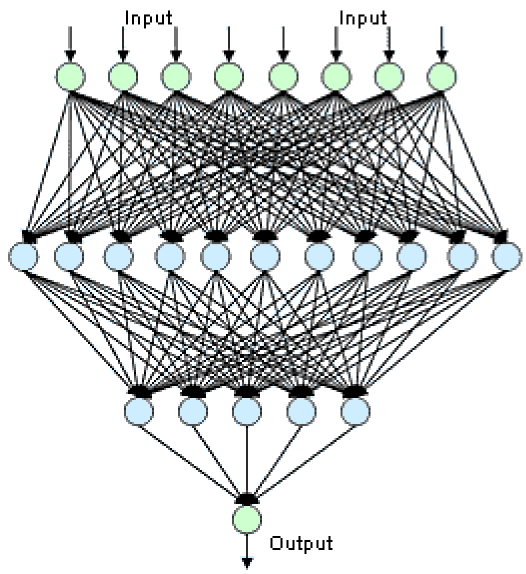
In the realm of financial markets, the random walk theory and efficient market hypothesis survive in spite of considerable evidence of their fallacy (Murphy 2004) . The random walk theory (or hypothesis) concludes that the stock market cannot be predicted. The efficient market hypothesis states that share price reflects all relevant information and therefore as a corollary one cannot outperform the overall market (Zweig 1997). Those who believe these suppositions would conclude that timing the financial markets is pointless, and the only viable strategy is buying stocks and holding them long term (O'Shaughnessy 1997). A true believer might even argue that it doesn't matter what stock you buy, since all are priced efficiently.

This buy-and-hold advice has been repeated often and loudly by many offering financial advice. Yet there are numerous counterexamples of its veracity. If the markets were truly efficient, stocks prices would move smoothly up and down in concert with the state of the economy. Bubbles and crashes would occur infrequently if at all. Intermarket arbitrage would not be possible, nor would it be possible to exceed buy-and-hold returns using simple trading systems, such as the 4% swing system (Fosback 1991, Arnold 1993). If the markets of the past ten years have told us anything, it is that buy-and-hold is a losing proposition (see chart below, Nasdaq-100 exchange traded fund March 1999 – March 2009).



If traders behaved in a rational manner, the market would be efficient and trading would offer few opportunities for consistent profit, but time and again market participants behave illogically, basing their decisions on emotional responses. (Weissman).

Classical technical analysis teaches that the best trading systems are robust and applicable across different instruments and time frames. This is in part to assure that a particular trading system is not curve-fit to a single time series, and will therefore work with future data. However, there is no reason that a system can not be trained on a particular time series and interval, as long as the validity of the system is later verified to work on out-of-sample data which the system has not seen during trading. The challenge, given the relatively short time series available for most financial instruments, as well as the changing nature of these instruments over time, is to assure that the success of a system in trading out-of-sample data is statistically significant and not a random occurrence (Arnold 1993, Aronson 2007).



All things being the same, an artificial neural network based trading system using a smaller number of inputs is superior to a similarly performing system using more inputs. Trading systems which use technical indicators such as moving averages and oscillators need to define parameters for indicators and state how indicator(s) are evaluated to produce trading signals. For example, a simple system might state: Buy when the 20 period simple moving average rises above the close. A more complex system might use more indicators, similar indicators with different parameters, and use these indicators in rules to produce buy and sell signals. The assignment of parameter values, the weighting of indicators, and how indicators are combined into a trading signal are often arbitrary and suboptimal. This may lead to a system using more inputs, and thus having more degrees of freedom, than is necessary. Such a system may be susceptible to curve fitting and prove more difficult to validate (Zirilli 1997).

Artificial neural networks (ANN) and genetic algorithms (GA) can assist in designing trading systems which have as few degrees of freedom as necessary. Artificial neural networks can discover relationships between indicators (inputs) and (profitable) trading signals which may be nonlinear and not be obvious to inspection (Fishbein 2005). Genetic algorithms can help optimize systems and do so faster than exhaustive searches. This speed is particularly important when genetic algorithms and artificial neural networks are combined in an iterative cycle of indicator optimization (GA) and indicator mapping to trading signals (ANN). Elimination of inputs to just before the point of decreasing profitability will result in a system with the fewest necessary degrees of freedom.

Assume that one or more indicators are chosen, and a hybrid artificial neural network/genetic algorithm system produces a trading system which is profitable for the interval over which the system is trained. The next step is to test the system out-of-sample, using data not used during training. The success or failure over a single out-of-sample interval is not sufficient to evaluate a system. It's little more than a role of the dice. Results may be anomalous and not representative of the ability of a system over the long-term. A system may be profitable or not over a single out-of-sample test and say little about the ultimate profitability of the system.

Walk-forward testing trains a system over a portion of the available market data, then tests over a small interval forward in time. The system is then retrained using the original training interval and the out-of-sample segment, then tested out-of-sample over the next interval of market data. The process is repeated until the available data is exhausted, and the results over multiple out-of-sample tests are calculated.

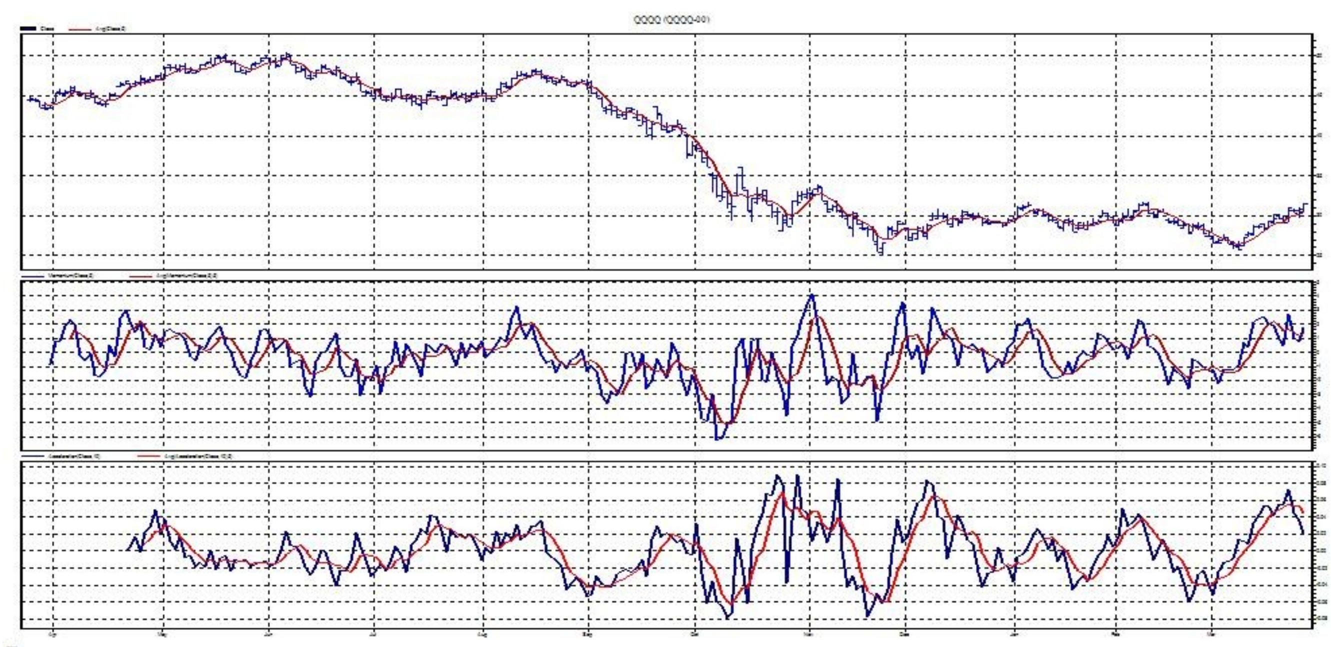
The fact that a system remains profitable over a number of walk-forward intervals doesn't in itself guarantee the system will remain profitable in the future. Even if enough data is available to reach levels of statistical significance, there is no certainty that the future will look like the walk-forward intervals. Consider the system mentioned several paragraphs back, Buy when 20 period simple moving average > close. This system performs well in a smoothly trending market, and poorly in most other markets. If the walk-forward periods encompass only a smooth uptrend, the system will show a successful walk-forward test and yet fail in a market which does not resemble the walk-forward periods.

What is needed is an limitless stream of market data which might represent all potential market conditions the system might be asked to trade under. While actual data is limited, Monte Carlo testing provides a method to generate synthetic data which closely simulates the data characteristics of the actual market under test, as detailed by several authors (Chande 1997, Aronson 2007). To summarize, synthetic data is created by picking an arbitrary starting value, and then incrementing the open, high, low and close for each period by a change or percentage change chosen from a bar in the actual data, chosen randomly for each period. By this method, a limitless supply of synthetic market data which retains characteristics of the original data can be produced. The system can be trained and tested over a number of synthetic data series to the desired level of statistical significance.

The following example shows the construction of a trading system using artificial neural networks and genetic algorithms, which is then validated using walk-forward and Monte Carlo testing. The six inputs are:

1. closing price
2. rate of change of the closing price (1st derivative, or momentum)
3. rate of change of the momentum (2nd derivative, or acceleration)
4. simple moving average of 1)
5. simple moving average of 2)
6. simple moving average of 3)

The six inputs are demonstrated on the chart below for the ETF QQQQ (Nasdaq-100) for the one year period ending 3/24/2009. The top graph shows the closing price (blue) and 5-day simple moving average (red), the middle graph the momentum and its simple moving average, and the bottom graph the acceleration and its simple moving average.



To train and test the system, 10 years of daily data for the exchange traded funds QQQQ (Nasdaq-100), SPY (S&P 500), DIA (Dow Jones 30), and IJR (S & P Small Cap 600) were used. Having specified the 6 inputs, the software evaluates neural networks with indicators and indicator parameters chosen by a genetic algorithm in an iterative fashion (Fishbein 2008). An initial 2 month out-of-sample period is specified, and the out-of-sample results shown in the table below:

	Annualized Return	Profitable Trades	Max Drawdown
DIA	6.2%	48%	13%
IJR	24.8%	50%	16%
QQQQ	37.9%	60%	11%
SPY	-4.8%	42%	10%

While three of the four exchange traded funds showed positive returns, results were variable. While somewhat encouraging, these results alone do not ensure the system would be profitable in future markets. As previously discussed, the single out-of-sample period might have characteristics particularly suitable to the design of the system. It would be useful to test the system against a number of out-of-sample scenarios.

Walk-forward testing was next performed using 40 periods of 40 samples each. In the initial test, data up to the last 1600 trading days was used to optimize the system. Then, the next 40 trading days were used as an out-of-sample test, in which the optimized system was used to trade these 40 days without any further optimization. The indicator parameters and network weights were maintained as determined during optimization, and the results of the out-of-sample test recorded. Next, the system was re-optimized using the original optimization period plus the 40 days previously used for the out-of-sample test, and the new system tested out-of-sample on the next 40 trading days. This process is repeated a total of 40 times, and the trading results for the 40 out-of-sample periods tabulated.

	Number of Walk-forward periods	Average Annualized Return	Average Profitable Trades	Average Max Drawdown
DIA	40	19.8%	56%	16%
IJR	40	29.7%	61%	18%
QQQQ	40	47.8%	63%	19%
SPY	40	12.4%	59%	16%

Profitability improved for each exchange traded fund, as did the percentage of profitable trades, at the expense of slightly higher drawdowns. It may be the interval over which the initial single out-of-sample test fell was not as favorably traded with this system as a longer segment of market activity. In any case, the results so far are favorable and this system has potential as a useful tool.

The number of walk-forward tests that can be performed is limited by the length of the trading history for the instrument under test. Sufficient data must be available for the initial optimization. It is also possible that even a lengthy walk-forward period may not adequately represent market conditions the system may see in the future. Monte Carlo testing offers a method to generate endless data which closely matches the characteristics of the original data series. The intricacies and applications on Monte Carlo testing are addressed elsewhere (Aronson 2007).

For each exchange traded fund under test, the method described by Chande (1997) was used to construct 1000 synthetic data series of 10 years in length. Each data series was then used to train the hybrid artificial neural network/genetic algorithm system described above, with the last two months of data used as an out-of-sample test. The average annualized return, percentage of winning trades, and maximum drawdown were calculated for each out-of-sample period for each ETF and summarized in the table below:

	Number of Synthetic Data Series	Average Annualized Return	Average Profitable Trades	Average Max Drawdown
DIA	1000	16.4%	61%	21%
IJR	1000	30.1%	52%	19%
QQQQ	1000	42.8%	57%	24%
SPY	1000	12.9%	62%	14%

Results for each ETF remained positive and did not show significant variation from those obtained with walk-forward testing. Average maximum drawdown was higher in Monte Carlo testing than in walk-forward testing. These results give added confidence that the system produces reproducible results and its results do not represent a statistical anomaly.

Summary

Buy-and-hold investing strategies for the stock market are untenable. Systematic mechanical trading provides one way to time the market, and lends itself to testing and validation. The combination of artificial neural networks and genetic algorithms offers a unique way to develop powerful trading systems. A hybrid artificial neural network/genetic algorithm system using the closing price, its 1st and 2nd derivatives, and simple moving averages of the same was described. The system generated positive returns in the initial out-of-sample test period. A walk-forward test extending back 1600 trading days was performed, again showing positive returns for each instrument. Finally, Monte Carlo testing was performed using 1000 synthetic data series for each instrument, showing positive returns. Timing the stock market using a mechanical trading system based on artificial neural networks and genetic algorithms provides a statistically significant increase in trading returns over a buy and hold strategy.

BIBLIOGRAPHY

Arnold, Curtis M. *Timing The Market.* Weiss Research, Inc., 1993

Aronson, DR. *Evidence Based Technical Analysis*. John Wiley and Sons, 2007.

Chande, TS. Beyond Technical Analysis: *How to Develop and Implement a Winning Trading System*. John Wiley and Sons, 1997.

Crichton, M. *State of Fear. Appendix I: Why Politicized Science Is Dangerous*. Harper Collins, 2005.

Fishbein DS. How to time the stock market Using Artificial Neural Networks and Genetic Algorithms. Trenton Computer Festival Proceedings, 2004

Fishbein DS. Identifying Short Term Market Turns Using Neural Networks and Genetic Algorithms. Trenton Computer Festival Proceedings, 2008

Fosback, NG. Stock Market Logic. Dearborn Financial Publishing, 1991.

Masters, T. Monte Carlo Evaluation of Trading Systems. 24 Mar 2009, 16:13 UTC. <http://www.evidencebasedta.com/MonteDoc12.15.06.pdf>

Murphy John. Intermarket Analysis. Wiley Trading, 2004.

"Monte Carlo method." *Wikipedia, The Free Encyclopedia*. 18 Mar 2009, 15:14 UTC. 25 Mar 2009 <http://en.wikipedia.org/w/index.phptitle=Monte_Carlo_method&oldid=278118547>.

O'Shaughnessy, JP. What Works on Wall Street. McGraw-Hill, 1997.

Zirilli, Joseph S. Financial Prediction Using Neural Networks. International Thomson Computer Press, 1997.

Zweig, M. Winning of Wall Street. 1986, 1997(revised), Warner Books.