www.ijecs.in
International Journal Of Engineering And Computer Science
Volume 12 Issue 11 November 2023, Page No. 25937-25941
ISSN: 2319-7242 DOI: 10.18535/ijecs/v12i11.4775

Automated Trading Strategies Using Artificial Intelligence

Ritesh Biswas[1], Abhay Das[2], Anirban Bhar[3], Soumya Bhattacharyya[4]

[1,2] B. Tech student, Department of Information Technology, Narula Institute of Technology, Kolkata, India.
[3,4] Assistant Professor, Department of Information Technology, Narula Institute of Technology, Kolkata, India.

Abstract
There are a lot of variables that might affect the financial market, and those elements aren't always predictable. As a result, predicting future stock price changes is no easy feat. The goal of machine learning is to discover patterns in massive datasets automatically. Machine learning algorithms have the potential to automate trading methods that are based on their predictions of stock price movements because of their self-organizing and self-learning qualities. Market moves have been predicted using AI methods, yet published methods seldom involve testing in an actual (or virtual) trading setting. In this paper we are in search of a complete automatic trading, where even the buy/sell decisions are taken by the computer.

Keywords: Artificial Intelligence, Machine Learning, Automated Trading, Algorithmic Trading.

1. Introduction

Shares of publicly traded corporations can be bought and sold on the Stock Market. Investors from all over the globe can purchase and sell shares of stock in a company when it goes public through an Initial Public Offering, or IPO. The price of these shares reflects the market value that investors assign to a portion of the company's ownership, or the stock's potential future appreciation. Therefore, market forces of supply and demand are the only factors that impact stock prices.

Investors often have two options: one is to enter a long position, where they hold shares of a firm, and the other is to place a "buy" order for those shares on the market. This happens when the order is executed, and the investor owns those shares. Investors who hold long positions in this company's stock stand to gain as the demand for its shares rises in tandem with the number of others holding this view. Otherwise, demand falls, the stock price falls, and these investors lose out since more people think the company will be worth less in the future. But in this scenario, an investor who also predicts a decline in a company's share price can short the stock by placing a "sell" order. If he is a long-term shareholder in this company, the sell order will liquidate his holdings to the specified amount. "Short selling" occurs when he does not already own the stock but borrows it from someone else to sell quickly. "Buy cover" occurs when he wants to repurchase the same number of shares and gives them back to the original owner. Investments can benefit from a decline in price through the practice of short selling.

There are a wide variety of investors with varying risk tolerances and return expectations who take part in the stock market. Everyone from retail traders to large-scale financial institutions like hedge funds, mutual funds, and exchange-traded funds (ETFs) and even computer trading algorithms are in it for the long haul: to make a killing by correctly predicting when stock values will rise or fall.

Both the trading frequency and the trading "universe" must be specified by any investor, but notably by computer trading algorithms. The frequency with which a trade is made is called the trading frequency. While individual investors may have more leeway in deciding how often to trade, high-frequency traders like Warren Buffett tend to trade seldom, and day traders place numerous orders daily in pursuit of intraday profit. The systematic nature of automated trading systems necessitates that their developers specify the trading frequency prior to their creation. A lifetime of buying and holding is one extreme end of the spectrum, whereas High Frequency Trading algorithms trade as often as once every nanosecond. Therefore, trading frequency can have a significant impact on model selection and strategy formulation.

The range of equities that an investor chooses to trade on is called a universe. A "global market" investor, for instance, would not restrict his stock selection to any particular region. The United States Stock Market, the S&P 500, or even a specific S&P 500 sector can also be considered a universe. A popular measure of the health of the American stock market as a whole is the Standard & Poor's 500 Index, more often known as the S&P 500. Among other things, market capitalization, liquidity, and industry clustering were considered while selecting the 500 equities that make up the index. The goal of the S&P 500 is to capture the volatility and potential reward of the US large cap universe [1]. The primary goal of the first and most widely used exchange-traded fund (ETF) in the United States, with the ticker SPY, is to mimic the overall return of the S&P 500 index as closely as possible [2]. Those interested in investing in the US market can do so through the SPY ETF.

Further, the S&P 500 can be segmented according to various industries, such as energy, healthcare, materials, utilities, financial services, healthcare, industrials, information technology, and consumer discretionary and staples [3]. Within the S&P 500 universe, this study proposes an automated trading system that focuses on the Energy, IT, and Utilities sectors. Our algorithm compares its performance to that of a buy-and-hold strategy on SPY, and it trades every day.

2. Motivation

Trading becomes more methodical with algorithmic trading, which is its main advantage. Emotions run high among human investors. Feelings of euphoria or the "fight or flight" reflex can be experienced depending on whether a position is profitable or not. One can become more greedy or afraid, leading to a loss of trading discipline, and these reactions also shut down the regions of the brain that are responsible for reasoning and logic [4]. Making money, and making it consistently, in the stock market requires discipline, in the form of a trading plan. The emotional variables mentioned earlier can easily undermine trading discipline when confronted with these temporary drawdowns, and no plan can consistently provide positive returns. Consequently, an automated trading system allows us to attain consistency by following our trading plan systematically, which eliminates emotions throughout the trading process, provides discipline even in unpredictable market situations, and helps us stay on track.

The capacity to test strategies in the past is another perk of algorithmic trading that sets it apart from human traders. By comparing a trading system's performance to data from a previous time period, or "back testing," we may see how the system would have fared. System developers can gain insight into the past and future performance of a trading strategy in specific scenarios. By adjusting the model's parameters with each iteration, it also offers the chance to optimize a trading strategy [5]. lucrative trading rules from the past will continue to be lucrative in the future, according to the crucial premise that the statistical features of the price series are unchanging [6]. This is how back testing derives its predictive ability. But there's a chance that this assumption will turn out to be wrong if things like the financial market's structure, the company's fundamentals, or the macroeconomic outlook change. Additionally, traders can get a practical and effective way to evaluate the strategy by separating historical data into in sample and out of sample sets during back testing [7]. To optimize the trading strategy using in-sample data, it is vital to test the system on clean out-of-sample data to see if it works and to remove overfitting.

Plus, unlike human investors, automated trading systems can accurately time the market. As the renowned investor Warren Buffett once said, "We simply attempt to be fearful when others are greedy and to be greedy only when others are fearful." When it comes to day trading, this is one way to try to time the market. When investors are overly eager to make a quick buck, stock prices tend to rise, which means traders should cash out. Even Buffett warned against trying to timing the market by saying, "our favorite holding period is forever," but it is quite difficult to reliably tell when individuals are being greedy or scared. Conversely, High Frequency Trading businesses engage in another method of market timing by identifying arbitrage opportunities and trading them in microseconds. A human being just cannot do so at these frequencies. Nevertheless, computerized trading programs are more adept at market timing and can even multitask, allowing a system to time thousands of stocks all at once. For this reason, high-frequency trading (HFT) companies rely on algorithmic trading. Trading systems can also do a respectable job of detecting market greed and fear when coupled with reliable financial timeseries forecasting and sentiment analysis (the study of news items or tweets).

3. Methodology:

3.1. Individual Prediction Model

First, we developed an individual stock price prediction model that leverages a stock's historical performance to estimate its future price. The objective is to leverage historical stock movement patterns to estimate future prices in new Nday windows. This simplistic model ignores essential elements like general market information and rival company performance, but we think it can be a suitable starting point and baseline for complex models.

Some model implementation specifics must be addressed. First, we want to make this a classification problem instead of a regression problem since we can use probability models like logistic regression to manage the “confidence level” of the predictions and we only need to forecast the direction, not the price. Let's use a regression problem with a stock price of $100 and $101 today and tomorrow. Two models forecast $99 and $110 tomorrow. The first one is near to the genuine price but suggests selling, so we lose money, while the second one, with bigger mistake, gets the proper direction and profits us. Second, we generate the dataset using daily returns, not prices. As geometric random walks, most price series contradict the econometrics condition that the anticipated value of error in a regression be zero. However, returns are frequently random around zero [6].

To prevent overfitting, we added ridge penalty to logistic regression. By restricting parameters, a ridge penalty can enhance logistic regression parameter estimates and reduce out-of-sample error.

We can limit any calculated coefficient to be too large by modifying lambda. This would help our model anticipate tomorrow's return using last seven days' returns. Since yesterday's return is more relevant to today, its coefficient would be much higher. One parameter dominating increases the danger of overfitting, so the model will assign too much weight to yesterday when we want to consider the complete week. By imposing the penalty, we may prevent this and weight the “further back” characteristics more.

3.2. Sector Prediction Model

The sector technique lets our model consider all stocks in a sector when predicting a stock's price. This is far superior to the individual approach. The logistic regression model above cannot predict Apple's stock price for tomorrow because it doesn't know what's happening outside of Apple's stock price. However, Apple's stock price may decline if Google's stock rises on news of its complete smartphone entry. The model cannot identify and evaluate occurrences like this one, but it can take advantage of correlations between a company's stock price and that of others in the same industry by looking at stock returns.

Our sector approach model used lasso logistic regression, decision tree, naive bayes, and support vector machine (SVM) machine learning algorithms. We also used majority voting and adapted random forest ensemble approaches.

Majority vote is a simple ensemble method that uses all four basic algorithms. Lasso, decision tree, naive bayes, and SVM each vote for each observation, and the majority vote model predicts when at least three votes agree. This makes majority vote model forecasts more reliable.

The second ensemble approach we created, Random Subset, uses a random forest. Randomly select subsets of sector stocks as predictors. We use a simple method like naive bayes as the model. We next create 100 training sets, each randomly selecting 10% of sector stocks, and train naive bayes on them. Once we have 100 votes, Random Subset will forecast a path when at least half agree. Each underlying model can forecast up, down, or none depending on our threshold. Random Subset may not anticipate every observation.

We will choose the best four basic and two ensemble models to back test on real out-of-sample data in the next step.

4. Conclusion and Future Scope

It turned out that looking at a company's stock price history alone wasn't enough to forecast its future profits. It would have been more accurate to consider the entire sector that the target firm was a member of and utilize the historical price data of all companies in the sector to forecast the target's return the next day. A few models reached statistically significant levels of accuracy near 60% in a three-month in sample validation period. We also discovered that using Ensemble Methods, which combine basic algorithms, or even just tweaking a basic machine learning algorithm—like ROC curve optimization—can significantly enhance accuracy.

There were indications of market timing success using a tailored trading strategy that makes use of our model forecasts. During the out-of-sample testing period, our trading systems demonstrated encouraging Sharpe ratios, with a return of approximately 20% after transaction costs. Not only that, they surpassed the risk-free interest rate and, in certain cases, even the S&P 500 index. We were able to confirm that our automated trading method could reliably produce positive returns through back testing on unseen data.

As a future scope we may propose to develop and use a market simulator that will give you a place to test your trading tactics. The model is expected to use real stock prices and should take on a sped-up version of how the real market works. It must allow the strategies to put in buy or sell orders, to execute at the market price (including trade costs), and must check the value of the portfolio every day.

References

1. Investopedia. "Standard & Poor's 500 Index (S&P 500) Definition" 2016. http://www.investopedia.com/terms/s/sp500.asp
2. Investopedia. "ExchangeTraded Funds: SPDR S&P 500 ETF" 2008. http://www.investopedia.com/university/exchangetradedfund/etf3.asp
3. Wikipedia. "List of S&P 500 companies" 2011. https://en.wikipedia.org/wiki/List_of_S%26P_500_companies
4. Beder, Tanya S., and Cara M. Marshall. Financial Engineering: The Evolution of a Profession . Hoboken, NJ: John Wiley & Sons, 2011. Print.
5. Ni, Jiarui, and Chengqi Zhang. "An efficient implementation of the backtesting of trading strategies." Parallel and Distributed Processing and Applications (2005): 126131.
6. Chan, Ernest P. Algorithmic Trading: Winning Strategies and Their Rationale (Wiley Trading Series) . John Wiley & Sons, 2013. Print.
7. Investopedia. "Backtesting And Forward Testing: The Importance Of Correlation" 2010.

http://www.investopedia.com/articles/trading/10/backtestingwalkforwardimportantcorrelation.asp