A Machine Learning Approach to Stock Selection using Technical Analysis

Brian S. Wickman

April 13, 2022

An undergraduate thesis presented for Honors in Economics

Department of Economics
Washington and Lee University

Abstract

This thesis introduces a novel machine learning framework for stock selection that only uses technical indicators and chart patterns as inputs. In contrast to other papers, the machine learning model first employs a recursive feature elimination algorithm to carefully select model inputs before a support vector machine predicts the direction of the following trading day's price movement. I then evaluate the accuracy of the model's predictions and compare the economic returns of the machine learning algorithm's trading strategy to a buy-and-hold approach and a simple MACD trading strategy on 48 stocks from 2010 through 2019. The 48 stocks selected for the study represent three different types of stocks: there are 30 large-cap U.S. stocks, 10 small-cap U.S. stocks, and 9 European stocks. I find that the machine learning model generated a higher economic return than the buy-and-hold approach for 10 of the 48 stocks. All ten of these stocks were large-cap stocks which suggests that the machine learning model performs best with large-cap stocks over this time period. All in all, this paper supports the adaptive market hypothesis and provides evidence that machine learning algorithms and technical analysis could not be used to consistently generate returns in excess of the buy-and-hold in the low-volatility market conditions of the 2010s.

Acknowledgements

I would like to express my gratitude to my main supervisor, Dr. Aliaa Bassiouny, who provided me with invaluable guidance and direction over the past year. Furthermore, I would like to thank Dr. Joseph Guse for his continued feedback throughout the process and Dr. Mario Negrete for his support. I also wish to acknowledge the resources provided by Washington and Lee University, especially the IQ center and Economics department. Lastly, I would to thank my parents, family, teammates, and God for making this possible.

Contents

List of Figures

Chapter 1

Introduction

I Neoclassical Finance

Theoretical asset pricing models in neoclassical finance rely on the assumption of efficient markets and rational investors. One of the main principles of neoclassical finance is Eugene Fama's 1970 efficient market hypothesis. [23]. Fama's efficient market hypothesis (EMH) argues that competition between rational investors drives prices to their true intrinsic value that can be explained by traditional asset pricing models. This implies that asset prices reflect all available information regarding a security's value, making it impossible for investors to make money by buying (selling) undervalued (overvalued) stocks. The most basic of the asset pricing models is the Capital Asset Pricing Model (CAPM) which argues that returns compensate investors for investment risk and the time value of their money. Fama and French built on the CAPM and proposed their three-factor model in 1992. They argue that returns can be explained by investment risk, market capitalization, and book-to-market ratio. In their paper, they find that small cap stocks with high book-to-market ratios tended to outperform the market as a whole [25]. They conclude that more than 90% of asset returns could be explained by their three-factor model, which is considerably better than the CAPM's 70% explanatory power. If markets were perfectly efficient, one would expect 100% of asset returns to be explained by asset pricing models, but Fama and French attribute this to not having the ideal subset of factors. In 2015, Fama and French sought to improve on their three-factor model and proposed their five-factor model which added profitability and investment factors [26]. While the model outperformed their three-factor model, it failed to explain the low average returns of small companies that prioritize capital investment. More recent literature has confirmed these results and found the five-factor model is unable to consistently explain asset returns [32].

II Behavioral Finance

The failure of asset pricing models to explain equity returns brings into question the efficiency of markets and the rationality of investors. In response, behavioral finance is predicated on the idea that investors can be irrational at times and do not always process information correctly due to behavioral biases inherent within human nature. A variety of behavioral biases like confirmation bias and regret avoidance can lead to sub-optimal investment decisions. As a result, behaviorists reject Fama's assertion that financial markets are perfectly efficient. Instead, the literature is now evolving towards the adaptive market hypothesis (AMH).

Andrew Lo, a behavioral economist at MIT, proposed the AMH in 2003 as a weaker form of the EMH; he argues that markets are sometimes not efficient because investors can be irrational and overreact during periods of high market volatility [47]. This theory implies that investors can generate returns in excess of the market by identifying market inefficiencies caused by irrational investors. A powerful example of behavioral finance is the dot-com bubble, which was caused by unrestrained speculation on internet stocks from 1995-2002. This bubble is associated with the a 582% increase in the NASDAQ Composite Index from 1995-2000 and a subsequent 75% decrease from 2000-2002 that wiped out most of the aforementioned gains[1]. In short, behavioral finance can be used to explain these massive fluctuations in security prices despite no changes in the underlying assets.

III Technical Analysis

I Overview

There are two main investment approaches used to identify and profit from market inefficiencies: fundamental analysis and technical analysis.

Fundamental analysis (FA) seeks to intrinsically value a company by analyzing financial statements and economic factors. Balance sheets, 10-Ks, and macroeconomic indicators are the primary tools of the fundamental analyst. FA also accounts for microeconomic factors that can impact a firm's value like the effectiveness of the board of directors and specific industry conditions. Through the use of these tools, an investor can try to forecast a company's cash flows and determine if the company is undervalued or overvalued by the market. It is important to note that FA could still be effective if markets were perfectly efficient because FA attempts to forecast future financial events that are obviously not incorporated into a security's market value. Perhaps the most well-known practitioner of FA is Warren Buffet, or the 'Oracle of Omaha', who has amassed a personal fortune in excess of $102.5 billion with this investment approach.

In contrast with FA, technical analysis (TA) does not attempt to estimate the intrinsic value of a company because it assumes that the company fundamentals are already priced into the stock. Instead, TA is grounded in behavioral finance and focuses on statistical trends in a security's price and volume to understand the underlying market psychology. This implies that TA operates under the assumption that past price trends can provide valuable information about the future.

TA is best-suited for investors with a shorter-term investment horizon because the factors examined by fundamental analysts are only updated quarterly with the release of financial statements. Conversely, TA relies on price data that is recorded down to the millisecond. Additionally, TA is favored by retail traders[2] because price information is widely available online. On the other hand, it is difficult for retail traders to gain a competitive edge with FA due to the amount of holistic data required to accurately value a company. That being said, the largest investment banks and hedge funds around the world invest heavily in technical analysis and hire specialized technical analysts to provide input on investment decisions. A survey conducted among 692 fund managers across five countries found that 86% of fund managers rely on TA and

[1] On a side note, technical analysis has been shown to be profitable during financial crises [6]. This is not surprising because technical analysis uses price information as a barometer for market psychology, which dominates during asset bubbles.

[2] A retail investor is a non-professional investor who buys and sells securities or funds for their personal trading account [36].

26% consider TA to be the most important criterion in the investment process [51].

There are two subfields of technical analysis- technical indicators and chart pattern analysis- from which I would like to draw an important distinction. Both seek to exploit investor psychology, although their processes are considerably different. First, technical indicators represent a strict, empirical approach to technical analysis and are rooted in statistical formulas. These formulas provide black-and-white trading signals; an indicator will either highlight a profitable trading opportunity or not. Technical indicators include moving averages, momentum measures, and stochastic oscillators. On the other hand, chart pattern analysis examines patterns on price and volume charts to generate trading signals. Chart pattern analysis tends to be more of an art than a science as there exists a substantial amount of gray area regarding investment decisions. Please keep this distinction in mind as you progress through the remainder of the paper.

II History

Technical analysis dates back to 18^{th} century, Edo-period Japan, where traders applied technical analysis to the Osaka Dojima Commodity Exchange's rice futures market. A speculator by the name of Munehisa Homma wrote the earliest descriptions of chart patterns in his book *The Fountain of Gold- the Three Monkey Record of Money.* Homma was able to amass a spectacular fortune through the use of candlestick charts, a success that he attributed to the trends of human emotions exhibited in market behavior.

It was not until 1920 that TA rose to prominence in the United States as financial journalist Charles Dow developed his "Dow Theory." Dow and his followers contend that market trends have three phases driven by investor psychology. First, the accumulation phase, or *phase one*, begins when investors with privileged information actively buy (sell) stock against the general opinion of the market. Note that privileged information is not necessarily insider trading; Dow Theory assumes that the flow of information is imperfect. The stock price does not change during this period because the "in-the-know" investors are in the minority compared to the uninformed market. As the pertinent information flows through the economy, the market quickly corrects the price of the stock in *phase two.* At this point, the trend-followers participate until rampant speculation and a gross overcorrection occurs. Lastly, the well-informed investors unload their positions and collect their profits in *phase three.* In order to differentiate the different phases, Dow looked at the moving averages of historical prices, a primitive form of TA. Dow Theory is important because it formally recognizes that investor psychology can significantly impact the market and employs technical analysis to identify profitable opportunities.

IV Relevant Technical Indicators and Chart Patterns

Before reviewing the existing literature, I would like to offer a brief overview of the relevant technical indicators (**Figure 1.1, Figure 1.2**) and chart patterns (**Figure 1.3, Figure 1.4**).

Overview of Technical Indicators			
Indicator Name	Type	Overview	Formula
Average Directional Movement (ADX)	Trend	Quantifies trend strength using a moving average of price range expansion	$ADX_n = \frac{(ADX_{n-1} \cdot (n-1)) + DX_n}{n}$ *where:* $DX_n = \frac{100 \cdot (PDI_n - MDI_n)}{PDI_n - MDI_n}$ $PDI_n = (CurrentHigh) - (PreviousHigh)$ $MDI_n = (PreviousLow) - (CurrentLow)$ n = smoothing parameter
Bollinger Bands	Volatility	Measures a standard deviation above and below the simple moving average of the price	$UpperBand = MA(TP, n) + m \cdot \sigma[TP, n]$ $LowerBand = MA(TP, n) - m \cdot \sigma[TP, n]$ *where:* $MA = Moving\ Average$ TP = (High– Low + Close)/3 m = number of standard deviations
Chaikin Volatility	Volatility	Quantifies security volatility by comparing a security's high and low prices	Chaikin Volatility = $EMA(High - Low)$
Ease of Movement Value (EMV)	Momentum	Measures the momentum of a stock by focusing on the relationship between price and volume	$EMV = \frac{\left(\frac{High + Low}{2} - \frac{PreviousHigh - PreviousLow}{2}\right)}{\frac{\frac{Volume}{Scale}}{High - Low}}$ *where:* Scale ranges from 1,000 to 1,000,000,000 based on average daily trading volume
Exponential Moving Average (EMA)	Trend	Gauges trend direction over time by applying more weight to recent data	$EMA_n = Price \cdot \left(\frac{Smoothing}{1 + Days}\right) + EMA_{n-1}\left(1 - \left(\frac{Smoothing}{1 + Days}\right)\right)$

Figure 1.1: Overview of Technical Indicators [1].

Overview of Technical Indicators (cont.)			
Indicator Name	Type	Overview	Formula
Moving Average Convergence Divergence (MACD)	Trend	Examines the relationship between two moving average of a security's price	$MACD = EMA(12) - EMA(26)$ *where:* *MACD>0* is bullish *MACD<0* is bearish
On Balance Volume (OBV)	Volume	Uses the volume flow as a proxy for crowd psychology to make price predictions	$\text{OBV} = \text{OBV}_{prev} + \begin{cases} \text{volume}, & \text{if close} > \text{close}_{prev} \\ 0, & \text{if close} = \text{close}_{prev} \\ -\text{volume}, & \text{if close} < \text{close}_{prev} \end{cases}$
Rate of Change (ROC)	Momentum	Measures the percentage change in price between current price and the price a specified number of periods ago	$ROC = \left(\frac{(ClosePrice)_p - (ClosePrice)_{p-n}}{(ClosePrice)_{p-n}}\right) \cdot 100$
Relative Strength Index (RSI)	Momentum	Calculates the magnitude of recent price changes to identify trend reversals	$RS = \frac{SMA(Up\ changes\ in\ previous\ n\ days)}{SMA(Down\ changes\ in\ previous\ n\ days)}$ $RSI = 100 - \frac{100}{1 + RS}$
Simple Moving Average (SMA)	Trend	Examines the average price over a specified period of time	$SMA(n) = \frac{1}{n}\sum_{1}^{n}(p_n)$
Stochastic Oscillator (%K)	Momentum	Shows the location of the closing price to the high-low range over a specified number of periods	$\%K = \left(\frac{Close - Low_n}{High_n - Low_n}\right) \cdot 100$ *where:* $High_n$ is the highest close price in n periods
Vertical Horizontal Filter (VHF)	Trend	Evaluates the strength of a price trend using changes in the high-low price spread	$VHF = \frac{High_n - Low_n}{\lvert Sum(Change, n) \rvert}$ *where:* $High_n$ is the highest close price in n periods
Williams Percentage Range (%R)	Momentum	Ranges from 0-100 and quantifies overbought and oversold levels	$\%R = \frac{High_n - Close}{High_n - Low_n}$

Figure 1.2: Overview of Technical Indicators (cont.) [1].

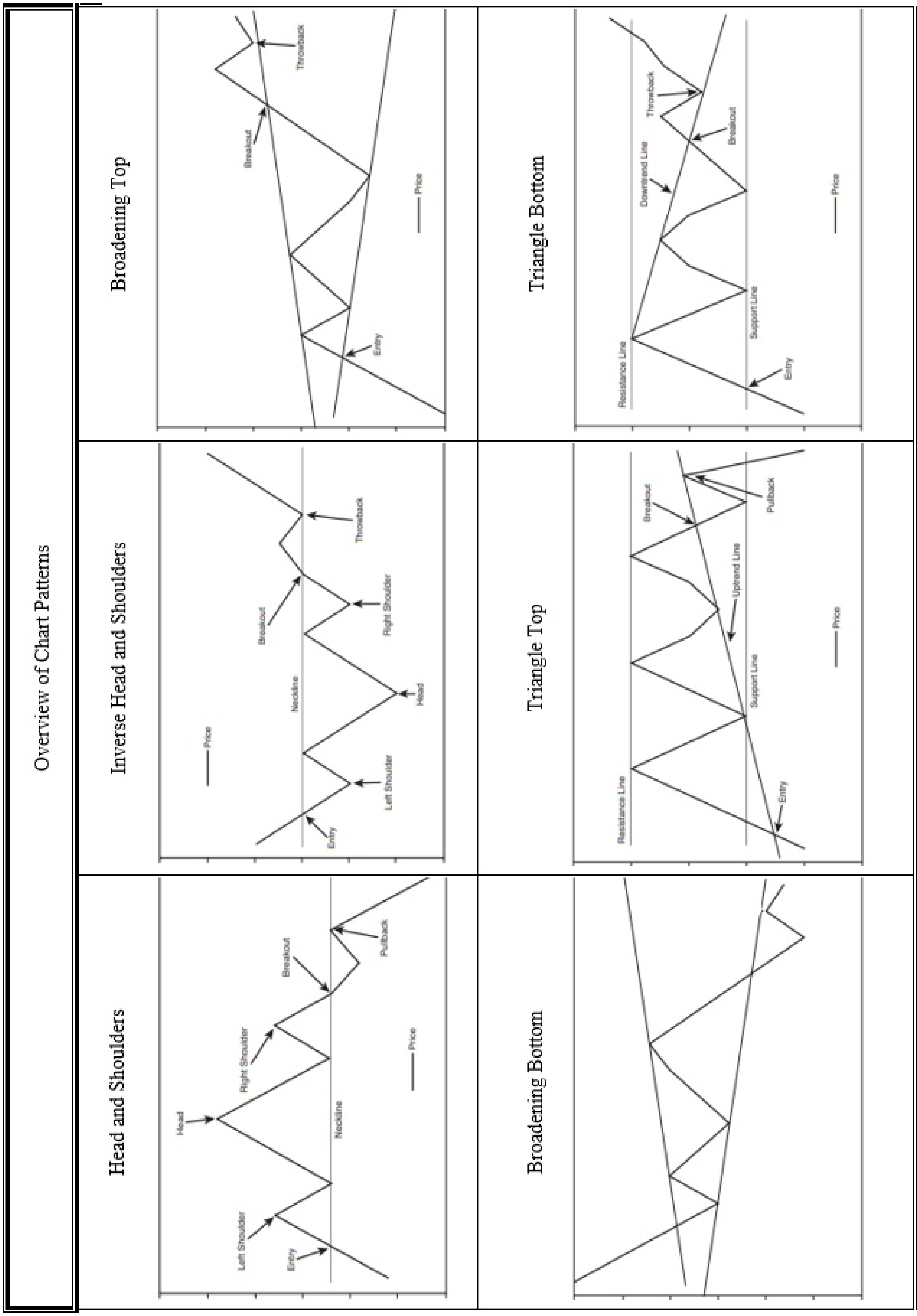


Figure 1.3: Overview of Chart Patterns [42].

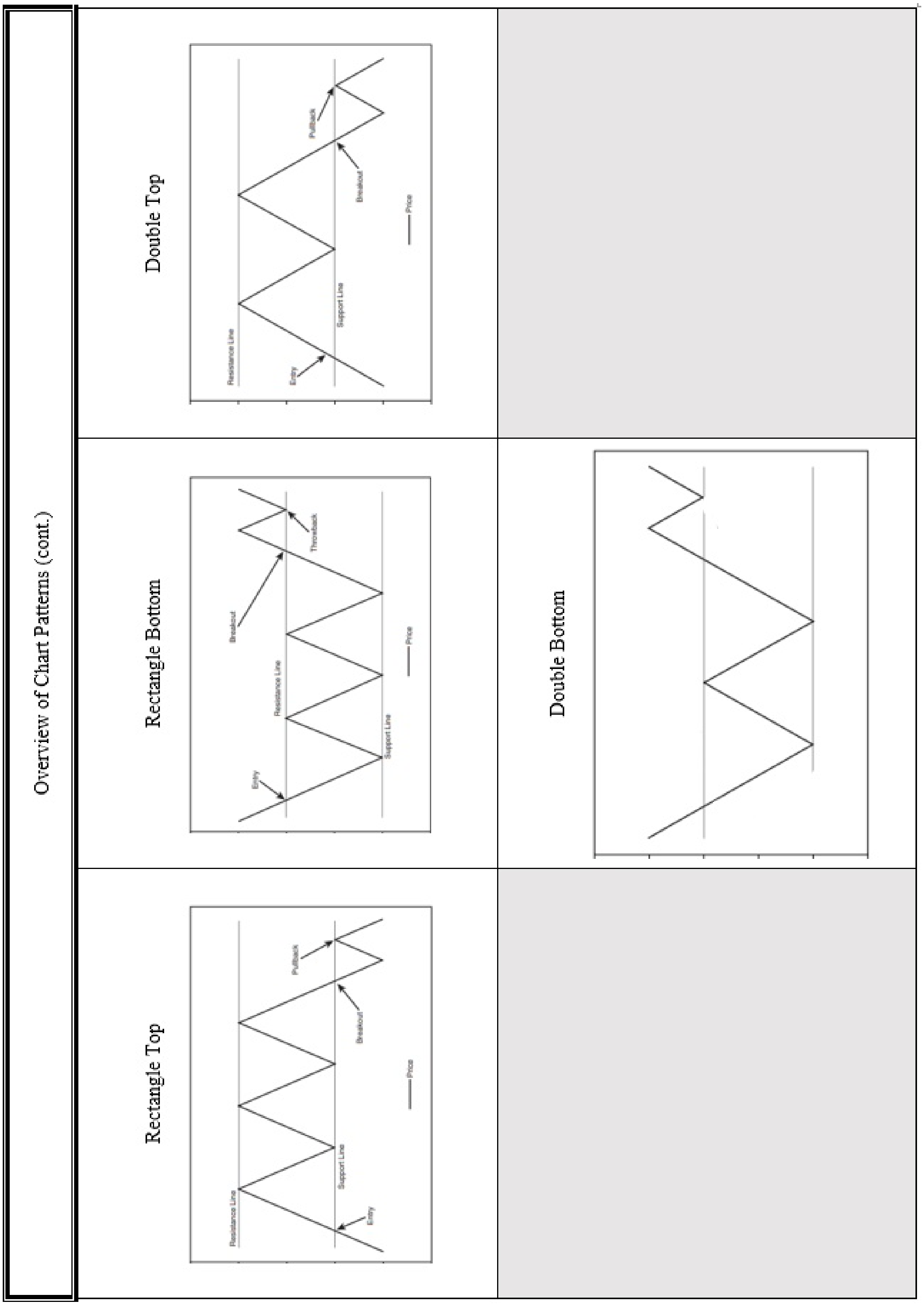


Figure 1.4: Overview of Chart Patterns (cont.) [42].

Chapter 2

Literature Review

I Overview

The literature review is organized as follows; In sections I and II, I review the existing literature on the profitability and effectiveness of technical indicators and chart pattern analysis. Then, I examine the usage of machine learning for stock selection and the profitability of machine learning algorithms in finance in section III. In section IV, I introduce the dilemma of feature selection and review the current literature. Lastly, I discuss my contribution to the field in section V.

II Profitability of Technical Indicators

I Early Studies (1962-1992)

Early studies from 1966-1992 investigate the profitability of technical indicators like filters, moving averages, and relative strength indices. Fama and Blume (1962) analyze the profitability of a simple filter rule on the daily closing prices of thirty stocks in the Dow Jones Industrial Average (DJIA) from 1956-1962 [24]. For reference, a filter rule, as introduced by Alexander (1961), is a technical indicator that generates trading signals based on percentage changes from prior prices [3]. Filters are considered a momentum indicator because they assume that rising prices should continue to rise and that falling prices should continue to fall. Fama and Blume find that only three small filters (0.5%, 1.0%, and 1.5%) generated returns in excess to a buy-and-hold strategy on the DJIA. They conclude that these filters would not be profitable once transaction costs and the idle times of funds invested were taken into account.

Similarly, Van Horne and Parker (1967) examine the profitability of a 200-day moving-average decision rule on a collection of thirty random stocks from the New York Stock Exchange (NYSE) over 1960-1966. They find that only five of the thirty stocks produced excess returns using the trading rule in comparison to the buy-and-hold strategy. On the whole, the buy-and-hold strategy greatly outperforms the moving-average decision rule even before transaction costs. Moreover, James and Bennington (1970) conduct a similar experiment using a relative strength trading rule with every stock on the NYSE from 1926-1966. They find that once transaction costs and risk is accounted for, the buy-and-hold portfolios outperform the trading rule on average.

While these previous studies align with the efficient market hypothesis, other studies at the time found evidence to support the profitability of trading rules. Leuthold (1972) applies 6 filter rules ranging from 1% to 10% to cattle futures traded on the Chicago Mercantile Exchange from 1964-1970 [45]. He finds that all of the filters generated positive returns and four of the six filters led to a profit after transaction costs were factored in. Moreover, Sweeney (1986) applies filter rules ranging from 0.5% to 10% to the US Dollar to German Mark foreign exchange market from 1975 to 1980 [68]. Sweeney finds statistically-significant risk-adjusted returns across the different filters and concludes that filters can be profitable.

The inconclusive findings of early studies can be attributed to several inconsistencies with the testing procedures. First, there exists an ex-post rule selection/ data snooping problem because studies only consider a couple trading systems over a relatively short period of time. This is in part due to the lack of computational power available at the time, but it also could be attributed to researchers cherry-picking results to support a certain theory. Secondly, these studies do not calculate the statistical significance of the returns. Thus, the returns could be the product of a sample that is not representative of the true population. Lastly, the majority of studies during the period fail to adjust for risk. Since investors are inherently risk-averse, this is a large omission that could impact the conclusions of certain studies.

II Modern Studies (1992-Present)

The next grouping of studies are those that apply a bootstrapping procedure to evaluate the statistical significance of trading profits. This model-based bootstrap methodology was pioneered by Brock (1992) in one of the most influential works regarding the profitability of technical analysis [12].

Brock, *et al.* (1992) introduce their bootstrapping methodology to evaluate statistical significance because conventional t-tests are not well-suited to analyze financial returns [12]. T-tests assume that data is normally distributed and homoscedastic whereas stock returns have been shown to have non-normal distributions that are leptokurtic, autocorrelated, conditionally heteroskedastic, and time varying. The model-based bootstrap procedure compares the returns conditional on buy (sell) signals from the actual market with a simulated return series generated using one of four null models: a random walk with drift, an auto regressive process of order one (AR (1)), a generalized autoregressive conditional heteroskedasticity in-mean model (GARCH-M) and an exponential GARCH (EGARCH). Brock, *et al.* take 500 bootstrapped samples of prices for each null model and then apply the technical trading rules to each of the 500 samples. The trading returns generated by the four null models can be estimated from the 500 samples and then compared to the returns from the real market. Brock *et al.* use this methodology to test two of the simplest and most popular technical trading systems: a moving average oscillator and a trading range break-out. In order to avoid data snooping, Brock, *et al.* use the entire Dow Jones Industrial Average (DJIA) from 1897-1986. They find that the buy (sell) signals generate positive (negative) returns across **all** of the technical rules and outperform the buy-and-hold strategy. While the results are statistically significant, Brock *et al.* do not report the transaction costs nor comment on the economic significance of their results. They conclude that "it is quite possible that technical rules pick up some of the hidden patterns [of stock price data]." This study is influential because the findings are consistent across all technical rules and provide strong support for the effectiveness of technical analysis.

Given the profound impact of Brock *et al.* on the literature, many studies sought to replicate their results with different data. Bessembinder and Chan (1995) use the bootstrap methodology with over 60 technical indicators on emerging Asian markets in Hong Kong, Japan, Malaysia, Taiwan, and Thailand from 1972-1989 [10]. They confirm that technical indicators can be economically profitable on these markets even after transaction costs are accounted for. Similarly, Chen and Metghalchi (2011) find that moving averages, relative strength indices, and moving average convergence divergence trading rules were profitable on the Danish stock market's OMXC20 index from 1993-2010 [15]. Anghel (2015) used the bootstrap methodology to examine the efficiency in stock markets of 75 countries around the world [4]. He applies the moving average convergence divergence (MACD) technical indicator to data from 2001 to 2012 and finds that the MACD indicator generated abnormal cost and risk adjusted returns on about half of the markets around the world. I contend that this presents a strong case for the effectiveness of technical indicators because the MACD indicator is one of the most popular indicators and is used so often that it might be priced into the stock's value. Lastly, Kwon and Kish (2002) extend Brock *et al.*'s model-based bootstrap procedure to the New York Stock Exchange (NYSE) and NASDAQ while using the same technical trading rules [43]. They also introduce a moving average indicator for volume and use slightly different null models [1]. They conclude that technical trading strategies can capture additional profit opportunities when compared to a buy-and-hold strategy.

Other studies use the bootstrap methodology of Brock, *et al.* with slight modifications to address the criticisms of the original paper. Critics claimed that Brock, *et al.* hand-picked technical trading rules that succeeded over their testing period. To circumvent this, Neely *et al.* (1997) use a genetic programming approach to find the optimal rules in an out-of-sample period before applying these rules to the training period [54]. They find that their out-of-sample, optimized trading rules produced returns in excess of the buy-and-hold strategy on foreign exchange markets from 1975-1980. These profits remain significant after accounting for the risk premium and transaction costs. Moreover, Bessembinder and Chan (1998) did not believe that the findings of Brock, *et al.* were reflective of the actual stock market because the price data was not adjusted for dividends [11]. They believe that the dividend-adjusted data will reduce the returns on short sales and, thus, reduce the returns on trading strategies. Thus, they opt to use the bootstrap methodology on dividend-adjusted DJIA data from 1926-1990. Using the same trading rules, Bessembinder and Chan conclude that technical indicators have predictive powers but are not necessarily profitable after break-even transaction costs. This is an important distinction because this does not mean that there exists a market inefficiency. In short, the consensus of the literature suggests that technical indicators can produce statistically significant returns, although there does exist some debate. This can be attributed to the complexity of transaction costs, as brokerage fees and time-specific premiums must be taken into account. Many of the previous studies on both sides of the spectrum explicitly acknowledge the difficulty of estimating transaction costs so they used imperfect approximations to draw conclusions. Also, the debate surrounding the profitability of technical indicators aligns with the AMH because the AMH contends that sometimes markets are efficient, thus rendering TA powerless. For a more comprehensive review of the literature see the following literature reviews: [53], [57].

[1] They use an ARIMA model in place of the GARCH model.

III Profitability of Chart Pattern Analysis

While technical indicators received the majority of academic focus, Osler and Chang (1995) were among the first researchers to examine the profitability of chart patterns [56]. In their seminal 1995 paper, they focus on the (inverse) head-and-shoulders, a chart pattern considered to be "one of the most common and, by all odds, the most reliable of the major reversal patterns" [22]. Osler and Chang use an objective, automated algorithm to identify the (inverse) head-and-shoulder pattern on the charts of daily exchange rates of six major currencies against the U.S. dollar during the floating rate period from March 1973 to June 1994. In order to avoid ex-post selection bias, six of the largest currencies were used including the German mark, Chinese yen, Canadian dollar, Swiss franc, French franc, and British pound. Once the chart pattern is identified, profits were calculated based on a market participant acting upon this information. Using BLL's bootstrapping methodology and a random walk null model, they find that the head-and-shoulders chart pattern produced statistically significant profits for the German mark and Chinese yen, but not for the other four currencies. However, they note that if an investor had speculated on all six currencies, the profits would have been both economically and statistically significant. They conclude that the (inverse) head-and-shoulders chart pattern has predictive power and could indicate the presence of market inefficiencies in the foreign exchange market.

With the exception of Osler and Chang's algorithmic approach to identifying head-and-shoulders chart patterns, computer algorithms were unable to extract more complex chart patterns from price charts in a consistent manner. Up until Lo, Mamaysky, and Wang (2000), the presence of white noise and random stock market fluctuations, while quickly discarded by the human eye, flummoxed computer software. Lo, *et al.* bridge this gap with their systematic and replicable approach to chart pattern analysis [48]. First, they employ a nonparametric kernel regression, a type of smoothing estimator, to average out the white noise. Now that the random fluctuations in price have been reduced, Lo, *et al.* define ten of the most popular technical patterns algebraically as a sequence of extrema. These ten patterns include (inverse) head-and-shoulders, broadening tops (bottoms), triangle tops (bottoms), rectangle tops (bottoms), and double tops (bottoms).[2] For example, consider their definition of head-and-shoulders in comparison to the visual on page 11.

$$\mathrm{HS} = \begin{cases} \mathrm{E}_1 \text{ is a local maximum} \\ \mathrm{E}_3 > E_1,\ E_3 > E_5 \\ \mathrm{E}_1,\ E_5 \text{ are within 1.5\% of their average} \\ \mathrm{E}_2,\ E_4 \text{ are within 1.5\% of their average} \end{cases}$$

In order to confirm the accuracy of their identification algorithm, they analyze one security, CTX, from 1992 to 1996 and compare the patterns identified by their algorithm with those found by professional traders. While not empirical evidence, the algorithm performed in a consistent manner with the professional traders and even picked up some patterns missed by the human eye. Then, Lo *et al.* test the profitability of their algorithm using the daily returns of securities on the NYSE and NASDAQ stocks from 1962 to 1996. They randomly select 10 stocks from each of the five market capitalization quintiles and use a Kolmogorov-Smirnov test to analyze the significance of their profits. They find that three of the ten patterns were significant on the NYSE while all of the ten patterns were significant on the NASDAQ. They conclude that chart patterns can provide valuable information and that "technical analysis can add value to the investment process." The economic and statistical significance of chart pattern analysis is supported by numerous other studies

[2] Given the success of this study, nearly all chart patterns have been expressed as a series of extrema. For 370 examples, see Thomas Bulkowski's *Encyclopedia of Chart Patterns* [13].

including Tsinaslanidis *et al.* on 560 NYSE stocks [69] and Masteika *et al.* on automated electronic futures exchanges [50].

Chart pattern analysis is currently experiencing a renaissance due to two recent technological innovations. First, the integration of complex event processing (CEP) into quantitative finance has enabled chart pattern analysis to be automated and conducted in real time. CEP is a type of software that can process large volumes of incoming data, conduct real time analysis, and then implement a response almost instantaneously [2]. The application of CEP to chart pattern analysis is clear: it is a tool that can quickly identify local extrema, classify chart patterns, and then execute trades in real time. Bandera *et al.* (2015) implement the chart pattern formulas described by Lo *et al.* into a CEP framework and test the accuracy and speed of the model on the closing price of the Barric Gold Corporation (NYSE) from 1995-2014. They find that their open-source CEP toolkit identifies patterns with 96% accuracy and a minimal 20 data point delay [9]. These findings are particularly impressive given the stock market volatility captured from 1995 to 2014 (dot-com bubble, 2007-2009 Great Recession) and a 20 data point delay is only a few milliseconds when working with high frequency data. Secondly, hedge funds have recently embraced CEP and similar AI technology in an attempt to expedite the execution of their trades [63]. In the world of high frequency trading, a few second delay could cost millions of dollars. In short, the emergence of CEP technology and the rush to AI in the hedge fund industry have revolutionized chart pattern analysis and made it into a legitimate real-time investment tool. While CEP technology is beyond the scope of this study, it is important to note that chart pattern analysis can be fully automated and traded real time like technical indicators.

IV Machine Learning in Finance

I Introduction

In a 1970 interview, Gordon Moore, the founder of the multinational technology firm Intel, predicted that the processing power of computers would double every two years. Moore's prediction, now formalized under Moore's law, has proven true; modern-day mobile phones have more computational power than the entire Department of Defense did in the 1980s [44]. One of the most important tools to emerge from this technological revolution is the field of machine learning. A subset of artificial intelligence, machine learning algorithms build mathematical models and make decisions without being explicitly programmed to do so. Machine learning is omnipresent in our daily lives; for example, consider web search engines, spam filters for email, and friend recommendations on social media.

Machine learning algorithms (MLAs) have been used extensively in quantitative finance because they are well-suited for risk-return optimization problems with lots of data. Practitioners cite two important properties of MLAs to explain their dominance in the industry. First, they can detect complex, nonlinear patterns and hidden relationships within data that were previously undetected by regression analysis. Additionally, MLAs can perform effectively in the presence of collinear variables. This property is particularly useful for technical analysis because technical indicators tend to move together with a substantial amount of correlation. Whereas regression analysis cannot predict the value of the dependent variable in the presence of collinearity, MLAs can use this correlation to make better-informed estimates.

In the early 1990s, artificial neural networks and genetic algorithms were quite popular on Wall Street due to their (relatively) low computational requirements. However, the dot-com bubble in the early 2000s and the 2007-2008 financial crisis aroused a feeling of distrust for machine learning in finance. These crises made practitioners question the effectiveness and risk of these tools even though they were not at fault for the economic turmoil. **Figure 2.1** shows some interesting (anecdotal) evidence from Google Trends about the negative relationship between search popularity for "economic crisis" and "artificial neural networks" [33]. As one can see, search queries for "artificial neural network" declined sharply from 2006-2010 before regaining popularity. After the 2007 "quant quake"[3] and the 2008 Great Recession, machine learning made

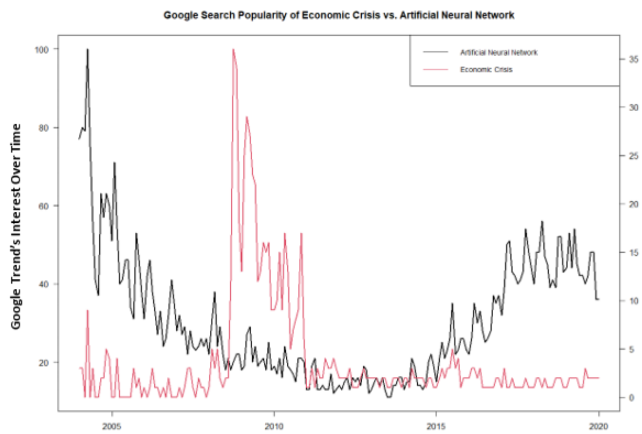


Figure 2.1: Google Trends for "Economic Crisis" and "Artificial Neural Network" from 2004 to 2020

a gradual come back. Given the technological and computational advancements during the early 2000s, MLAs were now much more powerful and easier to implement than before. As a result, MLAs have become ubiquitous on Wall Street and have found numerous applications ranging from portfolio optimization to stock selection. For the purpose of this study, I focus on two of the main applications of MLAs in finance: stock selection and feature engineering.

II Machine Learning for Stock Selection

In the case of machine learning for stock selection, analysts input a variety of factors that might be correlated with future returns into a model and let the MLAs discover how the information can be assembled to forecast returns (or any specified metric). Model inputs can include macroeconomic factors, company fundamentals, technical indicators, chart patterns, or a mixture of the above. Essentially, MLAs combine many weak sources of information with white noise to produce a stronger, aggregate investment signal. MLAs do this by uncovering complex patterns and hidden relationships between factors that might have been previously missed by outdated linear regressions. Once the model has been trained on a subset of the data called the *training set*, the model is used to forecast returns on the *testing set*. In order to assess the accuracy of the model, the forecasted returns on the testing set are compared to the actual returns during that period. If the performance of the model is satisfactory, the MLA can then be employed to actively forecast future returns and generate trading signals. In the context of stock selection and prediction, it is important to

[3]A one week period in August 2007 when quantitative strategies suffered unexplainable, substantial losses before making a full recovery [7].

note that machine learning algorithms tend to perform better with classification than regression because the signal-to-noise ratio is relatively low for stock returns [61]. This means that MLAs are mostly used to predict categories like outperformer versus underperformer (relative to the market) instead of exact stock prices. In the following paragraphs, I review the literature on machine learning for stock selection. I introduce the most popular machine learning algorithms and evaluate their benefits and drawbacks through the use of academic papers. Then, I conclude by focusing on support vector machines (SVM) because this algorithm is most relevant to the rest of this study.

ARIMA Model

One of the simplest machine learning algorithms for time series forecasting is the AutoRegressive Integrated Moving Average (ARIMA) model. ARIMA models integrate time series lags and lagged forecast errors into a framework to predict future values. Under the assumption of stationary data and constant time-series autocorrelation over time, these models seek to separate the signal from the noise. Adebiyi and Adewumi (2014) apply an ARIMA model to stock data from the NYSE and Nigeria Stock Exchange from 1996-2010 [5]. They find that their model is an effective predictor of stock prices for a maximum of thirty days in advance. However, the error of the model increases substantially in this 30-day window. This is one of the well-known limitations of ARIMA models for stock forecasting; they are only effective for short-term prediction. In the case of N-step ahead forecasting, the performance of the model quickly deteriorates. Additionally, the model is predicated on constant variance over time which is an unreasonable assumption given the stock market's volatility [8]. Given these severe limitations, ARIMA models are rarely used in practice. Instead, the most popular MLAs include Random Forests (RF), Recurrent Neural Networks (RNN), Long Short-Term Memory models (LSTM), and Support Vector Machines (SVM). A comprehensive review of these MLAs is beyond the scope of this paper so I offer the following literature reviews for more information: [62], [70], [55].

Random Forests

Random forests (RF) are widely used for classification tasks and consist of an ensemble of independent, uncorrelated decision trees. A decision tree is a non-parametric map that employs simple decision rules generated from data features to predict the value of a target variable. Random forests consist of an aggregation of uncorrelated decision trees because "a large number of relatively uncorrelated models operating as a committee will outperform any of the individual constituent models" [40]. In the context of stock selection, random forest models can be used to classify whether a stock price will increase or decrease by using various inputs like technical indicators and chart patterns. The model generates many decision trees that examine different combinations of the inputs and decision rules to forecast the movement of the stock (**Figure 2.2**). Then, the RF model adopts the consensus opinion of the deci-

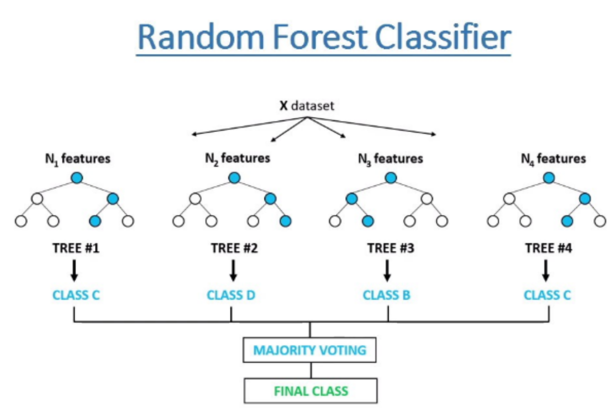


Figure 2.2: Random forests consist of uncorrelated decision trees and use a majority voting technique to reconcile the different outcomes [20].

sion trees as its forecast. The aggregation of many different models represents the main strength of RFs and one of its biggest drawbacks; RFs are computationally intensive and require a lot of time to train. Moreover, the performance of the RF model is highly sensitive to model parameters which lends itself to overfitting.

Patel, *et al.* (2014) implements four MLAs- an artificial neural network, support vector machine, random forest, and naive Bayes- to predict the stock price movement of two stocks- Reliance Industries and Infosys Limited- from 2003 to 2012 [58]. They calculate ten technical indicators including the RSI, MACD, and William's R%. Each of the 200 underlying decision trees are trained using three technical indicators to classify stock direction movement. They conclude that the RF model performs significantly better in terms of accuracy and F-measure than the other three MLAs for both of the stocks. However, I am skeptical of this conclusion for two reasons. First, the authors offer a small sample size of two stocks and do not justify why they choose the stocks they did. Thus, there could be something within the properties of these two stocks that lend themselves better to RF models. Secondly, the four MLAs are all sensitive to parameter inputs to some extent but the authors seem to select the parameters at will without an optimization process. Thus, the RF model might have received the best inputs resulting in the best performance. Nonetheless, this paper showcases that RFs can be used to generate significant economic profit using technical indicators.

Recurrent Neural Networks

In order to understand the strengths of recurrent neural networks (RNN), it is helpful to first consider artificial neural networks (ANN). ANNs, also known as feed-forward neural networks, have been used in quantitative finance since the 1980's. The universal approximation theorem states that ANNs are capable of approximating any continuous and bounded function by assigning weights that map any input to the output [65]. This property makes ANNs incredibly versatile and able to solve a wide variety of problems. However, there are two main limitations for ANNs in stock selection. First, the ANN backpropagation algorithm renders the ANN incapable of capturing sequential information as the weights and biases are adjusted in each cycle. This is troublesome because the ANN cannot account for seasonal trends that are quite influential in the stock market[4]. Secondly, while ANNs can capture complex relationships with a wide variety of interesting functions, they are susceptible to vanishing and exploding gradients [73]. For context, ANNs utilize backpropagation to assign weights to inputs, a two-step process that consists of propagating the input factors to the output layer through hidden layers and activation functions and then propagating backwards from the output layer to the input layer while computing error gradients. Once the error gradients for the weights and biases are calculated, the ANN updates the parameter values and takes a gradient descent step towards the minimum. If the gradients approach zero without converging, this is known as a vanishing gradient. Conversely, if the gradients keep getting larger and diverge, the gradient is said to be exploding. Vanishing and exploding gradients are problematic because they create an unstable neural network with misleading weights and biases. Thus, these networks cannot be used for classification, especially when large amounts of money are at stake.

[4]For example, stocks tend to rise before seasonal occasions like Thanksgiving, Christmas, and the Fourth of July [64].

RNNs improve upon the ANN backpropagation algorithm by using a looping constraint in the hidden layer (**Figure 2.3**). As a result, RNNs consider not only their original inputs but also what they have perceived in previous iterations. This allows RNNs to account for sequential data such as seasonal patterns between trading days, a powerful property for prediction in financial time series. They are also more computationally efficient than ANNs because RNNs employ a parameter sharing property that results in fewer parameters to train in each iteration. In turn, they are well-suited for high frequency trading applications with their reduced training time. However, RNNs with a large number of time series factors also suffer from vanishing and exploding gradients, a common issue across all the variations of neural networks. Despite this drawback, recurrent neural networks are widely used for stock prediction and selection problems.

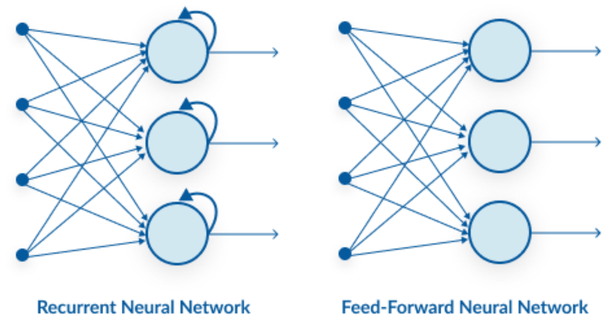


Figure 2.3: The looping constraint that enables RNN to process information sequentially and share parameters [72].

Chen, *et al.* (2018) incorporate sentiment analysis and technical indicators into a recurrent neural network framework to forecast the Shanghai-Shenzhen 300 stock index from 2015-2017 [18]. They analyze more than 800,000 posts from one hundred official accounts on Sina Weibo, China's largest online social network, as a proxy for the public mood. The technical indicators in the model include open/close/high/low price, volume, and daily percentage price and volume changes. Their RNN-boost model predicts the direction of the following day's price movement and achieves an impressive 70.17% prediction accuracy, outperforming the baseline ARIMA models. Chen *et al.* concludes that RNNs can generate economically significant profits even without a feature engineering algorithm.

Long Short-Term Memory Networks

Long short-term memory (LSTM) networks are a type of RNN that solves the vanishing and exploding gradient problem [37]. LSTM units[5] contain mechanisms called gates that are responsible for regulating the flow of information during model training (**Figure 2.4**). These gates are composed of different neural networks that carefully regulate the information that is used to update weights and biases. Consequently, the LSTM backpropagation stage is quite controlled through input and forget gates that prevent gradient divergence or convergence to zero. In addition to regulating the gradient descent, LSTM units enable better preservation of long-range dependencies through the use of memory cells. Mem-

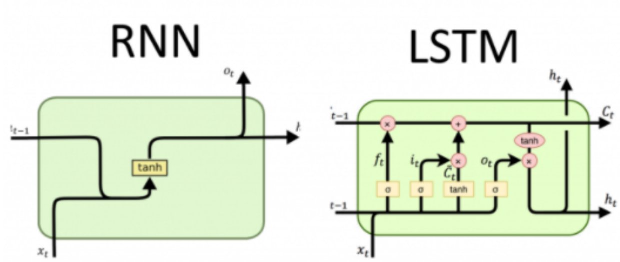


Figure 2.4: LSTM units are more complex than RNN cells and include gates that carefully regulate the training process [21].

[5]Comparable to cells in a classical neural network.

ory cells, located within LSTM units, can store information for long periods of time and reveal long-term relationships between factors. Thus, LSTM architecture represents a substantial improvement of the RNN without the gradient issues and longer-term memory. Even so, LSTM models can take a long time to train because the process is controlled with so many disjoint gates. Also, they are prone to overfitting given their increased memory storage. There does not exist a clear solution to either of these issues although current research suggests that dropout algorithms could prevent overfitting.

Fischer and Krauss (2018) deploy LSTM networks on the S&P 500 from 1992-2015 to predict out-of-sample directional price movements [31]. They find that the LSTM model produced economically significant profits (post-transaction costs) with an excellent Sharpe ratio of 5.8, indicating high returns with relatively low risk. Additionally, the LSTM model outperforms a random forest, deep neural network, and logistic regression over the same time period. Similarly, Siami-Namini, *et al.* compare the performance of a LSTM network and ARIMA model on monthly data from 1985-2018 on a variety of indices including the NASDAQ composite index, DJIA, and S&P 500 commodity price index [66]. The LSTM model acheived a lower Root-Mean-Square error (RMSE) than the ARIMA model on all ten indices considered. In fact, the LSTM model reduced the RMSE by an average of 85% which showcases the architectural superiority of the LSTM network over the ARIMA.

Support Vector Machines

As demonstrated, machine learning algorithms have been used extensively within quantitative finance to evaluate stocks. While each MLA has its benefits and drawbacks, recent literature endorses support vector machines as the best approach to stock selection. Support vector machine (SVM) is a machine learning algorithm that approaches classification problems from a geometric perspective. Each data item is plotted in an n-dimensional space (where n represents the number of inputs) with the value of each feature corresponding to a particular coordinate set. Then, the SVM fits a separation hyperplane to correctly classify the data within a specified margin using a decision function (**Figure 2.5**). SVMs are powerful because they are effective in high dimensional spaces with large quantities of inputs and memory efficient like LSTM models. This is particularly relevant for stock selection because there is an endless supply of inputs that can be used to explain asset prices. On the other hand, SVM performance can atrophy when the data set has a lot of noise. In order to remedy this issue, current research suggests the use of a feature selection algorithm to eliminate noisy input factors. A more comprehensive discussion of this issue is located in the Methodology section.

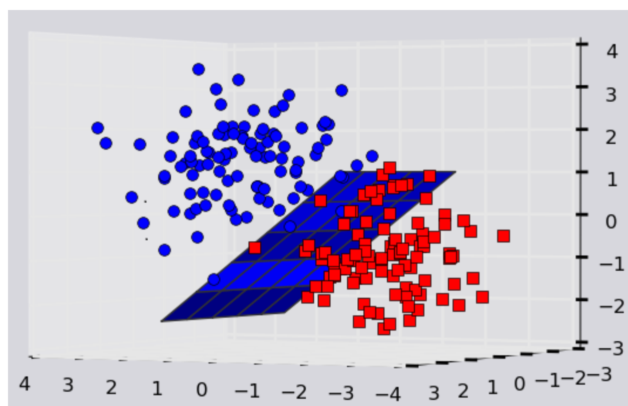


Figure 2.5: 3-dimensional space with red and blue points representing two different classes of objects within the data. The SVM models fits the blue separation hyperplane to classify the objects [59].

Fan and Palaniswami (2001) use a classification support vector machine in an attempt to generate returns in excess of the market [27]. They input fundamental accounting information and technical indicators of stocks trading on the Australian Stock Exchange from 1992-1995. The stocks predicted to increase by the SVM are assembled into an equally-weighted portfolio and assessed against a market benchmark from 1995-1999. The market benchmark returned 71.36% while the SVM portfolio accumulated an impressive 208% return over the same period. These strong results have been replicated by a number of other studies in different markets and time periods; Chicago Mercantile Exchange from 1988-1999 [14], NASDAQ and Shenzhen Stock Exchange from 2008-2010 [74], and Indian National Stock Exchange from 2009-2018 [52]. In short, SVM has been shown to be a very effective tool for stock selection.

III Machine Learning for Feature Selection

The second relevant application of machine learning is to the feature selection problem. With the recent explosion in our data production capabilities[6], analysts have access to more data than ever. As a result, there are millions of possible model inputs, a situation dubbed "the zoo of factors" by esteemed economist John Cochrane (2011) [19]. Furthermore, Harvey *et al.* (2016) found that more than three hundred factors have been proposed to explain asset returns in the last two decades [35]. Feature engineering seeks to determine which of these factors are most relevant in order to increase the signal-to-noise ratio and to avoid the computational costs that accompany high dimensionality. With respect to forecasting stock returns, this means that the selection of technical indicators and chart patterns as inputs into the model is arguably just as important as the parameters of the model. Thus, I devote the following paragraphs to reviewing different approaches to feature selection discussed in recent literature.

To demonstrate the importance of feature selection, Hwang and Rubesam (2019) hypothesize that the majority of the factor zoo is either redundant or the product of arbitrary data mining [38]. To test this theory, they construct a large database that contains 83 factors that might be correlated with asset returns ranging from technical indicators like 6-month momentum to nontraditional metrics like the number of analysts following a stock. They use a Bayesian approach to evaluate the efficacy of these different factors in explaining the monthly returns of all available U.S. common stocks from the CRSP and Compustat databases from 1980 to 2016. Their Bayesian variable selection process finds that only 10 factors (of the original 83) are ever selected and, of these ten, only eight are selected more than 50% of the time. The only factor that remained significant across all stocks is market excess return, defined as the stock return less the risk-free interest rate. The other selected factors include short-term reversal, change in 6-month momentum, earnings announcement return, change in the number of analysts covering stocks, industry concentration, unexpected quarterly earnings, and industry-adjusted size. Interestingly enough, short-term reversal and change in 6-month momentum are both technical indicators. These results align with previous research which suggest that the most effective models have five or less factors [25], [67], [17], [26].

On the other hand, Feng, Giglio, and Xiu (2020) evaluate the contribution of new factors introduced by asset pricing research from 2010 to 2016 in their award-winning paper titled "Taming the Factor Zoo" [28]. They gather 150 factors on a large set of standard portfolios of U.S. equities consisting of random stocks from the NYSE, AMEX, and NASDAQ. Then, they apply their model, an aggregation of recent econometric

[6] We produce more than 2.5 quintillion bytes of data every day. Also, more than 90% of data in existence has been produced in the past five years [41].

techniques ranging from Double-Selection to LASSO methodology, to determine the most impactful factors. They conclude that several newly developed factors are significant in explaining asset prices. They also find many more significant factors than previous studies, suggesting that asset prices cannot be explained by only five or six factors. This contradiction between studies is important because it demonstrates that the importance of feature selection.

Recursive feature elimination (RFE) is a popular, wrapper-style feature selection algorithm amongst the current literature. RFE searches for the most effective feature subset by building a model using all possible inputs and the desired machine learning algorithm. Then, RFE drops the weakest features, as ranked by the relative contribution of each of the inputs. This process repeats until the optimal number of features remain. In order to find the optimal number of features, cross-validation is used to evaluate the model accuracy of feature subsets of different sizes. RFE is widely-used because it is computationally efficient and not strongly influenced by the model's parameters. This property is important because it makes RFE performance consistent and not hypersensitive to parameters that can't be optimized. Gunduz (2021) implements a RFE algorithm to reduce the number of features before using a LTSM model to forecast the hourly directions of eight stocks in Borsa Instabul [34]. He finds that the RFE-LSTM model achieves the same model accuracy as the LSTM model without RFE using 20% less features. He concludes that RFE is a computationally efficient and effective method to feature selection with respect to stock selection.

V My Contribution

My contribution to the literature is two-fold. First, I integrate both technical indicators and chart pattern analysis into the same framework. Secondly, I use an original model that consists of a recursive feature elimination algorithm to carefully select model inputs before using an updating support vector machine to classify price direction. I hypothesize that this model can produce statistically- and economically-significant returns by only analyzing technical indicators and chart patterns.

Chapter 3

Methodology

I Overview

In this study, the daily price direction of 48 stocks are forecasted from 2010-2019 with a nonlinear, classification SVM. Using the R programming language, 33 technical indicators are calculated and 10 chart patterns are identified for each stock. A RFE algorithm is then implemented to identify the optimal subset of features for the SVM. Ultimately, a SVM classifies observations into two categories: stocks predicted to increase in price the following trading day and those predicted to decrease. Model performance is evaluated using the accuracy of the model's predictions and the returns of the underlying trading strategy. The following paragraphs delve into more detail on each step. **Figure 3.1** offers a visual depiction of the process. For more information on the training and testing periods, please refer to **Figure 3.8, 3.9**.

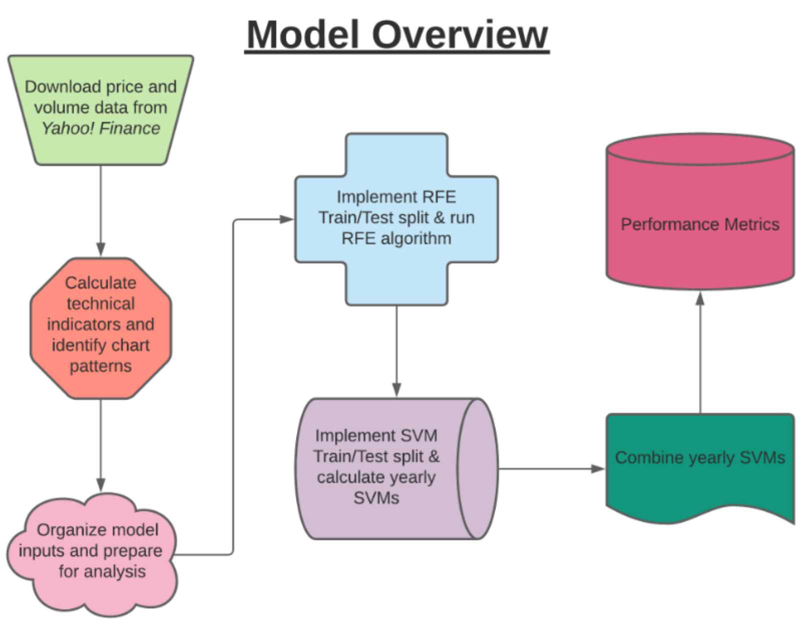


Figure 3.1: Flowchart of the model framework.

II Data

I collect price and information data for each of the 48 stocks from 1/1/2010 to 12/31/2019 with *Yahoo! Finance* [30]. Data is collected from 2010 through 2019 to capture the ebbs and flows of the economic cycle while avoiding the unprecedented market conditions caused by the 2008 Great Recession and the COVID-19 pandemic. The 48 stocks can be divided into three main groups. The first group consists of large-cap stocks and includes 29 of the 30 stocks[1] on the Dow 30 Index. The Dow 30 Index, also known as the Dow Jones Industrial Average (DJIA), is a price-weighted collection of thirty large, publicly-traded companies on the U.S. stock market. This index was developed to track the performance of the market in the late 1800s when information flow was limited [46]. Thus, the DJIA is selected as a holistic representation of the large-cap stocks on the market and to avoid data snooping. **Figure 3.2** lists the stocks on the DJIA.

Figure 3.2: List of the large-cap stocks from the Dow 30 index.

Large-Cap Stock List					
American Express (AXP)	Amgen (AMGN)	Apple (AAPL)	Boeing (BA)	Caterpillar (CAT)	Cisco Systems (CSCO)
Chevron (CVX)	Goldman Sachs (GS)	Home Depot (HD)	Honeywell International (HON)	International Business Machines (IBM)	Intel (INTC)
Johnson & Johnson (JNJ)	Coca-Cola (KO)	JPMorgan Chase (JPM)	McDonald's (MCD)	3M (MMM)	Merck (MRK)
Microsoft (MSFT)	Nike (NKE)	Procter & Gamble (PG)	Travelers Companies (TRV)	UnitedHealth Group (UNH)	Salesforce.com (CRM)
Verizon Communications (VZ)	Visa (V)	Walgreens Boots Alliance (WBA)	Walmart (WMT)	Walt Disney (DIS)	

The next group of stocks consists of 10 small-cap stocks from the VIOO index, Vanguad's S&P 600 Small-Cap ETF. In order to avoid selection bias, I choose the month-end top-10 holdings from VIOO index as of 12/31/21 [71] as long as the stocks were publicly traded from 2010-2020. **Figure 3.3** lists the small-cap stocks that were selected.

Figure 3.3: List of small-cap stocks from the VIOO index.

Small-Cap Stock List		
Agree Reality Corp. (ADC)	AMC Healthcare Services (AMN)	Balchem Corp. (BCPC)
Exponent Inc. (EXPO)	Chart Industries Inc. (GTLS)	Omnicell Inc. (OMCL)
Rogers Corp. (ROG)	UFP Industries (UFPI)	Vonage Holdings Corp. (VG)
	Watts Water Technologies (WTS)	

[1]Note that Dow Inc. is excluded because the company was founded on 4/1/2019 and does not possess the requisite price history.

The last group of stocks are the top-nine holdings of the Euro Stoxx 50 ETF as of 03/01/22 [29]. The Euro Stoxx 50 is considered to be Europe's prominent blue-chip index for the Eurozone and contains stocks from various stock exchanges throughout Europe including Paris, Brussels, and Germany. **Figure 3.4** lists the European stocks that were selected.

Figure 3.4: List of the European stocks.

European Stock List		
Anheuser-Busch InBev (ABI.BR)	ASML Holding (ASML.ML)	LVMH Moët Hennessy - Louis Vuitton (MC.PA)
L'Oréal S.A. (OR.PA)	Siemens Aktiengesellschaft (SIE.DE)	Sanofi (SAN.PA)
SAP SE (SAP.DE)	TotalEnergies SE (TTE.PA)	Volkswagen AG (VOW.DE)

I use three groups of stocks- a large-cap group, small-cap group, and European group- to investigate whether the model performs better with different types of stocks. I hypothesize that the model will perform best on the European stocks because European stock exchanges tend to have smaller market capitalizations than the U.S. markets and receive less attention from large institutional investors. On the other hand, I predict that the large-cap stocks will be the hardest to predict because they receive more scrutiny from big banks and hedge funds with large research departments. As a result, machine learning approaches might already by priced into the stocks.

III Software

The computational analysis is conducted in the R programming language [60].

IV Feature Engineering

In this section, I discuss the calculation of the 43 technical indicators and chart patterns used as inputs in the model. Definitions for the technical indicators come from the following sources: [16], [1]. Note that 25 data points are deleted at the beginning of the data set because certain indicators, like moving averages, require so many data points to be calculated.

I Technical Indicators

Price and Volume

The high, low, close, and adjusted daily price are included in the model. The adjusted daily price accounts for corporate actions like stock splits and buybacks that can manipulate the price of a stock. The daily trading volume is also included. This data comes directly from *Yahoo! Finance*.

Simple Moving Average

A simple moving average (SMA) is the average price or volume over a specified period and used to determine the direction of a trend. If the SMA is moving up (down), the trend is moving up (down). The five- and fifteen-day SMA for price and volume are calculated as well as the ratio of the two. Let P_n represent the daily price or trading volume of a stock on day n:

$$\begin{aligned}
SMA_{P_5} &= \frac{P_1 + P_2 + ... + P_5}{n} \\
SMA_{P_{15}} &= \frac{P_1 + P_2 + ... + P_{15}}{15} \\
SMA_{P_{Ratio}} &= \frac{SMA_{P_5}}{SMA_{P_{15}}} \\
SMA_{V_5} &= \frac{V_1 + V_2 + ... + V_5}{n} \\
SMA_{V_{15}} &= \frac{V_1 + V_2 + ... + V_{15}}{15} \\
SMA_{V_{Ratio}} &= \frac{SMA_{V_5}}{SMA_{V_{15}}}
\end{aligned}$$

Exponential Moving Average

An exponential moving average (EMA) is a moving average that places more weight on the most recent data points. Consequently, it reacts more significantly to recent price changes than the SMA. The five- and fifteen-day EMA for price is calculated as well as the ratio of the two. Let P_n represent the daily price on day n:

$$\begin{aligned}
EMA_5 &= P_n \cdot \left(\frac{1}{3}\right) + EMA_{(n-1)} \cdot \left(\frac{2}{3}\right) \qquad (1/3 = \text{smoothing multiplier}) \\
EMA_{15} &= P_n \cdot \left(\frac{1}{8}\right) + EMA_{(n-1)} \cdot \left(\frac{7}{8}\right) \\
EMA_{Ratio} &= \frac{EMA_5}{EMA_{15}}
\end{aligned}$$

Relative Strength Index

The relative strength index (RSI) is a momentum indicator that evaluates whether a stock is overbought or oversold by examining the magnitude of recent price changes. The RSI ranges from $(0, 100)$ where $RSI > 70$ indicates an overvalued and overbought condition and a $RSI < 30$ denotes an undervalued condition. The five- and fifteen-day RSI are calculated as well as the ratio of the two. There are two steps (S_1, S_2) to calculate the RSI_5. For simplicity, I only show the RSI_5 formula. The RSI_{15} can be calculated with the same formula substituting fifteen in place of the five.

$$\begin{aligned}
RSI_{S_1} &= 100 - \left[\frac{100}{\frac{\text{5-day average gain}}{\text{5-day average loss}}}\right] \\
RSI_{S_2} &= 100 - \left[\frac{100}{1 + \frac{(\text{5-day average gain}\cdot 13)+\text{Current gain}}{(\text{5-day average loss}\cdot 13)+\text{Current loss}}}\right] \\
RSI_{Ratio} &= \frac{RSI_5}{RSI_{15}}
\end{aligned}$$

Bollinger Bands

Bollinger Bands are used to measure the volatility of a stock by adding and subtracting two standard deviations from the 20-day SMA. Analysts can then examine the stock price in relation to the Bollinger Bands to generate trading signals. For example, if the price touches the upper Bollinger Band, the price is considered to be overbought triggering a sell signal. Let σ_{20} represent the standard deviation of price over the last twenty trading days:

$$\text{Upper Band} = SMA_{20} + 2 \cdot \sigma_{20}$$
$$\text{Lower Band} = SMA_{20} - 2 \cdot \sigma_{20}$$

Moving Average Convergence Divergence

Moving average convergence divergence (MACD) is a momentum indicator that analyzes the relationship between the a longer and shorter term EMA. Trading signals are generated as other moving averages cross the MACD. Other methods involving the MACD include crossovers, divergences, and sudden movements up (down). I calculate two MACDs with different parameters; $MACD_{DT}$ is optimized for day trading whereas $MACD_S$ is designed for longer-term trading horizons:

$$MACD_{DT} = EMA_d - EMA_t$$
$$MACD_S = EMA_m - EMA_n$$

Chaikan Volatility

Chaikan volatility (CV) measures the accumulation-distribution line of MACD. It measures the number of buyers/ sellers involved in the market in order to make a prediction about future price direction. I calculate the five- and fifteen-day CV and the ratio of the two. Here is the formula for the five-day CV where H, L, C represent high, low, close price:

$$S_1 = \frac{(C - L) - (H - C)}{H - L}$$
$$S_2 = S_1 \cdot \text{Volume}_5 \qquad \text{(Previous 5-day volume)}$$
$$S_3 = S_{2_{n-1}} + S_{2_n}$$
$$CV_5 = EMA_3(S_3) - EMA_{10}(S_3)$$
$$CV_{Ratio} = \frac{CV_5}{CV_{15}}$$

Rate of Change

Rate of Change (ROC) is a momentum indicator that measures the percentage change in price between today and the price a specified number of days ago. A positive (negative) ROC indicates an uptrend (downtrend) in price over the specified period. I calculate the five- and fifteen-day ROC as well as the ratio of the two.

Let P_n denote the current daily price and $P_{(n-x)}$ the price x days ago:

$$ROC_5 = \left(\frac{P_n - P_{(n-5)}}{P_{(n-5)}}\right) \cdot 100$$
$$ROC_{15} = \left(\frac{P_n - P_{(n-15)}}{P_{(n-15)}}\right) \cdot 100$$
$$ROC_{Ratio} = \frac{ROC_5}{ROC_{15}}$$

On-Balance Volume

On-balance volume (OBV) is a momentum indicator that adjusts stock price for volume flow. OBV is predicated on the assumption that volume is a key force behind market momentum. As OBV increases (decreases), an uptrend (downtrend) is occurring and a trend reversal might be looming in the near future. Let OBV_n represent the current OBV level, OBV_{n-1} the previous OBV level, P_n the current price, and $P_{(n-1)}$ the price yesterday:

$$OBV = OBV_{n-1} + \begin{cases} \text{volume}, & \text{if } P_n > P_{n-1} \\ 0, & \text{if } P_n = P_{n-1} \\ \text{-volume}, & \text{if } P_n < P_{n-1} \end{cases}$$

Ease of Movement

Ease of Movement (EMV) is a momentum and volume indicator that analyzes how easily a price can move up or down. It is also used to provide insight into the strength of an underlying trend by considering both price and volume. I calculate a five-day EMV using the following steps ($S_1, ..., S_4$). Let S = scale, a value ranging from 1,000 to 1,000,000 depending on the average trading volume of the underlying security, H (L) be the high (low) price, and PH (PL) be the prior high (prior low) within the past five days.

$$S_1 = \text{Distance Moved} = \left(\frac{H+L}{2} - \frac{PH+PL}{2}\right)$$
$$S_2 = \text{Box Ratio} = \frac{\text{Volume}/10000}{H-L} \qquad \text{(S=10,000)}$$
$$S_3 = \text{1-Period EMV} = \left(\frac{S_1}{S_2}\right)$$
$$S_4 = \text{14-Period EMV} = SMA_{14}(S_3)$$

Stochastic Oscillator

A stochastic oscillator (SO) is a momentum indicator that contextualizes a stock's current price within a range of its price over a specified period of time. It is used to identify overbought and oversold conditions and ranges from $(0, 100)$; $SO > 80$ signals overbought conditions while $SO < 20$ signals oversold conditions. There are a variety of parameters that can influence the trading signals produced by the SO. Thus, I calculate six distinct stochastic oscillators that offer unique information about the stock price. Let P_n represent the current price, HH_n be the highest high in the past n days, and LL_n be the lowest low in the same period:

$$\text{Stoch5\_FastK} = \left(\frac{P_n - LL_5}{HH_5 - LL_5} \cdot 100\right)$$
$$\text{Stoch5\_FastD} = SMA_3(\text{Stoch5\_FastK})$$
$$\text{Stoch5\_SlowD} = SMA_{15}(\text{Stoch5\_FastK})$$
$$\text{Stoch15\_FastK} = \left(\frac{P_n - LL_{15}}{HH_{15} - LL_{15}} \cdot 100\right)$$
$$\text{Stoch15\_FastD} = SMA_3(\text{Stoch15\_FastK})$$
$$\text{Stoch15\_SlowD} = SMA_{15}(\text{Stoch15\_FastK})$$

Williams Percent Range

Williams %R is a momentum indicator that compares a stock's price to the high-low range over a specified lookback period. In contrast to the SO, Williams %R looks at the closing price in relation to the highest high, not the lowest low. I calculate the five- and fifteen-day Williams %R and the ratio of the two. Let P_n represent the current price, HH_n be the highest high in the past n days, and LL_n be the lowest low in the same period:

$$\text{Williams }\%\text{R}_5 = \frac{HH_5 - P_n}{HH_5 - LL_5}$$
$$\text{Williams }\%\text{R}_{15} = \frac{HH_{15} - P_n}{HH_{15} - LL_{15}}$$
$$\text{Williams }\%\text{R}_{Ratio} = \frac{\text{Williams }\%\text{R}_5}{\text{Williams }\%\text{R}_{15}}$$

Average Directional Index

Average directional index (ADX) is a technical indicator that measures the strength of a price trend. I calculate the five- and fifteen-ADX as well as the ratio of the two. I show the calculation for the ADX_5

below. Let H_n (L_n) denote the current high (low) and H_{n-5} (L_{n-5}) the high (low) five days ago:

$$
\begin{aligned}
+DM &= H_n - H_{n-5} \\
-DM &= L_n - L_{n-5} \\
CDM &= \text{Current DM} \\
\text{Smoothed} + / - DM &= \sum_{t=1}^{14} DM - \left(\frac{\sum_{t=1}^{14} DM}{14} \right) + CDM \\
ATR &= \text{Average True Range} \\
+DI &= \left(\frac{\text{Smoothed } + DM}{ATR} \right) \cdot 100 \\
-DI &= \left(\frac{\text{Smoothed } - DM}{ATR} \right) \cdot 100 \\
DX &= \left(\frac{| + DI - -DI|}{| + DI + -DI|} \right) \cdot 100 \\
ADX &= \frac{\text{Prior}ADX \cdot 13 + \text{Current}ADX}{14} \\
ADX_{Ratio} &= \frac{ADX_5}{ADX_{15}}
\end{aligned}
$$

Vertical Horizontal Filter

A vertical horizontal filter (VHF) measures the strength of a trend by analyzing the relationship between various moving averages. I calculate the five- and fifteen-VHF as well as the ratio of the two. I show the calculation for the VHF_5 below. Let H_5 (L_5) represent the highest (lowest) closing price in 5 periods, P_n the current close price, and P_{n-1} yesterday's close:

$$
\begin{aligned}
VHF_5 &= \frac{(H_5 - L_5)}{\sum_{n=1}^{5} |P_n - P_{n-1}|} \\
VHF_{Ratio} &= \frac{VHF_5}{VHF_{15}}
\end{aligned}
$$

II Chart Patterns

There are two main approaches to locate technical patterns on stock price movement charts: template-based and rule-based matching. Template-based matching defines the shape of the query patterns visually and then uses point-to-point comparisons across price charts to identify patterns. Template matching was dominant in the 20^{th} century before technical analysts developed rule-based matching, a method that reduces chart patterns to a series of mathematical extrema [48]. By reducing patterns to familiar mathematical objects, analysts can make use of the tools of quantitative finance to make the pattern recognition process more reliable and efficient. I opt to use a rule-based matching approach in order to consistently identify chart patterns and give the model the most accurate possible inputs.

I use a R package called *RPatRec* to institute this rule-based matching approach [49]. *RPatRec* offers a couple of nice features that makes the process customizable and efficient. First, it includes a recognition function that allows users to define their own chart patterns by inputting a series of extrema. This allows me to define the following ten technical chart patterns. Secondly, the package offers a non-parametric kernel regression function (as described by Lo, Mamaysky, and Wang [48]) to smooth time series data by averaging out the white noise (**Figure 3.5**). Lastly, rule-based pattern matching is sensitive to a specified window size used to look for local extrema and correctly identify patterns. *RPatRec* allows the user to customize the length of the window and the number of data points in between in order to identify patterns at any scale of interest. For this study, I use ten years of stock data which equates to roughly 2500 price points. Therefore, I use a window size of 500 points with a 250 point overlap. The presence of a chart pattern is encoded in a binary variable where **1** signifies a pattern and **0** a lack thereof. Since I focus on ten of the most popular technical patterns, there exists ten chart pattern variables. While *RPatRec* provides inbuilt technical chart patterns, I update some of the mathematical formulas using Bulkowski's *Encyclopedia of Chart Patterns* [13]. Each chart pattern is defined as the following series of extrema. In order to visualize these definitions, refer to **Figure 1.3, 1.4**.

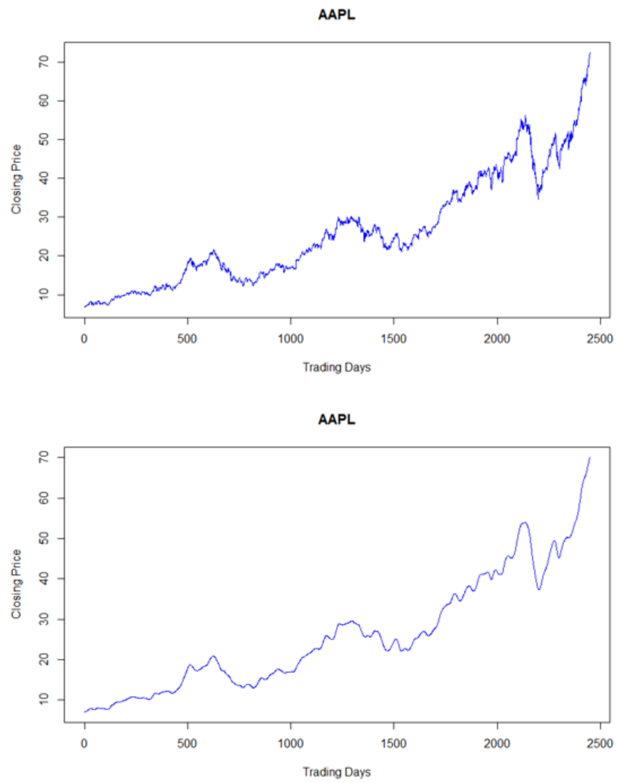


Figure 3.5: AAPL stock before and after the kernel regression smoother is used to reduce white noise.

Head and Shoulders

$$\text{Head and Shoulders} = \begin{cases} E_1 \text{ is a maximum} \\ E_3 > E_1, E_3 > E_5 \\ E_1 \text{ and } E_5 \text{ are within 2\% of their average} \\ E_2 \text{ and } E_4 \text{ are within 2\% of their average} \end{cases}$$

Inverse Head and Shoulders

$$\text{Inverse Head and Shoulders} = \begin{cases} E_1 \text{ is a minimum} \\ E_3 < E_1, E_3 < E_5 \\ E_1 \text{ and } E_5 \text{ are within 2\% of their average} \\ E_2 \text{ and } E_4 \text{ are within 2\% of their average} \end{cases}$$

Broadening Top

$$\text{Broadening Top} = \begin{cases} E_1 \text{ is a maximum} \\ E_1 < E_3 < E_5 \\ E_2 > E_4 \end{cases}$$

Broadening Bottom

$$\text{Broadening Bottom} = \begin{cases} E_1 \text{ is a minimum} \\ E_1 > E_3 > E_5 \\ E_2 < E_4 \end{cases}$$

Triangle Top

$$\text{Triangle Top} = \begin{cases} E_1 \text{ is a maximum} \\ E_1 > E_3 > E_5 \\ E_2 < E_4 \end{cases}$$

Triangle Bottom

$$\text{Triangle Bottom} = \begin{cases} E_1 \text{ is a minimum} \\ E_1 < E_3 < E_5 \\ E_2 > E_4 \end{cases}$$

Rectangle Top

$$\text{Rectangle Top} = \begin{cases} E_1 \text{ is a maximum} \\ \text{tops are within 1\% of their average} \\ \text{bottoms are within 1\% of their average} \\ \text{lowest top } > \text{ highest bottom} \end{cases}$$

Rectangle Bottom

$$\text{Rectangle Bottom} = \begin{cases} E_1 \text{ is a minimum} \\ \text{tops are within 1\% of their average} \\ \text{bottoms are within 1\% of their average} \\ \text{lowest top } > \text{ highest bottom} \end{cases}$$

Double Top

$$\text{Double Top} = \begin{cases} E_1 \text{ is a maximum} \\ E_1, E_3 \text{ are within 2\% of their average} \\ \text{At least 20 days must pass before consecutive tops} \end{cases}$$

Double Bottom

$$\text{Double Bottom} = \begin{cases} E_1 \text{ is a minimum} \\ E_1 < E_3 < E_5 \\ \text{At least 20 days must pass before consecutive bottoms} \end{cases}$$

Chart Pattern Detection

Figure 3.6 presents the results of the chart pattern detection algorithm. It appears that the head-and-shoulders and double bottom patterns were the most commonly identified patterns whereas zero rectangle tops and bottoms were detected. This makes sense because rectangle tops and bottoms have very strict definitions.

Chart Pattern Identification by Stock

Ticker	*HS*	*INVHS*	*BTOP*	*BBOT*	*TTOP*	*TBOT*	*RTOP*	*RBOT*	*DTOP*	*DBOT*
AAPL	4	4	2	1	0	1	0	0	2	6
AMGN	4	3	3	2	0	1	0	0	0	9
AXP	4	4	1	1	2	1	0	0	2	7
BA	4	3	2	1	0	2	0	0	2	5
CAT	4	2	0	0	1	1	0	0	3	4
CRM	4	4	1	2	0	1	0	0	1	8
CSCO	3	5	0	1	1	0	0	0	1	8
CVX	2	5	1	2	1	2	0	0	2	7
DIS	3	4	0	1	1	1	0	0	2	5
GS	4	4	3	0	0	2	0	0	3	5
HD	4	1	1	0	0	2	0	0	2	7
HON	6	3	1	0	1	2	0	0	1	7
IBM	4	4	2	1	0	1	0	0	5	4
INTC	0	6	1	0	0	2	0	0	0	4
JNJ	5	4	1	0	0	3	0	0	2	7
JPM	3	5	2	1	1	1	0	0	1	8
KO	6	2	2	1	1	0	0	0	4	5
MCD	3	6	1	2	0	2	0	0	2	7
MMM	3	5	1	0	0	0	0	0	2	7
MRK	6	3	2	0	0	2	0	0	0	9
MSFT	2	5	0	0	0	2	0	0	1	8
NKE	5	2	2	0	0	1	0	0	0	9
PG	3	4	3	0	0	3	0	0	2	7
TRV	3	5	3	0	0	2	0	0	1	8
UNH	5	1	2	0	2	2	0	0	1	7
V	2	2	1	0	0	2	0	0	1	7
VZ	4	5	1	0	1	4	0	0	1	7
WBA	3	4	1	0	1	2	0	0	4	5
WMT	4	5	1	0	1	2	0	0	1	7

Figure 3.6: The number of chart patterns detected in each stock from 2010-2019.

V Recursive Feature Selection Algorithm

With the 33 technical indicators and 10 chart patterns in the data set, I use RFE with cross-validation (CV) to identify the optimal factors for the SVM model. I use a random forest algorithm to fit the model in each iteration and to evaluate the performance of four different-sized feature subsets consisting of 4, 8, 16, and 40 factors. Then, I select the subset of features that optimizes the trade-off between dimensionality and the root-mean-square error (RMSE). **Figure 3.7** visualizes the results of the RFE-CV algorithm on Apple's stock (AAPL) in 2011. As one can see, the feature subsets have little predictive power with four features and then the RMSE drops with the addition of four more factors. Then, the RMSE gradually increases as noisy features are added to the model. In this case, eight features is the optimally-sized feature subset.

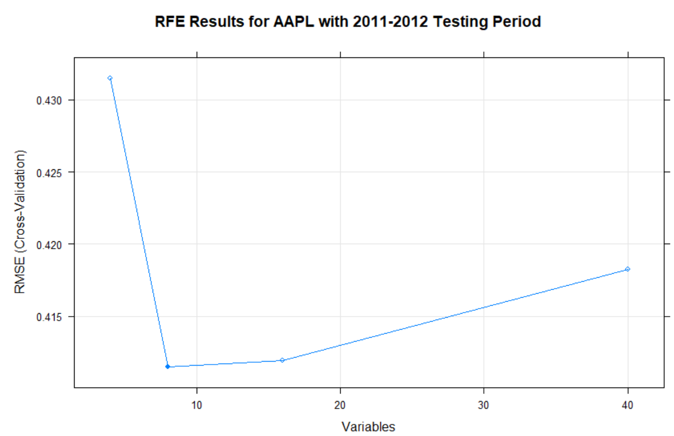


Figure 3.7: The results of the RFE algorithm on AAPL with a 2011-2012 testing period.

I recalculate the RFE for each stock and each testing year in order to get the most updated feature subsets. For example, consider AAPL stock from 2010-2019. The first testing period is from 2011-2012 because the model first needs to be trained on data from 2010-2011. For this first testing period, the RFE algorithm is implemented on data from 2010-2011. For the second testing period, 2012-2013, the feature subset is optimized from 2010-2012 in order to give the RFE algorithm more data. The RFE for the last testing period, 2018-2019, uses data from 2010-2018 to find the best feature subset. Thus, I expect that the model for the last testing period (2018-2019) will be more accurate than the first testing period model (2011-2012) because the RFE algorithm has more data to make better-informed decisions. **Figure 3.8** depicts the RFE training periods in terms of the test year. The graph can be interpreted as follows: For the 2015-2016 test year, the RFE is trained from 2010-2015.

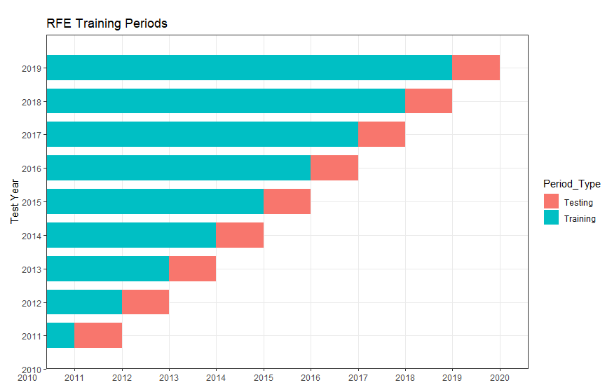


Figure 3.8: Visualization of the RFE train/test split.

VI Support Vector Machine

Once the optimal feature subset is selected, the data is passed to the SVM to classify the price direction for each observation. The data is split into a training/testing split that is constantly updating. For example, consider AAPL stock from 2010-2019. The first SVM is trained from 2010-2011 and then tested from 2011-2012. The second SVM is trained from 2011-2012 and tested from 2012-2013. **Figure 3.9** depicts the SVM train/test splits. I opt to have the SVM update every year in order to capture macroeconomic conditions and yearly trends that are omitted from the data set. The SVM model contains three parameters that must be specified. First, I specify the kernel to use a radial basis function (RBF). This allows the SVM to construct non-linear hyperplanes to separate the different classes of data. Secondly, I set the gamma value, or the kernel coefficient for the RBF, equal to $\frac{1}{5}$ which is relatively low in order to avoid over-fitting. Lastly, the cost parameter, or the penalty parameter of the error term, is defined as 3. These parameters are carefully selected based on the current literature in order to avoid over-fitting.

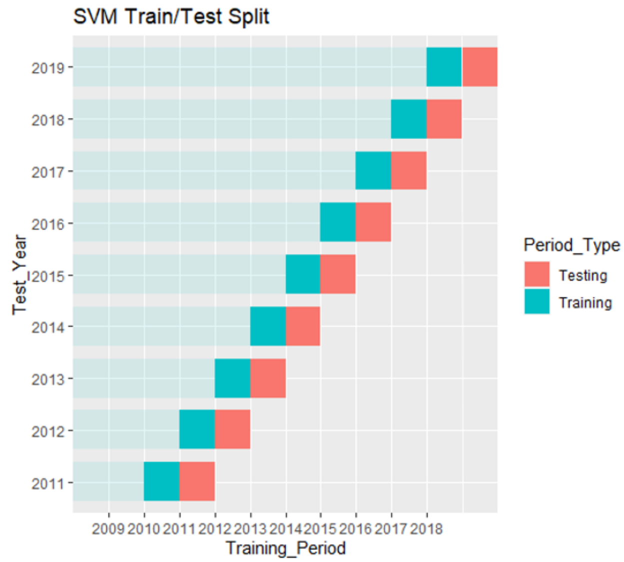


Figure 3.9: Visualization of the SVM train/test split.

The yearly models for each stock are combined such that each stock has daily price predictions from 2011-2019. Model performance is then assessed with two measures. The statistical accuracy of the SVM compares the predicted price direction with the actual movement of the price direction. The accuracy is defined by the number of correct predictions divided by the number of total predictions. The 95% confidence interval for the accuracy is also calculated. Model performance is also considered in light of the returns generated by the underlying trading strategy. The trading strategy is straight-forward. Assume that the SVM models are calculated every trading day right before market close. If a stock is predicted to increase (decrease) in price the following day, one share of the stock is bought (shorted) until the models are recalculated the next trading day. The ability of the model to hold long and short positions is advantageous because the algorithm can generate returns in both bull and bear markets. Stocks are then held (shorted) for consecutive days as long as they are predicted to increase (decrease) in value. I consider the returns of the strategy without accounting for transaction costs because they are typically assumed to be negligible.

I compare the returns from the RFE-SVM model to a buy-and-hold approach and a simple MACD strategy. The buy-and-hold strategy assumes that a share of the stock was purchased on 1/4/2011 and held through 12/31/2019. The MACD strategy generates a signal when the MACD line crosses over a signal line. The MACD line is defined as the difference between a 26-day simple moving average of closing price and a 12-day simple moving average. The signal line is a 9-day exponential moving average of the MACD signal. When the MACD line crosses above (below) the signal line, a long (short) position is acquired. The buy-and-hold approach serves as a passive investment baseline whereas the MACD strategy represents a basic technical approach to investment.

Chapter 4

Results

I Feature Selection

The results of the RFE algorithm for the 29 large-cap stocks[1] are displayed in **Figure 4.2** on the next page and arranged in descending order of popularity. The interpretation of the table is as follows. In the first row, on-balance volume (OBV) was included in the optimal subset of features for 15 of the 29 large-cap stocks when the RFE was trained on 2010-2011 data. When the RFE was trained on 2011-2012 data, OBV was selected in 14 of the 29 optimal subsets. The column on the far-right is the sum of the yearly selections. For example, OBV was selected in 135 of the 290 (29 stocks over 10 years), or $\approx 47\%$, possible subsets.

Figure 4.1 showcases the selection trends of the seven features with the most total selections (*left*) and the seven least popular features (*right*). Of the ten chart patterns, only four were selected in at least one cycle. The four chart patterns that were selected were among the five least popular features. The majority of chart pattern selections occurred in 2012, with no chart pattern being selected after 2015.

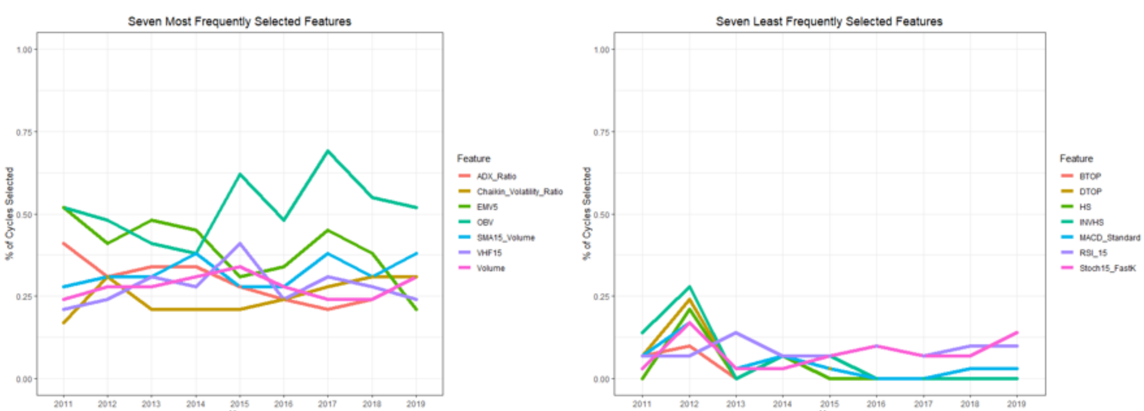


Figure 4.1: Trends over time of the seven most and least selected features.

[1]This section focuses on the RFE results for only the large-cap stocks due to computational limits.

Figure 4.2: Feature selection results

Feature Selection Results for Large-Cap Stocks

Feature	*2011*	*2012*	*2013*	*2014*	*2015*	*2016*	*2017*	*2018*	*2019*	*Total Selections*
OBV	15	14	12	11	18	14	20	16	15	135
EMV5	15	12	14	13	9	10	13	11	6	103
SMA15 Volume	8	9	9	11	8	8	11	9	11	84
ADX_Ratio	12	9	10	10	8	7	6	7	9	78
VHF15	6	7	9	8	12	7	9	8	7	73
Volume	7	8	8	9	10	8	7	7	9	73
Chaikin Volatility Ratio	5	9	6	6	6	7	8	9	9	65
WPR5	5	7	9	8	6	7	7	6	7	62
EMA15	10	7	6	4	4	7	8	7	7	60
ADX 15	1	4	6	7	6	9	9	10	7	59
Chaikin_Volatility15	4	7	9	7	9	7	5	6	5	59
RSI_Ratio	11	4	5	7	7	8	6	7	4	59
WPR Ratio	5	6	5	6	6	5	10	8	5	56
Chaikin Volatility5	4	3	9	7	5	8	6	7	6	55
Stoch5_FastK	3	2	5	5	7	7	9	7	7	52
MACD_DayTrade	4	5	5	8	5	8	5	6	5	51
ROC Ratio	8	5	9	4	8	3	5	5	4	51
ROC15	9	4	8	7	4	6	3	5	5	51
SMA_Volume_Ratio	6	3	5	4	6	5	5	6	11	51
BBand_Upper	7	7	5	2	4	6	5	5	7	48
Stoch5 FastD	1	4	8	7	6	7	4	5	3	45
EMA5	6	4	4	5	5	5	4	6	5	44
SMA_15	5	4	2	2	4	7	5	6	9	44
BBand_Lower	5	3	5	2	4	7	6	5	4	41
RSI_5	8	5	3	6	4	3	2	3	6	40
Stoch5 SlowD	4	4	4	6	5	5	4	3	4	39
WPR15	4	5	6	4	6	2	4	5	3	39
SMA5_Volume	3	1	2	4	6	4	7	6	4	37
ROC5	4	7	6	4	2	4	1	1	7	36
Stoch15 SlowD	6	2	2	3	5	2	3	7	6	36
SMA Ratio	5	5	4	3	2	6	5	3	2	35
SMA_5	5	4	3	1	3	7	4	4	3	34
ADX 5	2	3	5	4	4	6	3	2	4	33
Stoch15 FastD	4	4	6	4	6	1	3	3	2	33
VHF5	4	5	4	3	3	4	3	3	2	31
VHF_Ratio	2	0	3	5	3	3	6	4	4	30
EMA Ratio	2	0	1	4	3	2	3	4	6	25
RSI 15	2	2	4	2	2	3	2	3	3	23
Stoch15_FastK	1	5	1	1	2	3	2	2	4	21
INVHS	4	8	0	2	2	0	0	0	0	16
MACD Standard	2	5	1	2	1	0	0	1	1	13
DTOP	2	7	0	2	1	0	0	0	0	12
HS	0	6	0	2	0	0	0	0	0	8
BTOP	2	3	0	2	1	0	0	0	0	8

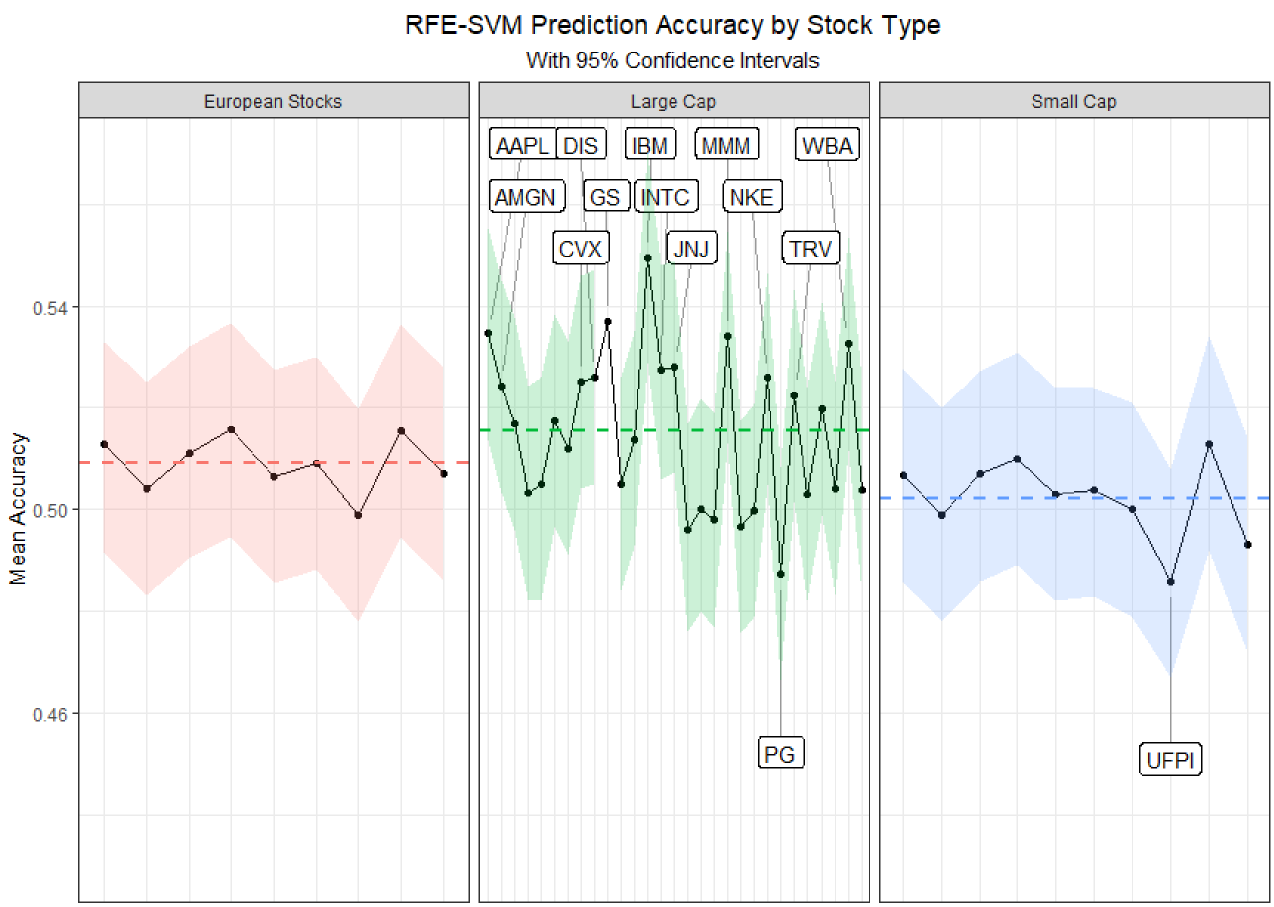
II Stock Selection

The results of the RFE-SVM algorithm are considered with respect to the statistical accuracy of the model as well as the economic returns generated by the signal.

I Statistical Accuracy

The statistical accuracy of the RFE-SVM model is defined by the number of correct price predictions divided by the total number of predictions. **Figure 4.3** displays the accuracy of the model for each stock and contains three panes, one for each of the three groupings of stocks. The shaded regions represent the 95% confidence interval for the model accuracy. Stocks that are labelled either have better than a 52% accuracy or below a 49% accuracy. The large-cap stocks had the highest average model accuracy with 51.5%, followed by European stocks with an average 50.9% accuracy and then the small-cap stocks with 50.2% accuracy. After confirming the homogeneity of variances with Levene's test and the normality of the data with a Shapiro-Wilk test, I used an ANOVA and a Tukey multiple pairwise-comparisons test to find a statistically significant difference between the model accuracy of the large-cap stocks and the small-cap stocks.

Figure 4.3: Statistical accuracy of the RFE-SVM model by stock type.



II Economic Returns

The following figures display the compound returns for the buy-and-hold strategy, the RFE-SVM model, and a simple MACD trading strategy. **Figure 4.4** shows the returns for large-cap stocks. The average return for the RFE-SVM model was 4.44% which outperformed the buy-and-hold approach with mean return 3.14% and the MACD signal with mean return -0.24%. The RFE-SVM model outperformed the buy-and-hold strategy for 10 of the 29 large-cap stocks.

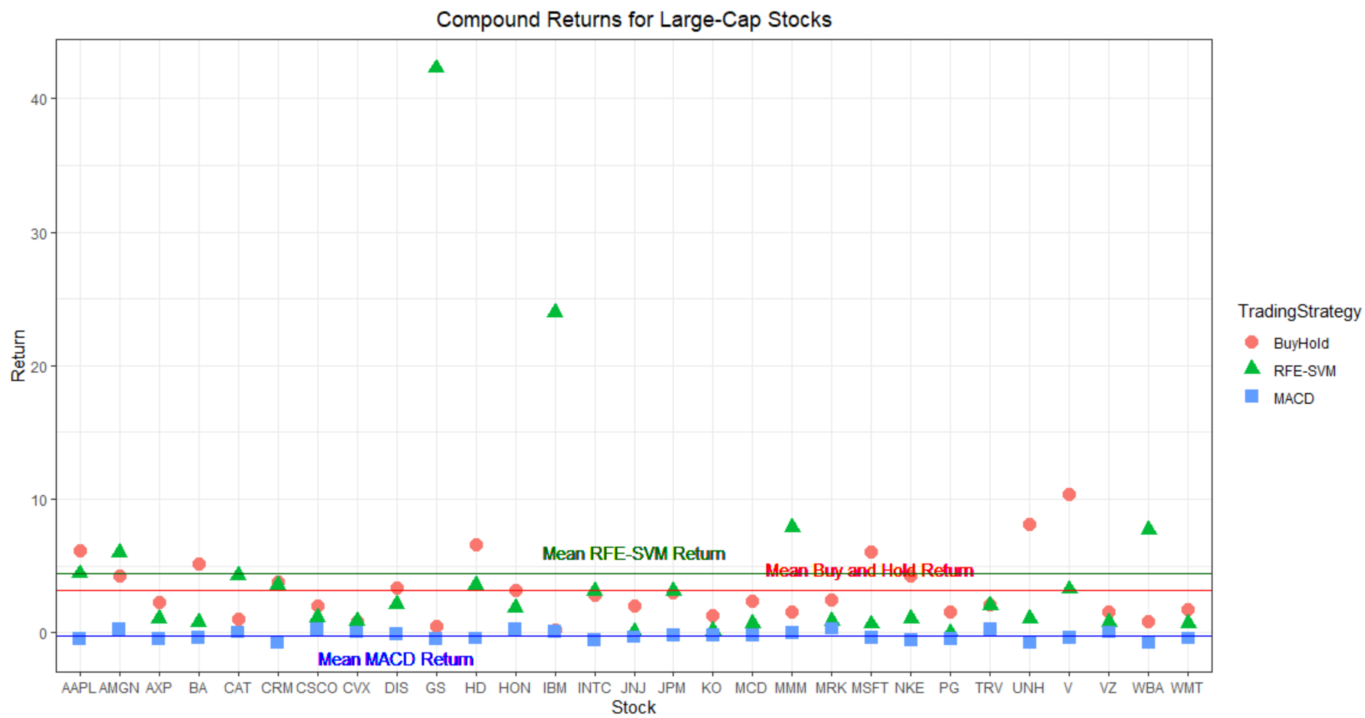


Figure 4.4: Returns of the large-cap stocks.

Figure 4.5 displays the returns for the small-cap stocks. The average return for the buy-and-hold strategy was 3.57% which outperformed the RFE-SVM model with mean return -0.81% and the MACD signal with mean return -0.29%. The RFE-SVM model did not outperform the buy-and-hold strategy for any of the 10 small-cap stocks. **Figure 4.6** displays the returns for the European stocks. The average return for the buy-and-hold strategy was 2.14% compared to the RFE-SVM model with mean return 0.02% and the MACD signal with mean return -0.45%. The RFE-SVM model did not outperform the buy-and-hold strategy for any of the nine European stocks.

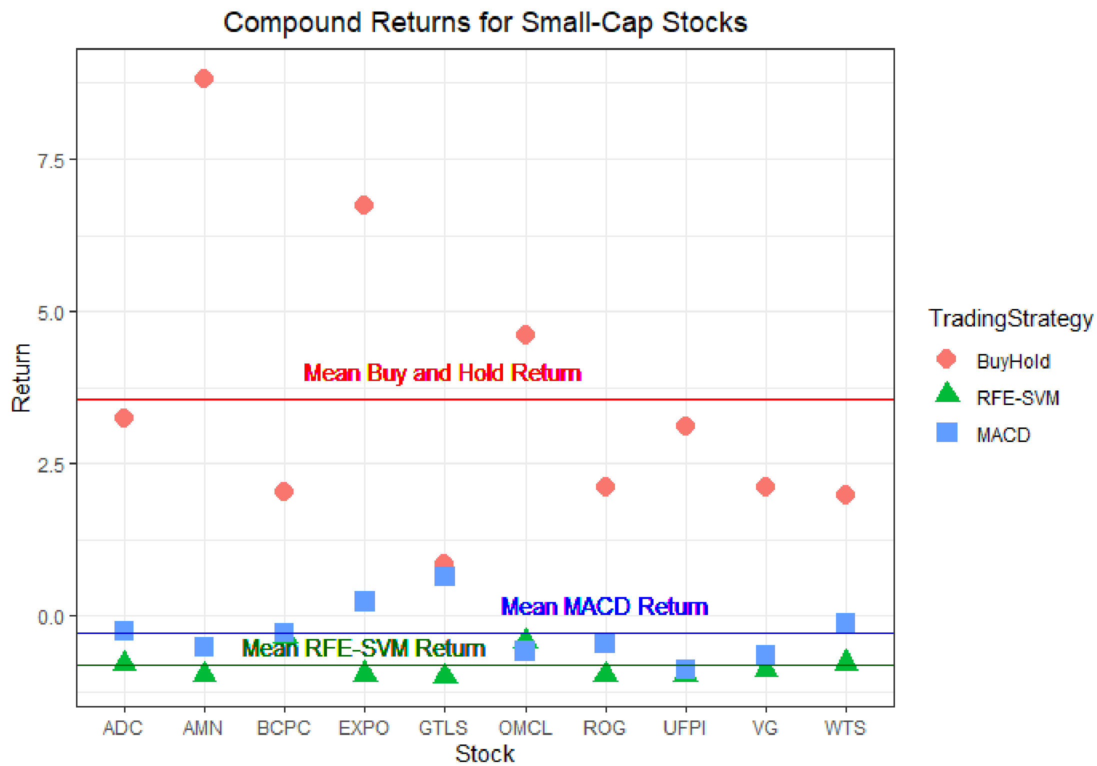


Figure 4.5: Returns of the small-cap stocks.

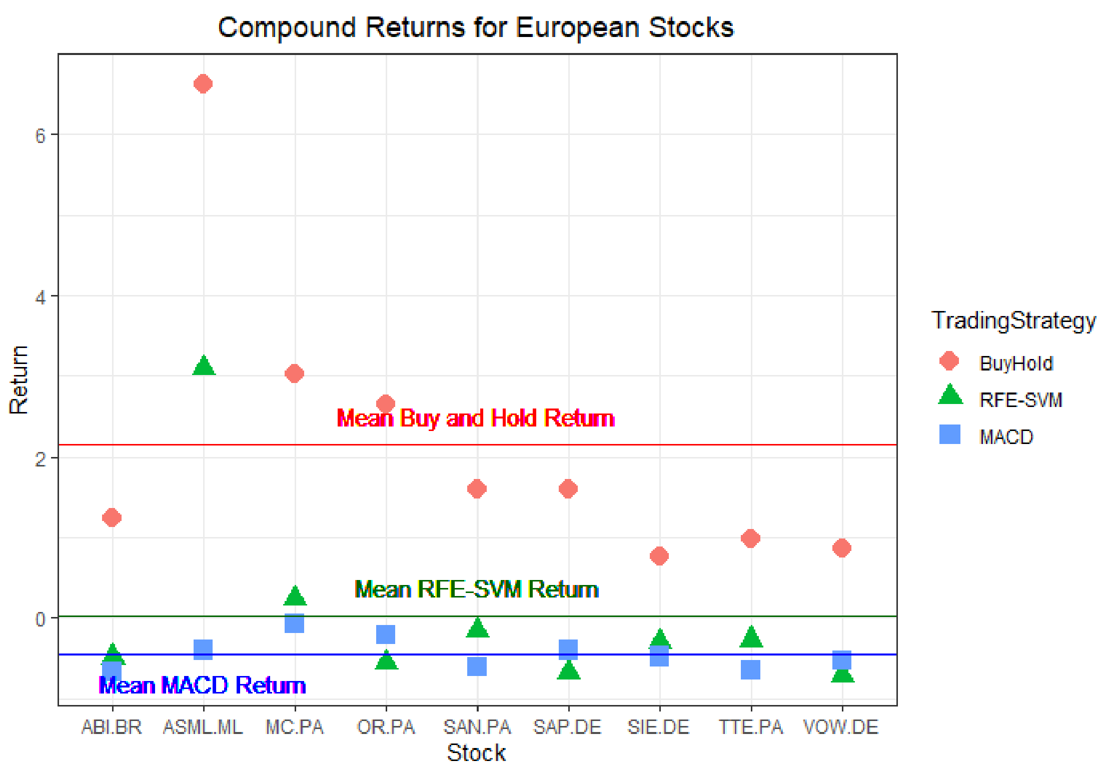


Figure 4.6: Returns of the European stocks.

Lastly, **Figure 4.7** displays the distributions of the returns for the different stocks. The three panes once again represent the three different groups of stocks. The dashed vertical line marks the mean return for each strategy in each stock group. The buy-and-hold strategy generated the highest average return for the small-cap stocks, followed by the large-cap stocks and then the European stocks. The RFE-SVM model was most effective on the large-cap stocks, but produced an average return of about zero for both the small-cap and European stocks. The standard deviations of the buy-and-hold distributions are the highest amongst the different strategies. The distributions of the MACD strategy is centered below zero for the three groups of stocks and clustered tightly around the median value. The distributions of the RFE-SVM model vary based on the type of stock but generally demonstrate more upside potential than the MACD strategy. Note that two returns for the RFE-SVM on large-cap stocks are omitted because they are outliers (GS- 42.3, IBM- 24.0). If you are interested in the individual cumulative return graphs for the 48 stocks, please reference the **Appendix**.

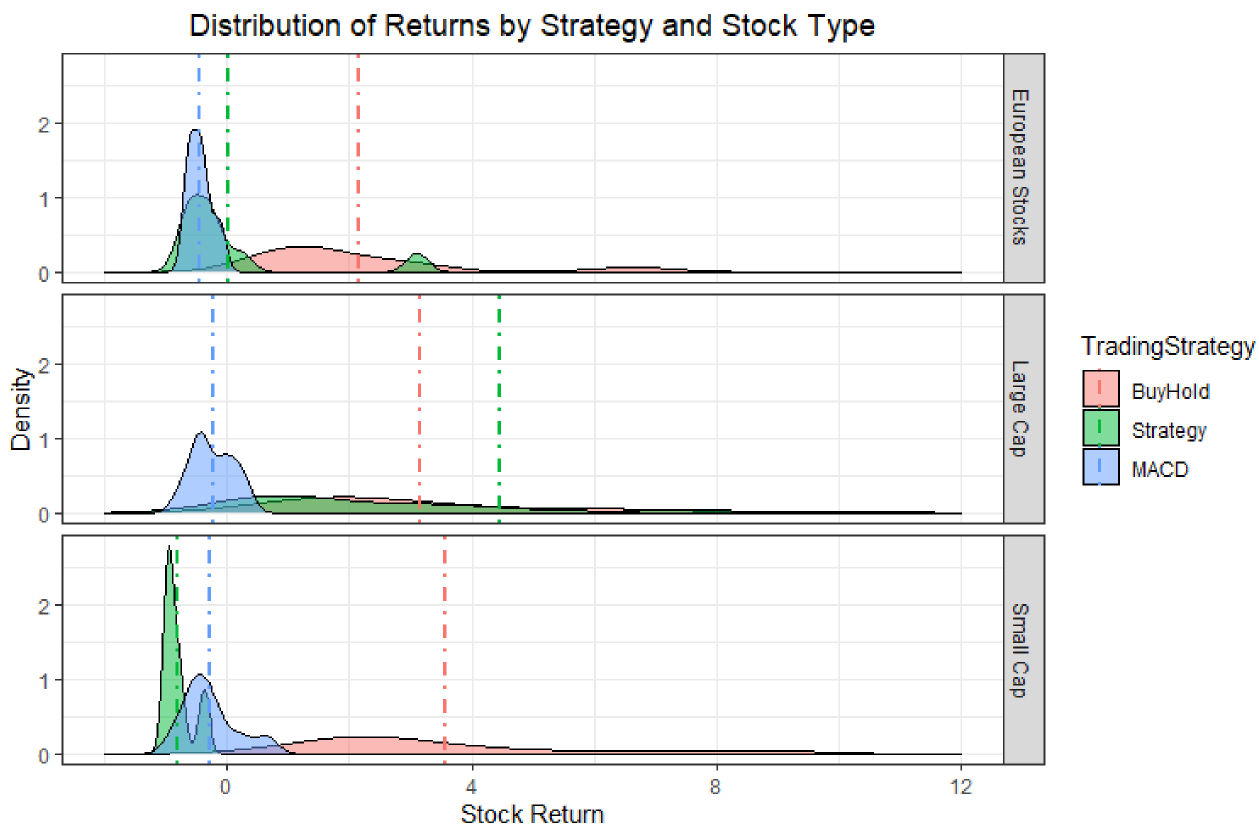


Figure 4.7: Density plots for the returns.

Chapter 5

Discussion

In light of the economic returns and predictive accuracy of the model, I reject the hypothesis that the RFE-SVM model could produce economically- and statistically-significant results using technical indicators and chart patterns. These results suggest that SVMs could not outperform a passive buy-and-hold strategy from 2010-2019. This aligns with the adaptive market hypothesis which states that technical indicators can only be used to generate a profit in periods of high market volatility where investors behave irrationally. The time period I selected for my analysis specifically omitted the high volatility conditions of the 2008 Great Recession and the COVID-19 pandemic, so these results make sense in context. The results also support the recent shift of funds away from active management funds to passive index funds. In 2010, actively managed funds and ETFs controlled 81% of the fund market whereas index mutual funds and index ETFs composed the other 19% [39]. In 2020, actively managed funds decreased in size to 60% while passive funds grew to control a 40% share of the fund market. This shift towards passive funds can most likely be explained by a recent crop of literature that shows that active funds can not consistently outperform passive portfolios in the long run.

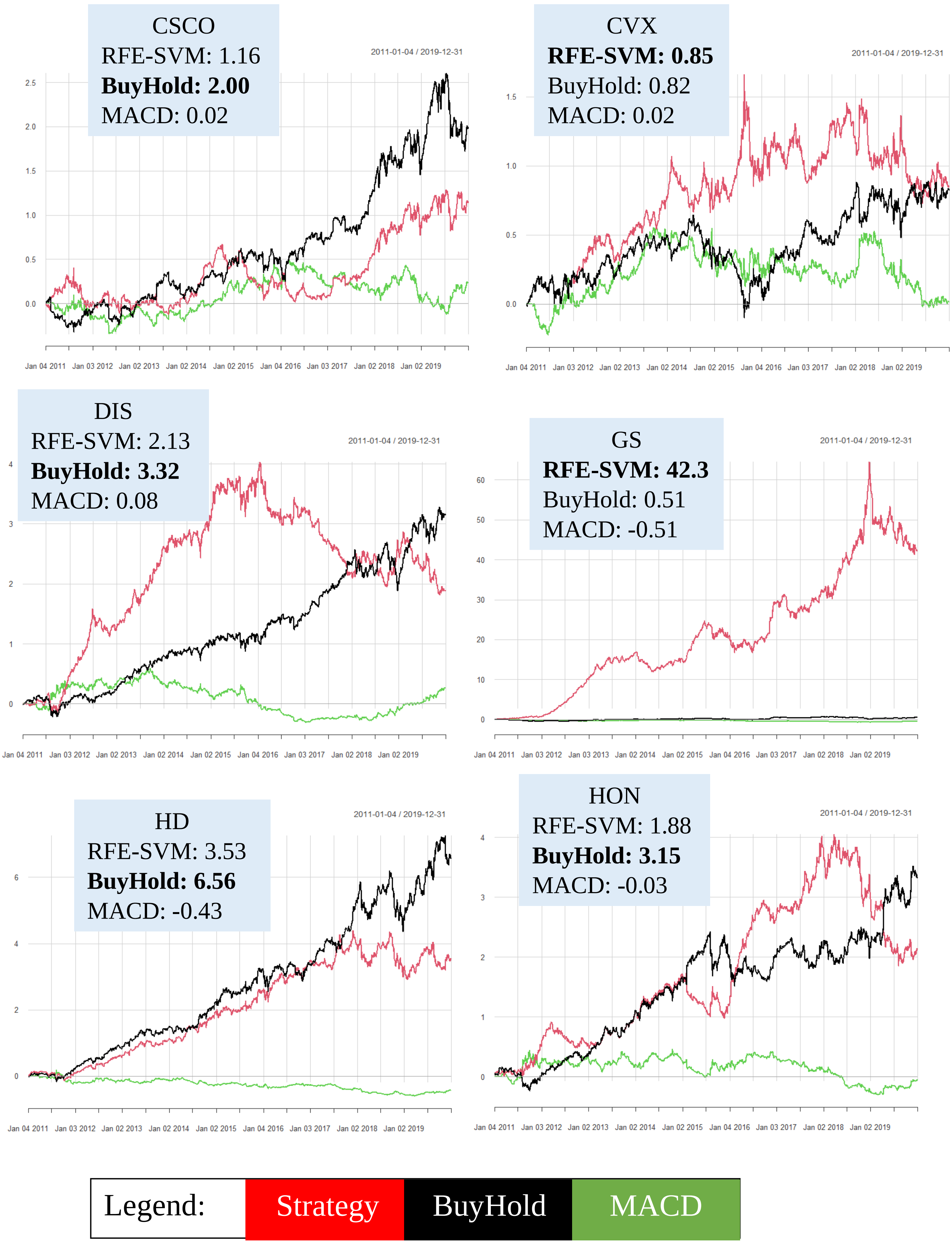
All in all, I believe that there still exists certain periods where MLAs can outperform passive strategies. In the future, I would like to continue my work on preemptively identifying periods of high market volatility. This is important because MLAs can be employed when periods of high market volatility are anticipated and then sidelined when volatility is forecasted to decrease. Additionally, I plan to run the RFE-SVM model on the same stocks during the 2008 Great Recession and the COVID-19 pandemic to investigate how this model performs in unprecedented and highly volatile market conditions.

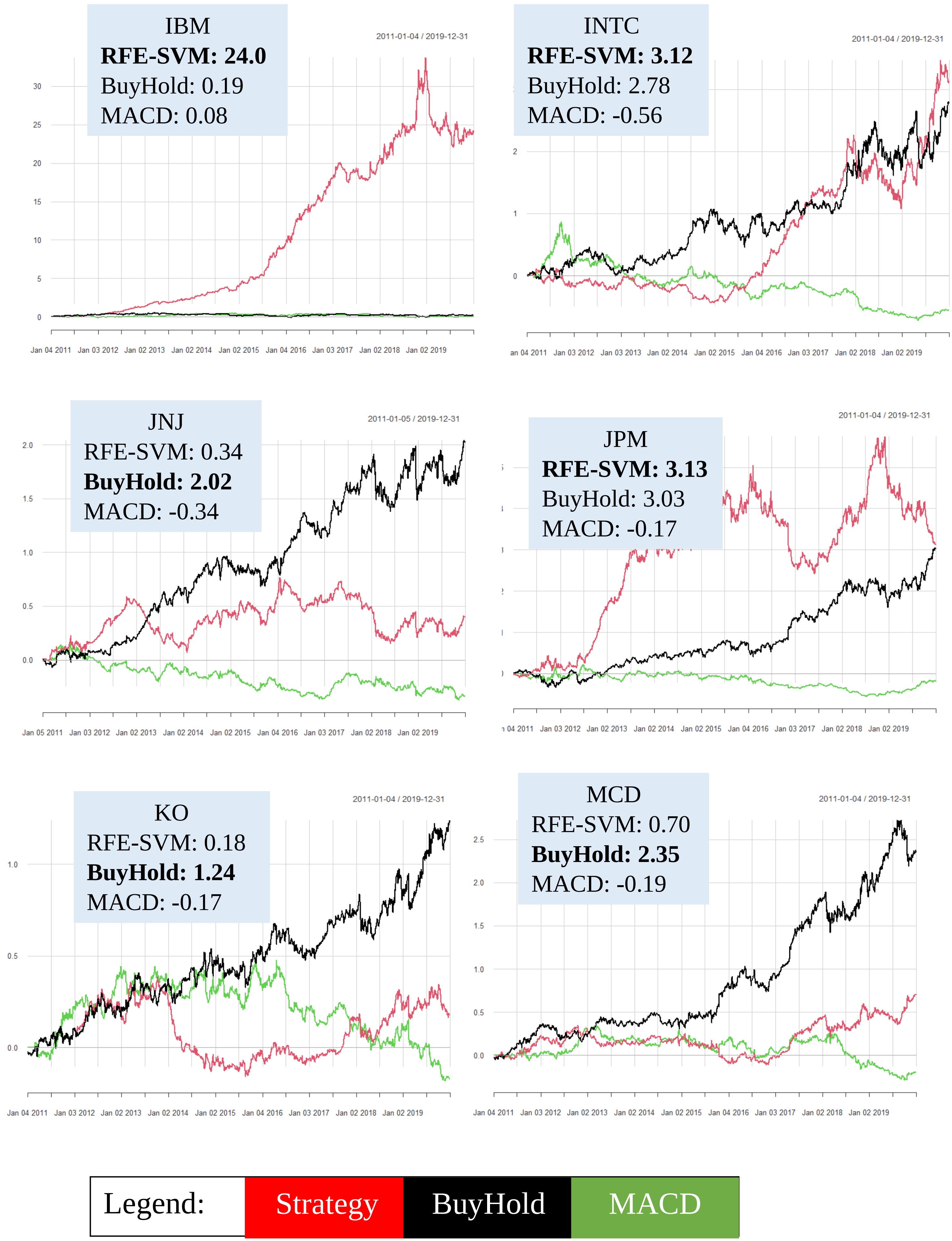
Chapter 6

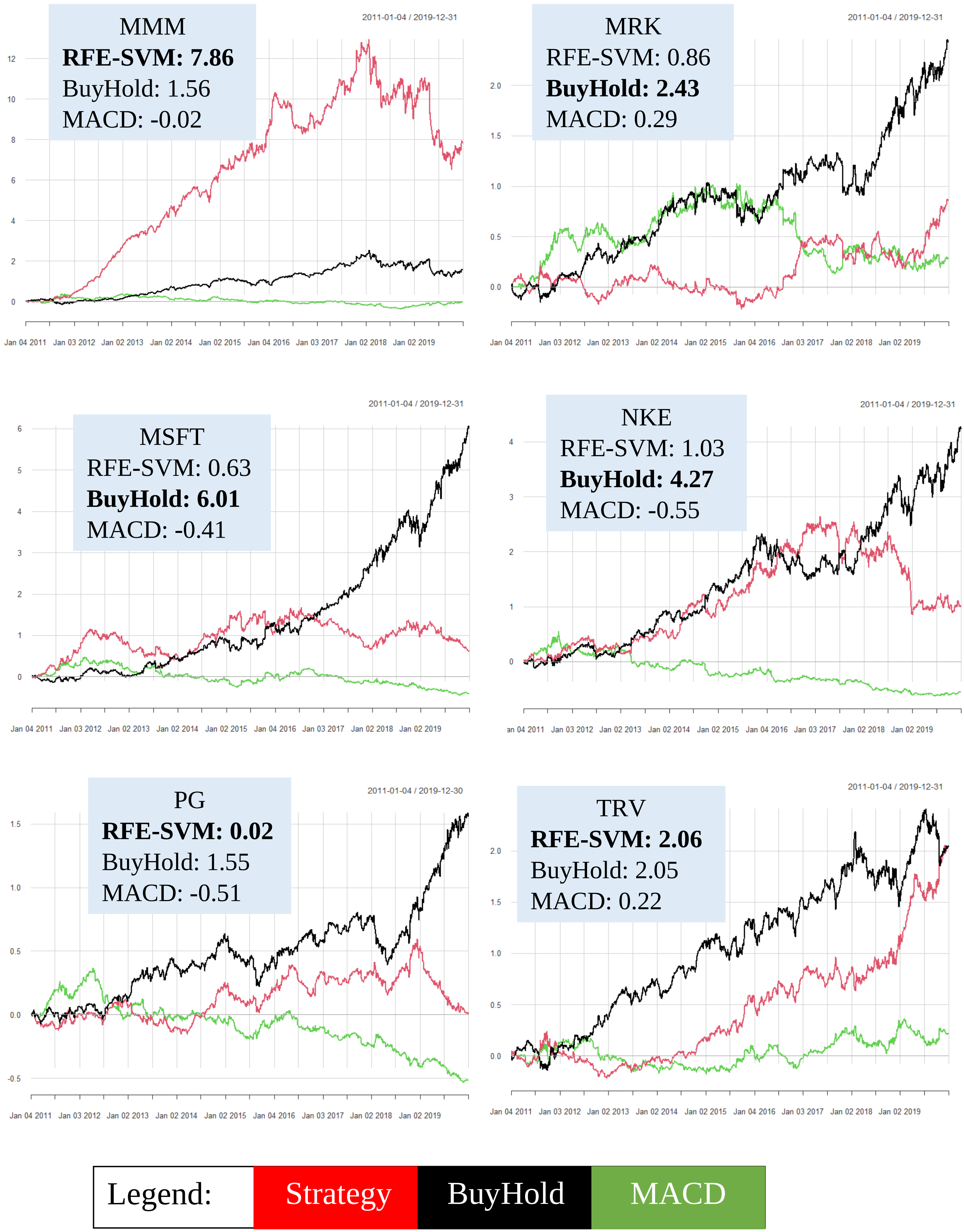
Appendix

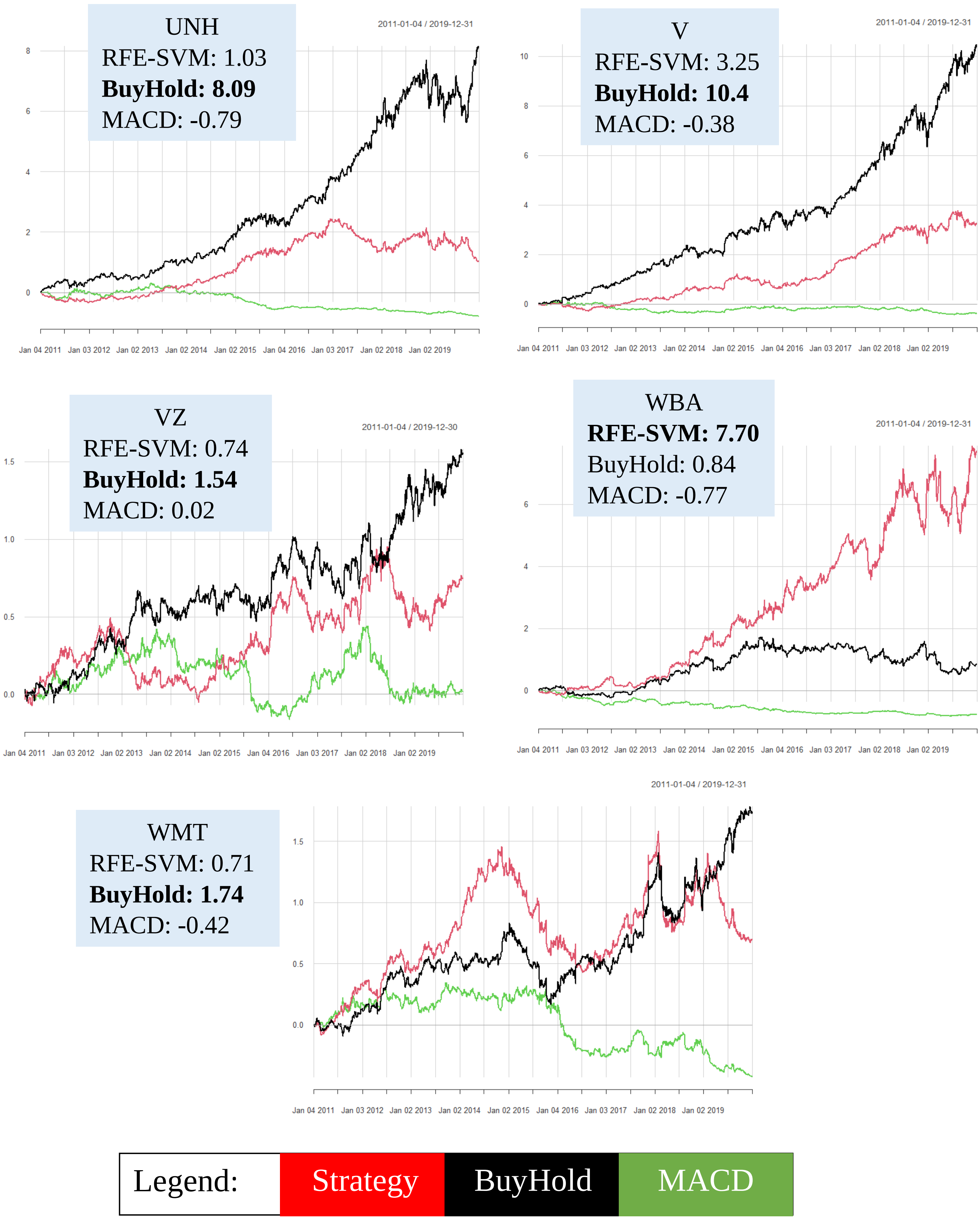
This section includes all of the cumulative return graphs for the three trading strategies (buy-and-hold, RFE-SVM, and MACD) for the 48 stocks. The graphs are organized by stock group and then alphabetically. The large-cap stocks are displayed first, followed by the small-cap stocks and European stocks, respectively.

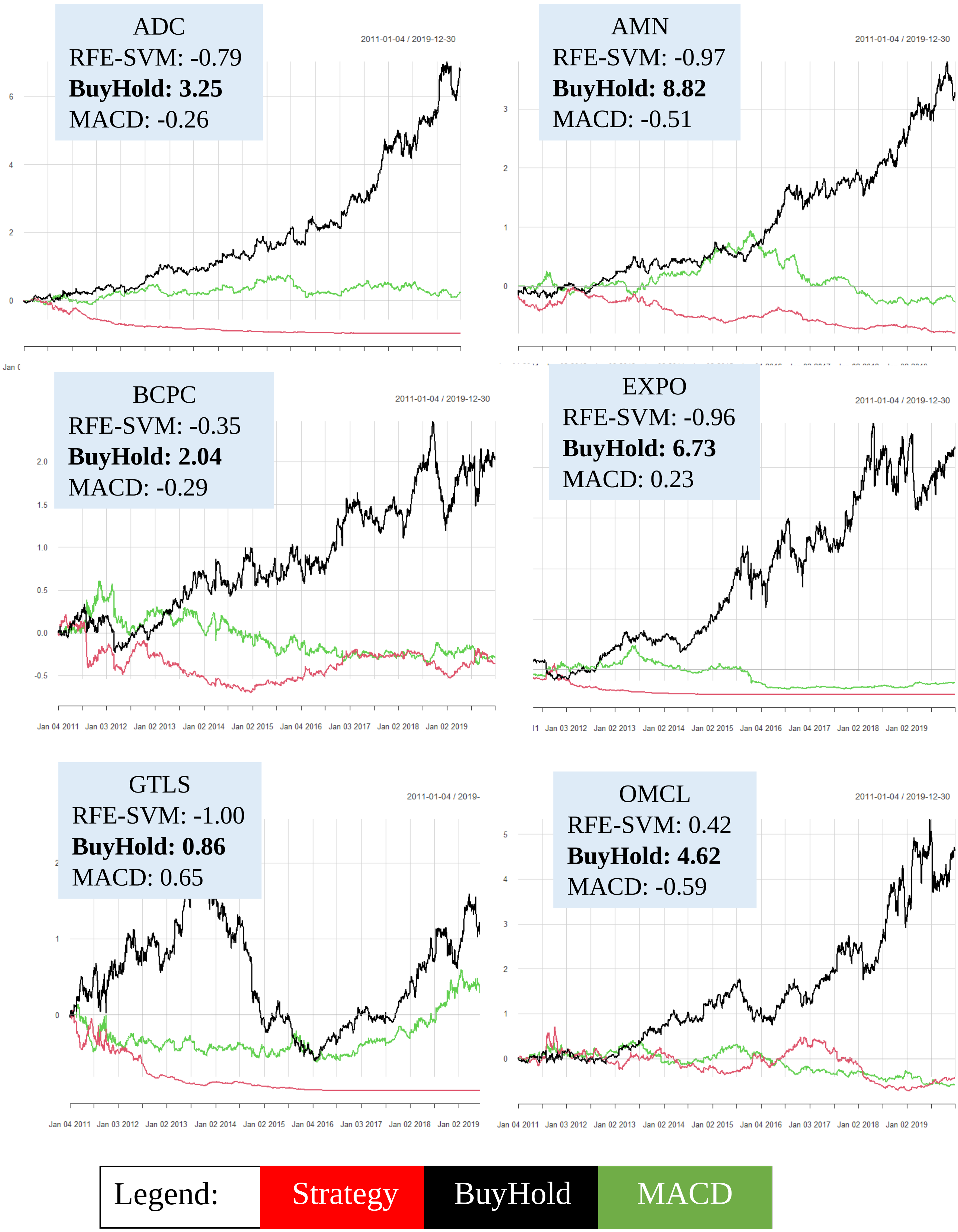

AAPL
RFE-SVM: 4.41
BuyHold: 6.17
MACD: -0.5
2011-01-04 / 2019-12-31
AMGN
RFE-SVM: 5.96
BuyHold: 4.28
MACD: 0.28
2011-01-04 / 2019-12-31
AXP
RFE-SVM: 1.03
BuyHold: 2.27
MACD: -0.49
2011-01-04 / 2019-12-30
BA
RFE-SVM: 0.80
BuyHold: 5.12
MACD: -0.42
2011-01-04 / 2019-12-31
CAT
RFE-SVM: 4.25
BuyHold: 1.01
MACD: 0.02
2011-01-04 / 2019-12-31
CRM
RFE-SVM: 3.56
BuyHold: 3.76
MACD: -0.75
2011-01-04 / 2019-12-31
Jan 04 2011
Jan 03 2012
Jan 02 2013
Jan 02 2014
Jan 02 2015
Jan 04 2016
Jan 03 2017
Jan 02 2018
Jan 02 2019
Legend:
Strategy
BuyHold
MACD

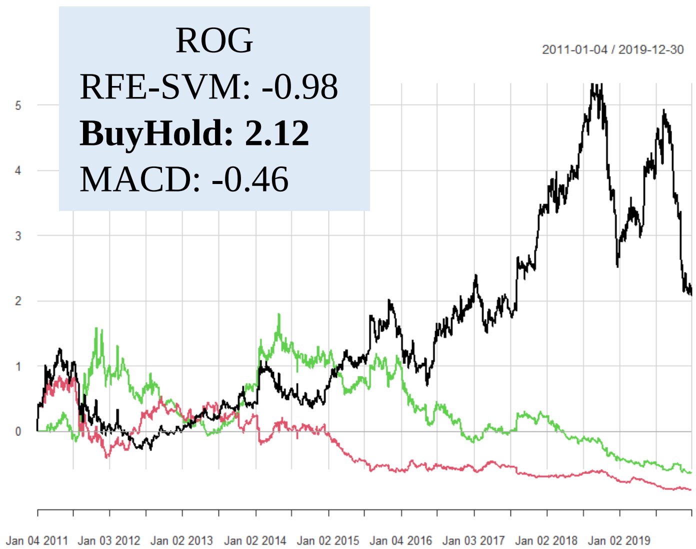

CSCO
RFE-SVM: 1.16
BuyHold: 2.00
MACD: 0.02
2011-01-04 / 2019-12-31
CVX
RFE-SVM: 0.85
BuyHold: 0.82
MACD: 0.02
2011-01-04 / 2019-12-31
DIS
RFE-SVM: 2.13
BuyHold: 3.32
MACD: 0.08
2011-01-04 / 2019-12-31
GS
RFE-SVM: 42.3
BuyHold: 0.51
MACD: -0.51
2011-01-04 / 2019-12-31
HD
RFE-SVM: 3.53
BuyHold: 6.56
MACD: -0.43
2011-01-04 / 2019-12-31
HON
RFE-SVM: 1.88
BuyHold: 3.15
MACD: -0.03
2011-01-04 / 2019-12-31
Jan 04 2011 Jan 03 2012 Jan 02 2013 Jan 02 2014 Jan 02 2015 Jan 04 2016 Jan 03 2017 Jan 02 2018 Jan 02 2019
Legend:
Strategy
BuyHold
MACD

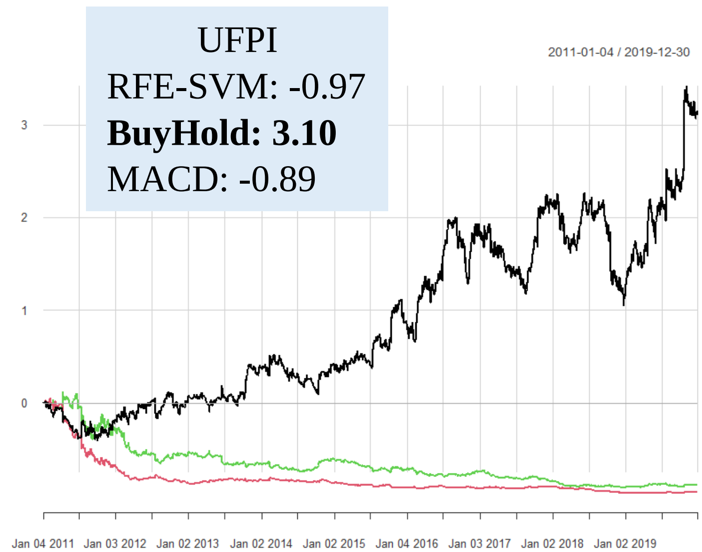

IBM
RFE-SVM: 24.0
BuyHold: 0.19
MACD: 0.08
2011-01-04 / 2019-12-31
INTC
RFE-SVM: 3.12
BuyHold: 2.78
MACD: -0.56
2011-01-04 / 2019-12-31
JNJ
RFE-SVM: 0.34
BuyHold: 2.02
MACD: -0.34
2011-01-05 / 2019-12-31
JPM
RFE-SVM: 3.13
BuyHold: 3.03
MACD: -0.17
2011-01-04 / 2019-12-31
KO
RFE-SVM: 0.18
BuyHold: 1.24
MACD: -0.17
2011-01-04 / 2019-12-31
MCD
RFE-SVM: 0.70
BuyHold: 2.35
MACD: -0.19
2011-01-04 / 2019-12-31
Legend:
Strategy
BuyHold
MACD

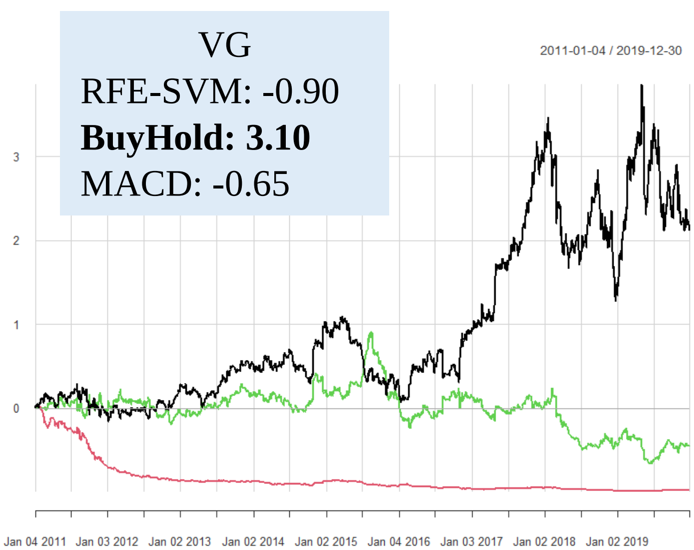

MMM
RFE-SVM: 7.86
BuyHold: 1.56
MACD: -0.02
2011-01-04 / 2019-12-31
MRK
RFE-SVM: 0.86
BuyHold: 2.43
MACD: 0.29
2011-01-04 / 2019-12-31
MSFT
RFE-SVM: 0.63
BuyHold: 6.01
MACD: -0.41
2011-01-04 / 2019-12-31
NKE
RFE-SVM: 1.03
BuyHold: 4.27
MACD: -0.55
2011-01-04 / 2019-12-31
PG
RFE-SVM: 0.02
BuyHold: 1.55
MACD: -0.51
2011-01-04 / 2019-12-30
TRV
RFE-SVM: 2.06
BuyHold: 2.05
MACD: 0.22
2011-01-04 / 2019-12-31
Jan 04 2011
Jan 03 2012
Jan 02 2013
Jan 02 2014
Jan 02 2015
Jan 04 2016
Jan 03 2017
Jan 02 2018
Jan 02 2019
Legend:
Strategy
BuyHold
MACD


UNH
RFE-SVM: 1.03
BuyHold: 8.09
MACD: -0.79
2011-01-04 / 2019-12-31
V
RFE-SVM: 3.25
BuyHold: 10.4
MACD: -0.38
2011-01-04 / 2019-12-31
VZ
RFE-SVM: 0.74
BuyHold: 1.54
MACD: 0.02
2011-01-04 / 2019-12-30
WBA
RFE-SVM: 7.70
BuyHold: 0.84
MACD: -0.77
2011-01-04 / 2019-12-31
WMT
RFE-SVM: 0.71
BuyHold: 1.74
MACD: -0.42
2011-01-04 / 2019-12-31
Jan 04 2011 Jan 03 2012 Jan 02 2013 Jan 02 2014 Jan 02 2015 Jan 04 2016 Jan 03 2017 Jan 02 2018 Jan 02 2019
Legend:
Strategy
BuyHold
MACD

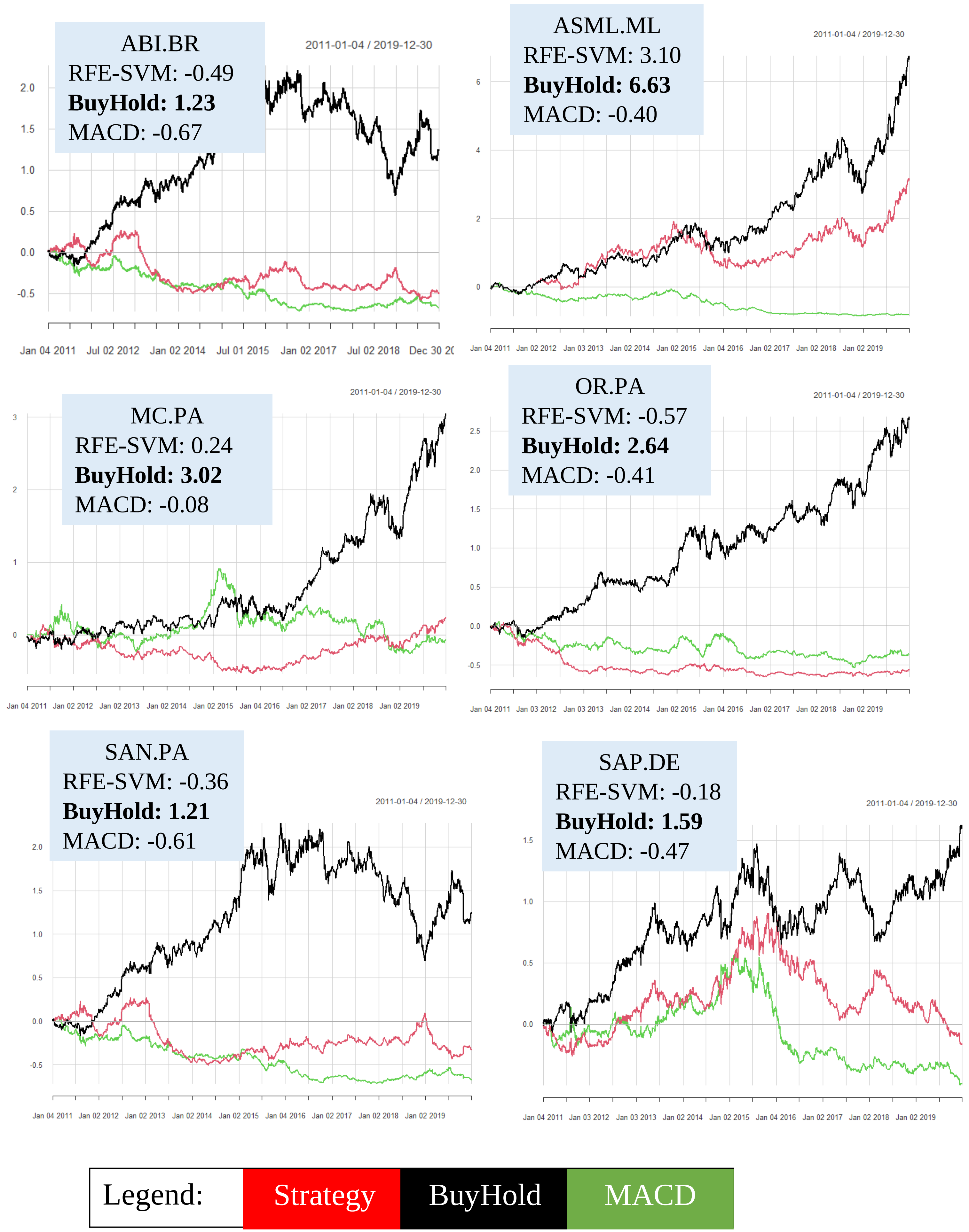

ADC
RFE-SVM: -0.79
BuyHold: 3.25
MACD: -0.26
2011-01-04 / 2019-12-30
AMN
RFE-SVM: -0.97
BuyHold: 8.82
MACD: -0.51
2011-01-04 / 2019-12-30
BCPC
RFE-SVM: -0.35
BuyHold: 2.04
MACD: -0.29
2011-01-04 / 2019-12-30
EXPO
RFE-SVM: -0.96
BuyHold: 6.73
MACD: 0.23
2011-01-04 / 2019-12-30
GTLS
RFE-SVM: -1.00
BuyHold: 0.86
MACD: 0.65
OMCL
RFE-SVM: 0.42
BuyHold: 4.62
MACD: -0.59
2011-01-04 / 2019-12-30
Jan 04 2011 Jan 03 2012 Jan 02 2013 Jan 02 2014 Jan 02 2015 Jan 04 2016 Jan 03 2017 Jan 02 2018 Jan 02 2019
Legend:
Strategy
BuyHold
MACD

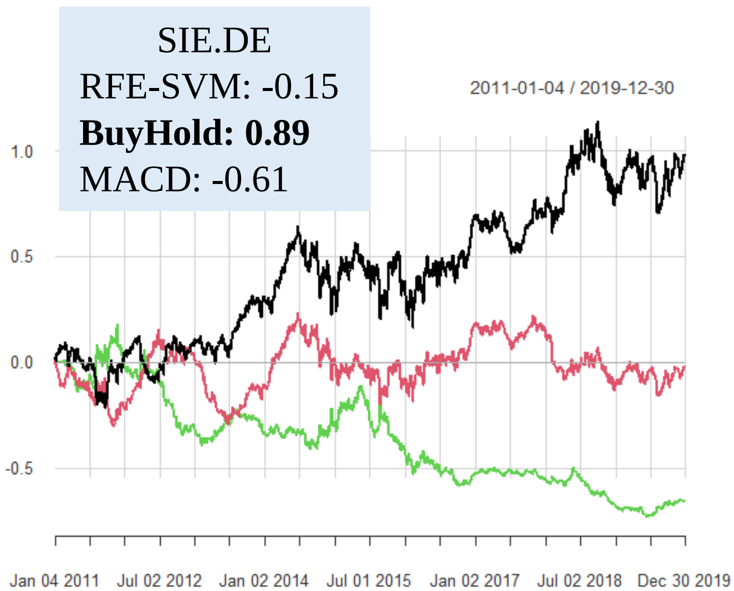


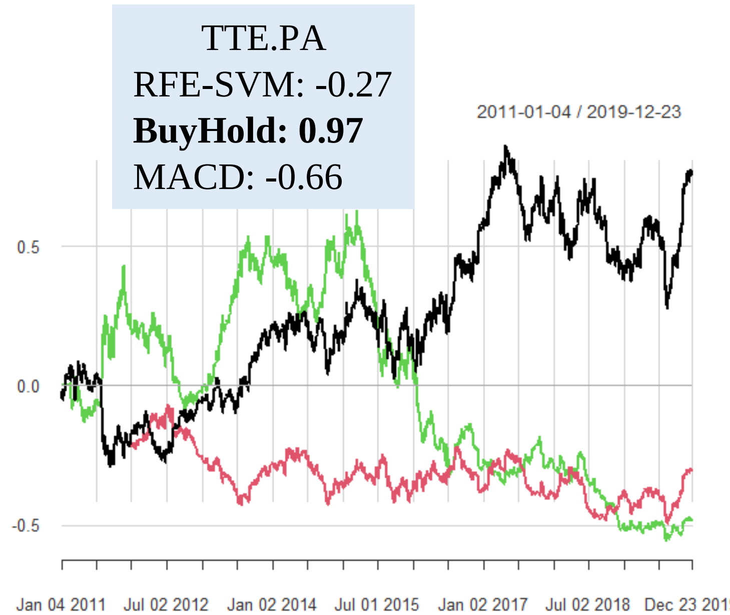


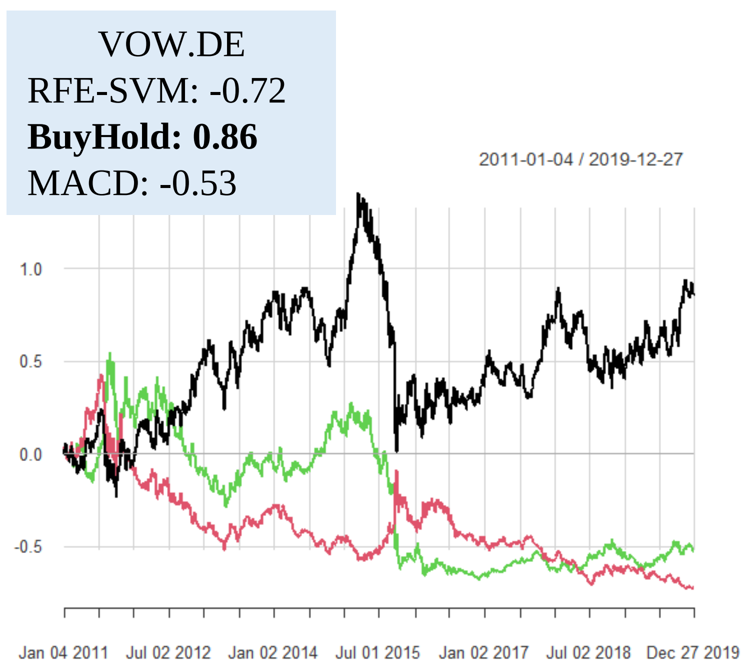


Legend:	Strategy	BuyHold	MACD


ABI.BR
RFE-SVM: -0.49
BuyHold: 1.23
MACD: -0.67
2011-01-04 / 2019-12-30
Jan 04 2011 Jul 02 2012 Jan 02 2014 Jul 01 2015 Jan 02 2017 Jul 02 2018 Dec 30 20
ASML.ML
RFE-SVM: 3.10
BuyHold: 6.63
MACD: -0.40
2011-01-04 / 2019-12-30
Jan 04 2011 Jan 02 2012 Jan 03 2013 Jan 02 2014 Jan 02 2015 Jan 04 2016 Jan 02 2017 Jan 02 2018 Jan 02 2019
MC.PA
RFE-SVM: 0.24
BuyHold: 3.02
MACD: -0.08
2011-01-04 / 2019-12-30
Jan 04 2011 Jan 02 2012 Jan 02 2013 Jan 02 2014 Jan 02 2015 Jan 04 2016 Jan 02 2017 Jan 02 2018 Jan 02 2019
OR.PA
RFE-SVM: -0.57
BuyHold: 2.64
MACD: -0.41
2011-01-04 / 2019-12-30
Jan 04 2011 Jan 03 2012 Jan 03 2013 Jan 02 2014 Jan 02 2015 Jan 04 2016 Jan 02 2017 Jan 02 2018 Jan 02 2019
SAN.PA
RFE-SVM: -0.36
BuyHold: 1.21
MACD: -0.61
2011-01-04 / 2019-12-30
Jan 04 2011 Jan 02 2012 Jan 02 2013 Jan 02 2014 Jan 02 2015 Jan 04 2016 Jan 02 2017 Jan 02 2018 Jan 02 2019
SAP.DE
RFE-SVM: -0.18
BuyHold: 1.59
MACD: -0.47
2011-01-04 / 2019-12-30
Jan 04 2011 Jan 03 2012 Jan 03 2013 Jan 02 2014 Jan 02 2015 Jan 04 2016 Jan 02 2017 Jan 02 2018 Jan 02 2019
Legend: Strategy BuyHold MACD







Legend:	Strategy	BuyHold	MACD

Bibliography

[1] Steven B Achelis. *Technical Analysis from A to Z: Covers Every Trading Tool– from the Absolute Breadth Index to the Zig Zag.* Probus Publishing Company, 1995.

[2] Asaf Adi, David Botzer, Gil Nechushtai, and Guy Sharon. Complex event processing for financial services. In *2006 IEEE Services Computing Workshops*, pages 7–12. IEEE, 2006.

[3] Sidney S Alexander. Price movements in speculative markets: Trends or random walks. *Industrial Management Review (pre-1986)*, 2(2):7, 1961.

[4] Gabriel Dan I Anghel. Stock market efficiency and the MACD. Evidence from countries around the world. *Procedia economics and finance*, 32:1414–1431, 2015.

[5] Adebiyi A Ariyo, Adewumi O Adewumi, and Charles K Ayo. Stock price prediction using the ARIMA model. In *2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*, pages 106–112. IEEE, 2014.

[6] Bala Arshanapalli, Matthew Lutey, William Nelson, and Micah Pollak. The profitability of technical analysis during financial bubbles. *The Journal of Portfolio Management*, 47(1):168–175, 2020.

[7] Cliff Asness. The august of our discontent: Once more unto the breach? *AQR Capital Management*, Aug 2017.

[8] C Narendra Babu and B Eswara Reddy. Selected Indian stock predictions using a hybrid ARIMA-GARCH model. In *2014 International Conference on Advances in Electronics Computers and Communications*, pages 1–6. IEEE, 2014.

[9] Madhushi Niluka Bandara, Rajitha Madhushan Ranasinghe, Rashmi Woranga Mudugamuwa Arachchi, Channa Gayan Somathilaka, Srinath Perera, and Daya Chinthana Wimalasuriya. A complex event processing toolkit for detecting technical chart patterns. In *2015 IEEE International Parallel and Distributed Processing Symposium Workshop*, pages 547–556. IEEE, 2015.

[10] Hendrik Bessembinder and Kalok Chan. The profitability of technical trading rules in the asian stock markets. *Pacific-basin finance journal*, 3(2-3):257–284, 1995.

[11] Hendrik Bessembinder and Kalok Chan. Market efficiency and the returns to technical analysis. *Financial management*, pages 5–17, 1998.

[12] William Brock, Josef Lakonishok, and Blake LeBaron. Simple technical trading rules and the stochastic properties of stock returns. *The Journal of finance*, 47(5):1731–1764, 1992.

[13] Thomas N Bulkowski. *Encyclopedia of chart patterns*. John Wiley & Sons, 2021.

[14] Li-Juan Cao and Francis Eng Hock Tay. Support vector machine with adaptive parameters in financial time series forecasting. *IEEE Transactions on neural networks*, 14(6):1506–1518, 2003.

[15] Chien-Ping Chen, Massoud Metghalchi, and Xavier Garza Gomez. Technical analysis of the danish stock market. *Huntsville, Texas, USA*, 181, 2011.

[16] James Chen. Investopedia. https://www.investopedia.com/, 2021. Accessed: 2022-03-06.

[17] Long Chen and Lu Zhang. A better three-factor model that explains more anomalies. *Journal of Finance*, 65(2):563–595, 2010.

[18] Weiling Chen, Chai Kiat Yeo, Chiew Tong Lau, and Bu Sung Lee. Leveraging social media news to predict stock index movement using RNN-boost. *Data & Knowledge Engineering*, 118:14–24, 2018.

[19] John H Cochrane. Presidential address: Discount rates. *The Journal of finance*, 66(4):1047–1108, 2011.

[20] Davis David. Random Forest classifier tutorial: How to use tree-based algorithms for machine learning. *freeCodeCamp.org*, Aug 2020.

[21] Author dprogrammer. RNN, LSTM, and Gru. *dProgrammer lopez*, Jun 2020.

[22] Robert D Edwards. Technical analysis of stock trends. *Boston, Mass*, 1966.

[23] Eugene F Fama. Efficient market hypothesis: A review of theory and empirical work. *Journal of Finance*, 25(2):28–30, 1970.

[24] Eugene F Fama and Marshall E Blume. Filter rules and stock-market trading. *The Journal of Business*, 39(1):226–241, 1966.

[25] Eugene F Fama and Kenneth R French. *The cross-section of expected stock returns*. University of Chicago Press, 1992.

[26] Eugene F Fama and Kenneth R French. A five-factor asset pricing model. *Journal of financial economics*, 116(1):1–22, 2015.

[27] Alan Fan and Marimuthu Palaniswami. Stock selection using support vector machines. In *IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No. 01CH37222)*, volume 3, pages 1793–1798. IEEE, 2001.

[28] Guanhao Feng, Stefano Giglio, and Dacheng Xiu. Taming the factor zoo: A test of new factors. *The Journal of Finance*, 75(3):1327–1370, 2020.

[29] Yahoo! Finance. SPDR Euro Stoxx 50 ETF Holdings. https://finance.yahoo.com/quote/FEZ/holdings/, 2022. Accessed: 2022-02-01.

[30] Yahoo! Finance. Yahoo Finance - stock market live quotes, business, and finance news. https://finance.yahoo.com/, 2022. Accessed: 2022-03-06.

[31] Thomas Fischer and Christopher Krauss. Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2):654–669, 2018.

[32] James Foye. Testing alternative versions of the fama–french five-factor model in the uk. *Risk Management*, 20(2):167–183, 2018.

[33] Google. Google trends search popularity: Economic crisis vs. artificial neural network. https://trends.google.com/trends/?geo=US, 2022. Accessed: 2022-03-06.

[34] Hakan Gunduz. An efficient stock market prediction model using hybrid feature reduction method based on variational autoencoders and recursive feature elimination. *Financial Innovation*, 7(1):1–24, 2021.

[35] Campbell R Harvey, Yan Liu, and Heqing Zhu. The cross-section of expected returns. *The Review of Financial Studies*, 29(1):5–68, 2016.

[36] Adam Hayes. Retail investor definition. *Investopedia*, Dec 2021.

[37] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-term Memory. *Neural computation*, 9:1735–80, 12 1997.

[38] Soosung Hwang and Alexandre Rubesam. Searching the factor zoo. 2019.

[39] Investment Company Institute. 2021 Investment Company Fact Book. https://www.icifactbook.org/, 2021. Accessed: 2022-03-25.

[40] Ashwini M Joshi and Sameer Prabhune. Random forest: A hybrid implementation for sarcasm detection in public opinion mining. *Int. J. Innov. Technol. Explor. Eng*, 8(12):5022–5025, 2019.

[41] Dhruba Karki. Can you guess how much data is generated every day? *Takeo.ai*, Nov 2020.

[42] Charles D Kirkpatrick II and Julie A Dahlquist. *Technical analysis: the complete resource for financial market technicians*. FT press, 2010.

[43] Ki-Yeol Kwon and Richard J Kish. Technical trading strategies and return predictability: NYSE. *Applied Financial Economics*, 12(9):639–653, 2002.

[44] David J Leinweber and Robert D Arnott. Quantitative and computational innovation in investment management. *Journal of Portfolio Management*, 21(2):8, 1995.

[45] Raymond M Leuthold. Random walk and price trends: the live cattle futures market. *The Journal of Finance*, 27(4):879–889, 1972.

[46] Library of Congress. This Month in Business History: Dow Jones Industrial Average. https://guides.loc.gov/this-month-in-business-history/may/djia-first-published, 2017. Accessed: 2022-03-06.

[47] Andrew W Lo. Adaptive markets and the new world order (corrected may 2012). *Financial Analysts Journal*, 68(2):18–29, 2012.

[48] Andrew W Lo, Harry Mamaysky, and Jiang Wang. Foundations of technical analysis: Computational algorithms, statistical inference, and empirical implementation. *The journal of finance*, 55(4):1705–1765, 2000.

[49] Stephan Maier. rpatrec: Recognising Visual Charting Patterns in Time Series Data. 2017. R package version 1.0.1.

[50] Saulius Masteika and Aleksandras Vytautas Rutkauskas. Research on futures trend trading strategy based on short term chart pattern. *Journal of Business Economics and Management*, 13(5):915–930, 2012.

[51] Lukas Menkhoff. The use of technical analysis by fund managers: International evidence. *Journal of Banking & Finance*, 34(11):2573–2586, 2010.

[52] Nagaraj Naik and Biju R Mohan. Stock price movements classification using machine and deep learning techniques-the case study of indian stock market. In *International Conference on Engineering Applications of Neural Networks*, pages 445–452. Springer, 2019.

[53] Rodolfo Toríbio Farias Nazário, Jéssica Lima e Silva, Vinicius Amorim Sobreiro, and Herbert Kimura. A literature review of technical analysis on stock markets. *The Quarterly Review of Economics and Finance*, 66:115–126, 2017.

[54] Christopher Neely, Paul Weller, and Rob Dittmar. Is technical analysis in the foreign exchange market profitable? a genetic programming approach. *Journal of financial and Quantitative Analysis*, 32(4):405–426, 1997.

[55] Mehtabhorn Obthong, Nongnuch Tantisantiwong, Watthanasak Jeamwatthanachai, and Gary Wills. A survey on machine learning for stock price prediction: algorithms and techniques. 2020.

[56] Carol L Osler and PH Chang. Head and shoulders: Not just a flaky pattern. *FRB of New York staff report*, (4), 1995.

[57] Cheol-Ho Park and Scott H Irwin. What do we know about the profitability of technical analysis? *Journal of Economic surveys*, 21(4):786–826, 2007.

[58] Jigar Patel, Sahil Shah, Priyank Thakkar, and Ketan Kotecha. Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert systems with applications*, 42(1):259–268, 2015.

[59] pioneer903pioneer903. Plotting 3D decision boundary from Linear SVM. *Stack Overflow*, Apr 1964.

[60] R Core Team. R: A Language and Environment for Statistical Computing. 2017.

[61] Keywan Christian Rasekhschaffe and Robert C Jones. Machine learning for stock selection. *Financial Analysts Journal*, 75(3):70–88, 2019.

[62] Francesco Rundo, Francesca Trenta, Agatino Luigi di Stallo, and Sebastiano Battiato. Machine learning for quantitative finance applications: A survey. *Applied Sciences*, 9(24):5574, 2019.

[63] Peter Salvage. Artificial Intelligence Sweeps Hedge Funds. *BNY Mellon*, Mar 2019.

[64] SN Sarma. Stock market seasonality in an emerging market. *Vikalpa*, 29(3):35–42, 2004.

[65] Franco Scarselli and Ah Chung Tsoi. Universal approximation using feedforward neural networks: A survey of some existing methods, and some new results. *Neural networks*, 11(1):15–37, 1998.

[66] Sima Siami-Namini, Neda Tavakoli, and Akbar Siami Namin. A comparison of ARIMA and LSTM in forecasting time series. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1394–1401. IEEE, 2018.

[67] Robert F Stambaugh and Luboš Pástor. Liquidity risk and expected stock returns. 2002.

[68] Richard J Sweeney. Beating the foreign exchange market. *The Journal of Finance*, 41(1):163–182, 1986.

[69] Prodromos Tsinaslanidis and Francisco Guijarro. What makes trading strategies based on chart pattern recognition profitable? *Expert Systems*, 38(5):e12596, 2021.

[70] Hrishikesh Vachhani, Mohammad S Obiadat, Arkesh Thakkar, Vyom Shah, Raj Sojitra, Jitendra Bhatia, and Sudeep Tanwar. Machine learning based stock market analysis: A short survey. In *International Conference on Innovative Data Communication Technologies and Application*, pages 12–26. Springer, 2019.

[71] Vanguard. Vanguard ETF profile. https://investor.vanguard.com/etf/profile/overview/VIOO/portfolio-holdings, 2021. Accessed: 2022-03-15.

[72] Analytics Vidhya. ANN vs CNN vs RNN: Types of neural networks. https://www.analyticsvidhya.com/blog/2020/02/cnn-vs-rnn-vs-mlp, 2020. Accessed: 2022-02-12.

[73] Jiong Zhang, Qi Lei, and Inderjit Dhillon. Stabilizing gradients for deep neural networks via efficient SVD parameterization. In *International Conference on Machine Learning*, pages 5806–5814. PMLR, 2018.

[74] Xiao-dan Zhang, Ang Li, and Ran Pan. Stock trend prediction based on a new status box method and AdaBoost probabilistic support vector machine. *Applied Soft Computing*, 49:385–398, 2016.