Forecasting Price Movements using Technical Indicators: Investigating Window Size Effects

Yauheniya Shynkevich[1,*], T.M. McGinnity[1,2], Sonya Coleman[1], Ammar Belatreche[1], Yuhua Li[3]

[1]Intelligent Systems Research Centre, Ulster University, BT48 7JL, Derry, UK

[2]School of Science and Technology, Nottingham Trent University, Nottingham, UK

[3]School of Computing, Science and Engineering, University of Salford, Manchester, UK

Abstract — The creation of a predictive system that correctly forecasts future changes of a stock price is crucial for investment management and algorithmic trading. The use of technical analysis for financial forecasting has been successfully employed by many researchers. Window size is a time frame parameter required to be set when calculating many technical indicators. This study explores how the performance of the predictive system depends on a combination of a forecast horizon and a window size for forecasting variable horizons. Technical indicators are used as input features for machine learning algorithms to forecast future directions of stock price movements. The dataset consists of ten years daily price time series for fifty stocks. The highest prediction performance is observed when the window size is approximately equal to the forecast horizon. This novel pattern is studied using multiple performance metrics: prediction accuracy, winning rate, return per trade and Sharpe ratio.

Keywords— decision making, evaluating forecasts, price forecasting, technical trading, stock market forecasting

1. INTRODUCTION

Analysis and accurate forecasts of stock markets become increasingly more challenging and advantageous [1]. Globalization of the economy continuously requires innovations in the field of computational science and information technologies. Financial forecasting is often based on computational intelligence techniques that can analyse large amounts of data and extract meaningful information [2]. A predictive system that is able to forecast the direction of a stock price movement

[*] Corresponding author.
E-mail address: shynkevich-y@email.ulster.ac.uk

helps investors to make appropriate decisions, improves profitability and hence decreases possible losses. Forecasting of the stock market prices and their directional changes plays an important role in financial decision making, investment management and algorithmic trading.

Financial forecasting based on computational intelligence approaches often uses technical analysis (TA) to form features used as inputs to the approaches. Time series of stock price and trading volume size are utilised to compute a technical indicator (TI) where a composition of open, low, high and close price values and volume size is taken over a certain time period. As reported by Atsalakis and Valavanis [2], approximately 20% of the financial market forecasting approaches use TIs as input features. In order to compute TIs, their parameters are required to be set. Every time a new predictive system is developed, its creators select a number of indicators suitable for their purposes and then choose appropriate parameters values to calculate them. The selection of indicators suitable for forming the input features and the choice of their parameters remains an area of active research. In order to overcome difficulties such as determining optimal combinations of indicators or tuning their parameters several efforts have been made [3], [4]. However, there is no sophisticated well-established technique that allows the system's developers to easily select appropriate parameters. To date, the dependency of a predictive system performance on a forecast horizon and indicator parameters has not been fully investigated. To the best of our knowledge, there is no existing research investigating the relationship between the forecast horizon and the time frame used to calculate TIs. However, every researcher that is developing a financial forecasting system based on TA faces the problem of selecting appropriate values of parameters for the chosen technical indicators (TIs).

The current research sheds light on this topic and studies how the performance of a predictive financial system based on TA changes when the forecast horizon is intended for prediction and a time frame is varied for computing TIs. Time period is used to calculate TIs and is required to be set prior to the calculation. Later in this paper this time period will be referred as the window size of an indicator. The paper investigates the dependency of the forecasting system performance on the combination of the window size and the forecast horizon, and searches for the optimal combination of these parameters that maximizes the performance of the predictive system when predicting the direction of a price movement. The following pattern was discovered: for each horizon the highest prediction performance is reached when the window size is approximately equal to the horizon. Sets of reasonable values of forecast horizons and window sizes were selected for analysis. Three well-established machine learning approaches, Support Vector Machines (SVM), Artificial Neural Networks (ANN) and k-Nearest Neighbours (kNN), were utilized to forecast directions of future price movements. The presented research studies the relationship between the forecast horizon and the window size utilising different performance measures that demonstrates that the observed pattern persists over a number of metrics. The prediction accuracy describes how good the developed prediction system is for the defined task. Return per trade, Sharpe ratio and winning rate characterize the prediction system from a trading point of view. These measures provide information about the potential profitability of the system and help evaluate the relationship between two examined parameters. The results are analysed and verified using t-test statistical analysis that confirms their validity. A previously undiscovered pattern is revealed in the current study that enables researchers to go for a simple solution when selecting a window size for a specific forecast horizon. It also can be used for setting initial values to window size parameters in order to find a better combination through varying their values. Taking into account the popularity of the TIs, this research explores meaningful empirical rules, which should be considered when creating a predictive system based on TA.

The reminder of the paper is organized as follows. A theoretical background to financial forecasting is reviewed in Section II. Related work is discussed in Section III. Section IV describes the feature extraction, the experimental model, parameter settings and aspects of the proposed algorithm. Section V describes the dataset used and Section VI discusses the obtained results and key findings. Finally, Section VII concludes the paper and outlines directions for future research.

2. Market Theories and Trading Philosophies

The efficient markets hypothesis (EMH) of Fama [5] is based on the idea that all the information available is continuously processed by the market and is embedded into asset prices which results in the instant assimilation of any piece of new information at any given point in time. There are three levels of market efficiency, strong, semi-strong and weak, defined by Fama's theory. The weak level claims that present market prices reflect all historical publicly available information. The semi-strong form of the EMH assumes that prices of the traded stocks already integrated and absorbed all the historical and present public information. The strong EMH supposes that even insider and latent information is immediately incorporated in a market price. The fundamentals of the EMH postulate that all historical, general and private information about an asset is embodied into its current price that allows for a possibility to systematically outperform the market. In the Random Walk Theory, stock price fluctuations are inter independent and follow the same distribution. Consequently, historical information about an asset price has no correlation with its future movements and cannot be used for predictions. Conforming to this theory, a random walk is the most probable way the asset price moves, and accurate predictions are not feasible.

The question about market efficiency with respect to its extent and applicability to different markets remains an active and ongoing area of research where contradictory results are present. Recently researchers have proposed a counter-theory named Adaptive Market Hypothesis (AMH) in an attempt to align the EMH with behaviour finance [6]. Behaviour finance looks at the market price as a purely perceived value instead of a derivative of its costs. Market agents have cognitive biases including overreaction, overconfidence, information bias and representative bias, which implies that many human errors in information processing and reasoning can be predictable [7]. A comprehensive empirical study on the AMH was conducted in [8] where three of the most developed markets were examined: the UK, US and Japanese stock markets. The authors used long run data and formed five-yearly subsamples subject to linear and nonlinear tests to distinguish various behaviours of stock returns over time. The results from linear tests reveal that each stock market provides evidence of being an adaptive market where returns are going through periods of dependence and independence. Nonlinear tests reveal strong dependence for each market in every subsample although the magnitude of the dependence varies considerably. The overall results strongly suggest that the AMH describes the behaviour of stock returns better than the EMH.

According to the results of recent research [2], financial markets do not exhibit random behaviour and it is possible to forecast market changes. In the trading world, two major trading philosophies exist.

A fundamental trading philosophy focuses on the analysis of the financial state of an entity that is determined through economic indicators. It studies the factors that influence supply and demand. The decisions are made based on the performance of the company, its competitors, industry, sector and general economy. The economic indicators taken into account are company's economic growth, debt level, inflation, industry return on equity (ROE), unemployment, earnings, etc. On the contrary TA utilizes historical data to forecast future behaviour of an asset price. TA is based on the idea that the behaviour of preceding investors and traders is often repeated by the subsequent ones. It is supposed that profitable opportunities can be disclosed through computing the averaged movements of the historical time series of price and volume and comparing them against their current values. It is also believed that some psychological price barriers exist and their observation can lead to profitable strategies. TIs help the traders to estimate whether the observed trend is weak or strong or whether a stock is overbought or oversold. Traders have developed many TIs such as moving average (MA), rate of change (ROC), relative strength index (RSI), oscillators, etc. A comprehensive analysis of technical trading strategies and their performance is presented in [9]. The authors separated the studies into early studies (1960-1987) and modern studies (1988-2004). Early studies feature several limitations in the testing procedure, and their results differ from market to market. Modern studies are enhanced in relation to the limitations of early studies, and in most cases (approximately 60%) the profitability of technical trading strategies was affirmed. Mixed results were presented in approximately 20% of studies, whereas the rest demonstrated negative results and rejected the usefulness of technical analysis. More recent studies have shown that the market predictability depends on business cycles and the performance of trading rules based on TA varies in time and depends on the financial markets conditions [10], [11]. Lately TIs have become extensively used as input features in machine learning based financial forecasting systems [2]. These systems learn to recognize complex patterns in market data and forecast future behaviours of an asset price. In this study, TA is employed to form input features for machine learning techniques, and the importance of the time frame used to compute the indicators is examined.

3. RELATED WORK

Technical indicators, such as MA and RSI, are mathematical tools used to determine whether a stock is oversold or overbought or a price trend is weak or strong, and therefore to forecast its future price movements. A number of efforts were made to determine optimal combinations of indicators or to tune parameters, such as time frames and the smoothing period. An attempt to find optimal parameters for a widely used indicator, moving average convergence/ divergence (MACD), was made using evolutionary algorithms [12]. Another commonly used TI, RSI, was added in the later research [13], and the same technique was applied to analyse these two indicators and determine appropriate values of their parameters. Subsequently, a parallel evolutionary algorithm was proposed for parameters optimization of MACD and RSI in [3]. The results of these experiments demonstrate that the developed predictive system showed better performance when the parameters of TIs were fine-tuned than when standard parameter values suggested in the literature [14] were utilised. In [4], close prices of the stock PETR4 were predicted using several combinations of the window size and prediction horizon, however no analysis of the relationship between these parameters was presented. In [15], the iJADE Stock Advisor system was evaluated for short-term and long-term trend predictions based upon different window sizes used for data preprocessing. The authors did not use TIs but mentioned that the concept of their price pre-processing is analogous to that of the TA. The optimal window size found for the short-term stock predictions was equal to three days, and that for the long-term prediction was found to be 20 days.

Financial forecasting is usually built on numerical information about financial assets and the market state. Many computational intelligence techniques have been utilized for this purpose. SVM is a popular machine learning technique used by many researchers. Tay and Cao [16], [17] compared the SVM approach with an ANN and explored its suitability for predicting market prices. According to their results, SVM outperforms the ANN in forecasting a relative change of bonds and stock index futures prices for a five day prediction horizon. Afterwards, Kim [18] examined the SVM sensitivity to its parameters, the upper bound C and kernel parameters. The SVM performance was compared to case-based reasoning (CBR) and ANN approaches. According to the experimental results, SVM surpasses both approaches and its accuracy is sensitive to the considered parameters. Huang et al. (2005) used SVM to investigate the predictability of stock market price movements by forecasting the weekly directional movements of the NIKKEI 225 index. Two macroeconomic variables, the exchange rate of US Dollars against Japanese Yen and the S&P 500 Index, were utilised as inputs. The authors found that the highest performance was achieved by a proposed combining model that integrates SVM with other methods. The performance of SVM and ANN in forecasting directional movements of a stock index was compared in [20]. The models were tested on emerging markets and both approaches showed strong capability in financial forecasting. Arroyo and Maté [21] forecasted histogram time series using the kNN approach and stated that promising results were achieved using meteorological and financial data. In [22] kNN was applied to create an automated framework for trading stocks listed on São Paulo Stock Exchange. The authors employed common tools of TA such as TIs, transaction costs, stop loss/gain and RSI filters and claimed that the developed trading system is capable of producing profit. SVMs are widely applied and extended in recent studies. Khemchandani [23] proposed a novel approach, regularized least squares fuzzy SVR, for financial forecasting, and demonstrated its efficacy. In [24] the authors proposed to use principal component analysis for forecasting directional changes in the Korean composite stock price and Hangseng indices. The authors stated that the method achieved high hit ratios. In [25], least square SVM was employed to examine the usefulness of TA and its prediction power for identifying trend movements in small emerging Southeast European markets. The results showed that specific TIs are not consistent in different time periods but proved that TA has a certain level of prediction power and allows for generation of excess returns.

Taking into account the reviewed literature, three well-established learning approaches, SVM, ANN and kNN, were selected to study the relationship between the forecast horizon and window size for the purpose of finding the optimal combination. Additionally, the Naïve Bayes approach was employed for comparison, however it showed low prediction performance and the corresponding results were not presented in the current research paper. The results obtained using SVM, ANN and kNN were compared to explore whether the observed pattern is specific to a selected machine learning technique or it is reproducible for the others. The comparison showed that the SVM method outperformed other methods on the underlying task and the presence of the pattern depends on the overall prediction performance of the applied machine learning technique.

4. Predictive System

This section provides details about the selected input features, the experimental model used for training and forecasting the directions of stock price movements, the parameters used to calculate the input features and a set of the forecast horizons chosen for the analysis.

4.1 Input Features

Ten TIs were selected to form input feature vectors based on reviewed financial forecasting literature [18], [20], [26], [27]. For each stock, TIs were calculated for each trading day from raw stock data which include open, close, high and low prices and trading volume. The following eight TIs were selected as they require a window size parameter to be set. Each indicator allows to include additional information derived from a stock price in a different way.

1. **Simple Moving Average (SMA)** is a trend indicator calculated as an average price over a particular period:

$$SMA_n = \sum_{i=0}^{n-1} C_{t-i} \Big/ n \tag{1}$$

where C_t is a close price on day t, n is a window size.

2. **Exponential Moving Average (EMA)** is a type of moving average where weights, ω_i, of past prices decrease exponentially:

$$EMA_n = \sum_{i=0}^{n-1} \omega_i C_{t-i} \Big/ n\,. \tag{2}$$

3. **Average True Range (ATR)** provides information about the degree of price volatility.

$$ATR_n = EMA_n(\max(H_t - L_t, \mathrm{abs}(H_t - C_{t-1}), \mathrm{abs}(L_t - C_{t-1})), \tag{3}$$

where H_t, L_t and C_t are the high, low and closing prices at time t respectively.

4. **Average Directional Movement Index (ADMI)** indicates the strength of a trend in price time series. It is a combination of the negative and positive directional movements indicators, DI_n^+ and DI_n^-:

$$ADMI_n = 100*(DI_n^+ - DI_n^-)/(DI_n^+ + DI_n^-)\ , \tag{4}$$

$$DI_n^+ = 100*EMA_n(DM^+)/ATR_n\,, \tag{5}$$

$$DI_n^- = 100*EMA_n(DM^-)/ATR_n\,, \tag{6}$$

where $DM^+ = \max(C_t - C_{t-1}, 0)$ and $DM^- = \min(C_t - C_{t-1}, 0)$ are positive and negative directional movements.

5. **Commodity Channel Index (CCI)** is an oscillator used to determine whether a stock is overbought or oversold. It assesses the relationship between an asset price, its moving average and deviations from that average:

$$CCI_n = (M^t - SMA_n(M^t) \Big/ \left(0.015\sum_{i=1}^{n}\left|M_{t-i+1} - SMA_n(M^t)\right| \Big/ n\right), \tag{7}$$

where M^t is a sum of the high, low and closing prices at time t, $M^t = H_t + L_t + C_t$.

6. **Price rate-of-change (ROC)** shows the relative difference in stock closing prices over a period of time:

$$ROC_n = (C_t - C_{t-n})/C_{t-n}\ . \tag{8}$$

7. **Relative Strength Index (RSI)** compares the size of recent gains to recent losses, it is intended to reveal the strength or weakness of a price trend from a range of closing prices over a given time period:

$$RSI_n = 100 - 100\Big/\left(1 + EMA_n(DM^+)\,/\,EMA_n(DM^-)\right). \tag{9}$$

8. The **William's %R oscillator** shows the relationship between the current closing price and the high and low prices over the latest *n* days:

$$Williams\_R_n = 100*(H_n - C_t)/(H_n - L_n)\ . \tag{10}$$

The following two indicators were selected as they also require a window size parameter to be set and additional parameters for smoothing. Smoothing parameters where selected in agreement with the suggestions in the literature [28].

9. **Stochastic %K** is a technical momentum indicator that compares a close price and its price interval during a trading period and gives a signal meaning that a stock is oversold or overbought:

$$\%\mathrm{K}_n = 100 * (C_t - LL_n)/(HH_n - LL_n), \tag{11}$$

where HH_n and LL_n are mean highest high and lowest low prices in the last n days respectively.

10. **Stochastic %D** is a 3-days EMA of Stochastic %K, it gives a turnaround signal meaning that a stock is oversold or overbought:

$$\%\mathrm{D}_n = EMA_3(\%\mathrm{K}_n). \tag{12}$$

Technical Analysis Library (TA-Lib) is an open-source library available at www.ta-lib.org which is widely used by trading software developers for performing TA of market data [29]. It was utilised for calculating TIs in this study. The main focus of this research is the uncertainty regarding the optimal value of a window size that should be used for calculation of indicators.

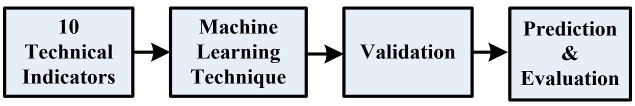

Fig. 1. The architecture of a prediction system.

4.2 Forecast Horizon

A set of different forecast horizons was used in the experiments. The values of 1, 3, 5, 7, 10, 15, 20, 25 and 30 trading days were chosen for analysis. There has been a lot of interest in one-day-ahead forecasting which remains an area of active research [20], [30], [31]. Therefore, the smallest horizon was set to one trading day. The successive values were selected so that the balance between the advantages of a detailed analysis using small increases in a forecasting horizon and the consumption of the computational time was kept. Starting from the forecast horizon equal to ten, each consecutive horizon is larger than the preceding one by five trading days. The largest horizon utilized is 30 trading days which is approximately equals to a month and half.

4.3 Window Size

In this paper, the range of the employed window sizes starts from the smallest value equal to three trading days because values of one or two days would not allow the calculation of all the indicators selected for the analysis. The subsequent values of the window size range are selected to be the same as values in the range of forecast horizons. Therefore, a range of window sizes consists of 3, 5, 7, 10, 15, 20, 25 and 30 days.

4.4 The experimental model

The architecture of the prediction system used for forecasting directional changes in stock prices is displayed in Fig. 1. For each data point, ten input features were used. SVM, ANN and kNN machine learning techniques were used to investigate the relationship between the system performance and the combination of the window size and the forecast horizon. A number of techniques were applied to understand whether this relationship depends on a chosen approach. A system was trained and tested separately for each stock with nine values of the horizons and eight values of the window sizes. Every performance measure utilised to test the system's ability to forecast price movements was calculated for a number of combinations {number of classes, forecast horizon, window size} is averaged over the total number of considered stocks.

4.5 Methodology

This subsection describes the methodology used for training, validation and testing. It provides details about the usage of the three machine leaning techniques employed in experiments and the parameters tuning. Additionally, it specifies the benchmark model utilised for statistical testing and analysis of results. Three machine learning techniques and the benchmark are described one by one below.

1. **SVM.** In this study, the SVM approach is implemented using the LibSVM library which is an open-source software [32]. In the experiments a sigmoid function was used as a kernel. It takes a gamma parameter, γ, that significantly affects performance and is required to be optimized. Another parameter of the SVM model that requires optimization is a penalty rate for misclassification, C. A grid search was employed to identify good parameters combinations where values of gamma and C were selected from exponentially growing sequences $\gamma=\{2^{-15}, 2^{-13}, \ldots, 2^{3}\}$ and $C=\{2^{-5}, 2^{-3}, \ldots, 2^{15}\}$ respectively as

suggested in [33]. Five-fold cross-validation was employed to find optimal parameters among different combinations of their values where the whole training dataset was divided into five folds. The system was trained using four folds and then tested using the remaining fifth fold. The procedure was repeated five times for each fold being used for testing. The system's performance under different parameters settings was measured using the overall prediction accuracy which is defined as the percentage of correctly classified data points. The obtained accuracy was averaged over the five folds and this measure was used to determine optimal parameter values.

2. **ANN.** The Matlab neural networks toolbox was used in all experiments. The feedforward ANN model employed contains three layers: input, hidden and output. The network has ten input neurons that correspond to the ten calculated input features. The number of neurons in the hidden layer was also set to ten for all stocks. The output layer contains two or three nodes depending on the number of classes considered. During the validation procedure, to minimize overfitting 25% of training dataset was employed.

3. **kNN.** In order to understand how the system performance depends on the selected approach, the kNN approach was used for comparison. The implementation of this machine learning technique in MATLAB was utilised for the experiments. The optimal number k of the nearest neighbours was selected from a range of $\{1,2,\ldots,10\}$ using five-fold cross-validation, and the Euclidean distance was employed as the similarity measure.

4. **Benchmark.** To evaluate the results produced by the developed predictive system and to get a better understanding of its performance, a standard benchmark [34], [35] following the conditions of the stock market was utilised. In the long-term, stock market prices tend to increase, and it is essential to assure that the trading system based on predictions outperforms a simple benchmark and actually generates value. The simplest trading strategy is a buy-and-hold strategy where an asset is bought at a starting point in time, held for a specified period of time and sold at the end. The idea is similar to the index investment and constitutes a common way for investment funds to benchmark themselves. The benchmark was used to derive the statistical significance of the results obtained using the machine learning techniques. These tests show whether the application of the described techniques to form

trading strategies adds extra value in comparison to the buy-and-hold strategy and whether this value is statistically significant.

5. Dataset

This section provides detailed information regarding the dataset used, the data pre-processing techniques applied and the process of assigning labels to the data points.

5.1 Raw Data

The prediction system discussed above is applied to predict future price movements of the components of the S&P 500 stock market index. The index comprises 500 large companies having high market capitalizations and publicly traded on the NASDAQ and NYSE markets. Only companies with a trading history started before January 29, 2002 were considered and 50 stocks were randomly selected from the list of the S&P 500 index components. The list of the analyzed stocks is available in Appendix. The dataset was downloaded from the Yahoo! Finance website which is a publicly available source of data. 2640 data points each corresponding to a single trading day were constructed from the data for each stock. A single data point contains daily open, close, high and low prices, adjusted for stock splits and paid dividends, and trading volume for the corresponding trading day. The dataset was divided into two sets, a training set and a testing set. The training dataset contains 1740 trading days from January 29, 2002 to December 23, 2008 and the testing dataset contains data for 900 trading days from December 24, 2008 to July 20, 2012. The relative size ratio between training and testing data sets is approximately 2:1.

5.2 Data pre-processing

Data pre-processing is required in order to transform raw time series data into a form acceptable for applying a machine learning technique. The pre-processing steps used are listed below.

- **Interpolation** was carried out when information about prices and volume for a trading day was not available. For some stocks, several points were missing in the data. Overall, the missing data constituted less than 0.1% out of all data points. The price and volume values for these data points were interpolated from the existing adjacent price and volume values using linear regression.

- **Transformation** of the original time series data into a set of TIs and the usage of the derived values are widely utilised in research techniques [2]. In the current study, ten TIs were computed for each data point of each stock and used as the input.

- **Normalization** of the data set was applied after transformation so that each input feature had zero mean and unit variance. The mean and variance were computed for each feature based on the training dataset. These values were then applied to normalize both the training and testing datasets.

5.3 Data Labelling

In the following experiments, the directions of future price movements are predicted by classifying them into two and three classes. The assignment of labels to each data point was performed according to the forthcoming behaviour of the closing prices, as described below.

In two class classification, class labelling is illustrated in (13). The label ‘Up’ was assigned to a data point when the corresponding closing stock price went up. The label ‘Down’ was assigned to a data point when the corresponding closing stock price went down,

$$Label_{2CL}\left(p_i\right)=\begin{cases}'Up' \quad , \; if \; \left(C_{t+s}-\mathrm{C}_t\right)/C_t>0; \\ 'Down', \; if \; \left(C_{t+s}-\mathrm{C}_t\right)/C_t\le 0,\end{cases} \tag{13}$$

where s is a forecast horizon, C_t and C_{t+s} are closing prices of a stock on the days t and $t+s$ respectively. Equation (14) explains how class labels were assigned to data points for three class classification,

$$Label_{3CL}\left(p_i\right)=\begin{cases}'Up', & if \; \left(C_{t+s}-\mathrm{C}_t\right)/C_t>\delta; \\ 'No\ Move', & if \; -\delta\le\left(C_{t+s}-\mathrm{C}_t\right)/C_t\le\delta; \\ 'Down', & if \; \left(C_{t+s}-\mathrm{C}_t\right)/C_t<-\delta,\end{cases} \tag{14}$$

where δ is a threshold used to define the level of an absolute value of a relative price change below which the change is considered to be insignificant. In three class classification, label ‘Up’ is assigned if the relative change in the price is higher than the pre-defined threshold. In a similar way, label ‘Down’ is appointed to an instance of data when a price has decreased noticeably so that a negative relative price change is lower than the threshold taken with a negative sign. If the relative change lies in the range between the negative and positive thresholds, it is considered to be insignificant and label ‘No Move’ is assigned to a data point. Considering the terminology for directional changes [36], the threshold is a minimal relative price change by which the price has risen or dropped so that this change can be regarded as a directional movement. In this research the negative and positive thresholds are equal in absolute value and opposite in sign, however the absolute value varies depending on the horizon. The threshold values used for different forecast horizons are shown in Table I. These values were selected such that on average one third of data points belongs to the 'No Move' class. The threshold values increase with an increase in a horizon because price movements become larger with the passage of time, and larger threshold values are required to assign one third of the data points to the 'No Move' class. The selection of the threshold values can also be justified from the profitability point of view. An accurate forecasting of a decrease or an increase in a stock price value does not necessarily enable a profitable strategy. Transaction costs, capital gain taxes and interest rates on borrowed funds or stocks are reducing the net profit from a trade [37]. With the usage of the threshold these losses are eliminated through avoiding trades when an asset price does not change significantly. The amount of interest spent to borrow funds or stocks is increasing with the passage of time. Therefore, the threshold used for a longer time period has to be larger than for a shorter time period which explains the selection of the threshold values in the current research. A typical value of the threshold for one day ahead forecasting is 0.5-1%.

The percentage of data points assigned to each class depending on the forecast horizon are given in Table II. Table II (a) provides information about the percentage of cases when an asset price increased ('Up' class) or decreased ('Down' class) after a number of trading days equal to the forecast horizon has passed. Table II (b) presents the fraction of 'Up', 'Down' and 'No Move' labels assigned to data points for the three class classification. Stock price fluctuates constantly around its market value in both increasing and decreasing directions. For short forecast horizons approximately the same number of data points belongs to each class whereas for longer horizons the percentage of 'Up' points dominates. It is caused by the fact that the market was generally rising during the testing period from December 2008 to July 2012, and that the overall trend tends to have a stronger influence on the price changes for longer forecast horizons than for shorter ones.

TABLE I. Threshold values

Forecast horizon, trading days	Threshold value (%)
1	0.63
3	1.15
5	1.49
7	1.79
10	2.14
15	2.65
20	3.08
25	3.48
30	3.94

TABLE II. The percentage of data points assigned to each class depending on the forecasting horizon

(a) Two Class Classification

Forecast horizon, trading days	1	3	5	7	10	15	20	25	30
Fraction of 'Up' points, %	51.7	54.1	55.2	56.6	57.8	59.8	61.3	62.1	62.8
Fraction of 'Down' points, %	48.3	45.9	44.8	43.4	42.2	40.2	38.7	37.9	37.2

(b) Three Class Classification

Forecast horizon, trading days	1	3	5	7	10	15	20	25	30
Fraction of 'Up' points, %	35.1	36.8	37.8	38.7	39.8	41.3	42.4	43.2	42.9
Fraction of 'Down' points, %	31.6	29.8	28.9	27.9	26.9	25.4	24.3	23.5	21.8
Fraction of 'No Move' points, %	33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3

6. Experimental Results and Discussions

This section discusses the experimental results obtained using the developed prediction system. Experiments were performed separately for each selected stock. For the sake of diversified analysis of the relationship between the window size and the forecast horizon, the performance of each machine learning techniques was measured using a number of performance metrics: prediction accuracy, winning rate, return per trade and Sharpe ratio. Prediction accuracy characterizes the classification performance of the machine learning technique. Determining the direction of a price move is important, which is described by the accuracy. But in particular those points that come with large price movements have to be identified, while mistakes in identifying movements with almost zero return will have little lasting effect on the performance of the trading system. In order to investigate the behaviour of the predictive system from a trading point of view using different settings, the predictive system was evaluated as a trading system. An assumption was made that each time the predictive system generates a buy/sell signal, an amount of money X is invested. When the system predicted an 'Up' price movement, a long trade was made where an underlying stock was bought for X at the moment of prediction and sold at the end of a forecast horizon. When the system predicted a 'Down' price movement, a short trade was made where an underlying stock was sold for X at the moment of prediction and bought back at the end of the forecast horizon. For two class classification, trades were made for each of 900 data points because the system predicted either 'Up' or 'Down' movements for every single data point. For three class classification, no trades were made when 'No Move' class was predicted, therefore the number of trades made during the testing phase varied. Based on the developed virtual trading system, winning rate, return per trade and Sharpe ratio were computed. These performance measures help to study the relationship between the forecast horizon and window size from the point of a risk and reward. All the results are provided in tables where each row corresponds to a certain horizon. The highest value in a row is highlighted in green whereas red indicates the lowest value. The background colours in the remaining cells are scaled depending on how close their values are to the highest and the lowest points. The colour map helps to identify the pattern in the experimental results. Every value in the tables represents a mean value of a considered measure over 50 stocks. It is accompanied by its standard deviation followed after a '±' sign. The indication of both the mean and standard deviation provides more detailed information about the estimated values and helps to get more insight about their distribution. In order to conclude whether the applied strategy generates additional value, each value of a metric was compared to the corresponding value of the benchmark model using independent two-sample t-test statistics. The confidence level of 99% of probability that the obtained value is higher than and statistically significant from that of the benchmark was selected for the test. If a mean value of a measure did not appear to be statistically higher than that of the benchmark, it was underlined and shown in Italic font.

6.1 Prediction Accuracy

The prediction accuracy obtained for a single stock was calculated using (15) and (16) for two and three class classifications respectively:

$$Accuracy_{2CL} = \frac{TrueUp + TrueDown}{N} \tag{15}$$

$$Accuracy_{3CL} = \frac{TrueUp + TrueNoMove + TrueDown}{N} \tag{16}$$

where N is the total number of classified data points, *TrueUp*, *TrueDown* and *TrueNoMove* are correctly classified up, down and no movement respectively. The averaged accuracy (the mean) was computed as an arithmetic mean of accuracies over 50 stocks.

The values of the averaged accuracies and their standard deviations obtained with respect to the forecast horizons and window sizes using different approaches are presented in Tables III and IV for two and three class classification respectively. The highest prediction accuracy of 75.4% was obtained by SVM for two class classification when predicting for 15 days ahead with the window size equal to 15 days. The obtained prediction accuracy is within a comparable range of that from the related literature. For example, the highest accuracy obtained by Kara et al. [20] is equal to 75.74%. The combined model developed by Huang et al. [19] showed 75% of the forecasting accuracy. These results indicate that the values produced by the developed predictive system are comparable with the state-of-the-art approaches and that the decision to select SVM to investigate the relationships between window size and forecast horizon is robust and reasonable. The following pattern is observed for SVM: the highest prediction accuracy for each value of a forecast horizon is generally reached when a window size is approximately equal to the horizon. Almost the same values of accuracy can be observed for several adjacent windows, but a range of windows that produces high values of accuracy is moving towards larger window sizes with the increase of the forecast horizon. This pattern is reproduced for both two and three class classification. The standard deviation is gradually increasing with an increase in the forecast horizon. However, it tends to be smaller for values around the highest value in a row, which corresponds to window sizes roughly equal to the forecast horizon. This behaviour emphasizes the idea that setting the window size approximately equal to the selected forecast horizon gives high classification performance and increases the robustness of the system.

TABLE III. Averaged prediction accuracy in percentage (%) for two class classification obtained using different classifiers (a) SVM, (b) ANN and (c) kNN

Horizon, days	Window Size, days 3	5	7	10	15	20	25	30
	(a) SVM							
1	67.5±2.2	64.5±2.2	64.4±2.4	62.6±2.2	57.6±1.9	57.1±1.8	56.0±1.9	55.3±1.9
3	71.9±2.8	72.8±2.8	69.5±3.1	67.8±2.9	65.0±3.2	63.5±2.5	62.0±2.0	60.9±2.4
5	70.4±2.9	74.4±3.1	72.1±3.6	71.0±3.0	69.0±3.0	67.3±3.1	65.4±2.9	64.0±3.2
7	68.3±2.6	74.2±3.6	73.4±3.5	72.8±4.1	72.0±3.3	69.9±3.4	68.3±3.1	66.5±3.4
10	64.5±4.5	71.6±3.6	72.6±3.8	74.5±3.5	74.0±3.7	72.3±3.9	70.3±3.7	68.7±4.4
15	61.8±6.0	68.4±5.4	70.6±5.1	74.1±4.2	75.4±4.0	74.1±4.0	73.3±4.0	72.4±4.7
20	*60.0±7.0*	65.7±6.1	67.9±6.0	71.9±5.7	74.3±4.8	74.6±4.5	73.9±4.9	73.5±5.3
25	*58.6±7.6*	*63.3±6.9*	65.5±6.5	69.7±6.1	73.3±5.0	74.3±4.7	74.4±4.7	74.4±5.4
30	*58.2±8.8*	*61.5±8.5*	*63.4±7.8*	67.2±7.0	71.0±5.4	72.8±5.3	73.7±5.7	74.4±6.0
	(b) ANN							
1	63.6±4.9	61.9±3.2	58.9±3.7	57.2±2.8	55.2±2.3	53.7±2.6	53.2±2.7	52.8±2.2
3	71.0±4.1	70.6±4.8	66.3±4.7	63.7±4.4	61.7±4.0	60.1±4.0	58.4±3.8	57.4±3.5
5	68.3±5.0	72.9±4.3	69.5±5.2	67.1±6.7	65.7±4.7	63.4±4.7	61.9±4.1	60.8±3.5
7	65.9±4.6	72.2±6.4	71.9±5.5	71.4±5.3	67.9±6.0	65.9±5.5	64.4±5.1	62.5±5.1
10	61.7±6.4	69.8±5.6	71.5±5.5	71.9±6.1	69.7±6.8	69.1±3.8	67.1±4.6	65.0±4.7
15	*59.2±6.7*	66.7±5.6	67.9±7.2	71.1±7.7	73.2±5.3	71.8±4.9	69.6±6.0	67.6±7.8
20	*57.4±7.2*	*62.3±8.4*	66.4±6.5	69.9±7.5	71.5±9.1	71.0±9.7	70.8±8.3	68.7±9.0
25	*56.3±8.1*	*61.2±7.2*	*63.5±7.6*	67.3±7.4	70.2±8.0	71.1±9.1	71.8±7.0	69.6±8.0
30	*55.4±8.5*	*58.2±9.3*	*60.7±9.2*	*63.4±8.8*	68.5±8.0	68.8±9.7	70±10.2	71.2±8.9
	(c) kNN							
1	55.9±3.2	54.2±2.3	52.8±2.2	*52.1±2.5*	*51.6±1.8*	*50.5±2.1*	*50.7±2.2*	*50.8±1.7*
3	58.9±4.0	58.9±3.9	56.5±3.4	*55.0±3.0*	*53.3±2.9*	*52.2±2.3*	*52.4±2.2*	*51.8±2.1*
5	58.1±4.1	60.3±4.3	58.6±4.1	56.8±4.1	*55.0±3.6*	*53.6±2.7*	*52.8±2.9*	*52.4±2.9*
7	*56.8±4.5*	59.8±6.1	59.4±5.3	*57.9±4.9*	*55.8±4.3*	*54.6±3.2*	*54.2±2.9*	*53.4±3.2*
10	*55.2±5.3*	*58.4±6.3*	*59.0±5.9*	*58.8±5.8*	*56.8±4.7*	*56.1±4.4*	*55.5±3.7*	*54.6±3.3*
15	*54.4±5.6*	*56.6±6.3*	*57.7±6.3*	*58.4±6.1*	*58.5±6.0*	*57.4±5.2*	*56.5±4.5*	*56.1±4.8*
20	*53.4±6.6*	*55.3±7.1*	*56.5±7.1*	*58.1±6.8*	*58.5±6.9*	*58.0±6.0*	*57.5±5.6*	*57.3±5.1*
25	*52.6±7.4*	*55.0±7.8*	*55.2±7.1*	*56.9±6.7*	*57.3±6.9*	*57.6±6.2*	*57.7±6.0*	*57.3±6.1*
30	*52.1±7.6*	*53.7±7.7*	*54.4±7.6*	*55.9±7.2*	*57.1±7.1*	*57.7±6.7*	*57.8±6.8*	*57.3±5.8*

When comparing the selected machine learning approaches for two class classification, ANN performs worse than SVM on average by -3.0% in terms of the mean and shows the standard deviation higher by 1.8%. For three class classification, the difference is -4.0% and 3.5% for the averaged mean and standard deviation respectively. This suggests that ANN showed slightly worse performance than SVM on the task considered. However, the prediction accuracy is still relatively high in comparison to the benchmark, and the pattern observed for SVM is clearly visible for ANN for both two and three classes. The vast majority of the accuracy values obtained for SVM and ANN are higher and statistically significant in comparison to those of the benchmark with a few exceptions when predicting long forecast horizons using short window sizes for input calculations.

TABLE IV. Averaged prediction accuracy in percentage (%) for three class classification obtained using different classifiers (a) SVM, (b) ANN and (c) kNN

Horizon, days	Window Size, days							
	3	5	7	10	15	20	25	30
	(a) SVM							
1	52.6±4.1	50.6±3.4	48.3±4.1	47.1±3.8	44.8±4.2	44.6±4.3	43.7±4.3	*43.7±4.7*
3	57.8±3.6	58.8±3.0	55.7±3.3	53.8±3.3	51.5±3.2	50.0±3.7	48.3±4.2	47.9±4.0
5	56.3±3.8	60.9±3.3	59.1±3.4	57.5±3.5	55.1±3.0	53.2±3.6	51.7±3.7	50.9±3.6
7	53.9±3.9	60.2±3.8	59.8±3.4	58.9±3.8	57.1±3.5	54.7±3.9	53.6±3.8	52.4±3.8
10	50.5±4.6	57.6±3.8	58.9±3.6	60.2±3.5	59.3±3.4	57.6±3.6	56.5±3.5	55.1±3.8
15	48.5±6.0	53.7±6.0	56.8±4.9	60.1±4.6	61.2±3.3	60.5±3.0	59.3±3.5	58.1±3.5
20	*47.5±7.2*	51.7±6.8	54.4±6.1	58.1±5.5	61.5±3.6	61.7±3.7	60.8±4.1	59.6±3.7
25	*45.7±8.1*	*48.9±7.8*	*51.4±7.3*	55.2±6.6	58.8±5.6	60.7±4.2	60.7±4.7	60.4±5.1
30	*45.3±8.3*	*47.9±8.6*	*49.6±7.9*	53.0±7.0	57.2±6.4	59.2±5.3	60.2±5.2	60.9±5.3
	(b) ANN							
1	48.3±7.0	47.1±6.1	46.0±4.9	43.7±4.5	43.0±5.4	43.2±5.2	40.7±6.4	41.1±5.8
3	55.2±7.5	54.0±8.4	50.7±8.2	49.4±6.7	47.2±6.8	47.0±5.0	44.0±7.3	44.2±5.2
5	53.3±7.5	56.3±9.7	56.1±7.5	53.6±8.9	50.7±6.9	50.6±4.9	47.6±7.6	46.5±7.6
7	51.5±7.5	56.6±8.7	56.4±7.8	55.9±8.9	54.9±6.1	50.9±6.3	49.3±7.0	46.8±7.6
10	47.6±7.2	56.7±5.0	55.2±9.3	56.1±9.3	54.7±8.8	54.3±6.3	51.3±7.7	50.6±5.7
15	*44.8±9.0*	49.1±8.4	53.5±8.4	55.9±10.5	56.5±8.5	56.2±7.1	55.9±7.6	53.7±7.6
20	*41.5±8.9*	47.4±8.9	49.5±11.0	53.7±9.4	57.0±8.1	56.4±9.7	55.3±8.8	52.6±10.0
25	*42.4±8.8*	*44.2±10.2*	47.5±10.3	51.2±10.1	54.3±10.0	57.0±9.0	54.1±11.0	54.4±9.5
30	*41.7±11.0*	*41.6±10.8*	*45.8±10.8*	48.8±11.0	53.2±11.1	54.6±10.8	56.7±7.2	55.4±8.8
	(c) kNN							
1	42.2±4.1	41.1±4.1	39.9±3.4	39.0±4.1	38.1±3.6	38.1±4.5	*37.3±4.3*	*37.2±4.7*
3	44.6±5.2	44.3±5.0	42.7±5.0	41.4±4.9	39.1±3.8	*37.5±4.7*	*37.2±3.2*	*36.9±2.9*
5	43.5±5.6	45.0±5.7	44.3±5.8	42.4±5.8	*40.2±5.5*	*39.1±4.1*	*38.9±4.3*	*38.1±4.3*
7	42.7±5.8	45.3±6.2	44.1±6.6	42.5±6.3	*40.9±5.6*	*40.1±4.8*	*39.0±4.1*	*38.7±3.9*
10	*41.3±6.1*	43.9±6.9	43.9±6.6	43.4±6.6	*41.4±5.4*	*40.3±5.1*	*40.4±4.7*	*39.5±4.1*
15	*39.6±6.9*	*41.8±7.0*	*42.3±7.4*	*42.5±7.1*	*42.1±6.4*	*41.2±5.4*	*41.1±4.3*	*40.2±4.5*
20	*38.6±7.1*	*39.8±7.6*	*40.8±7.5*	*42.2±7.4*	*42.1±6.4*	*41.9±6.0*	*41.4±5.5*	*40.8±5.1*
25	*37.5±8.0*	*38.7±8.0*	*39.5±7.7*	*40.6±7.5*	*41.5±7.0*	*41.9±6.8*	*42.0±5.9*	*41.4±5.2*
30	*36.7±8.7*	*38.0±8.8*	*38.6±8.1*	*40.1±8.1*	*41.6±7.2*	*41.9±7.0*	*41.7±6.2*	*41.8±5.4*

The KNN approach demonstrated significantly lower performance than SVM on the underlying task for two class classification, with the averaged mean lower by -12.7% and the averaged standard deviation higher by 0.7% than those of SVM. When classifying data points into three classes, kNN demonstrated poorer performance than SVM in terms of the averaged mean by -13.9% and showed higher averaged deviation by 1.2%. Results obtained for kNN are statistically significant from the corresponding values of the benchmark model only for short forecast horizons. This technique demonstrates especially weak ability to predict directional price movements for long forecast horizons which noticeably affects the outcomes. The pattern, found using ANN and SVM, can still be observed for kNN, however the system's performance has deteriorated and affected the visibility of the pattern. These results indicate that the pattern, observed for SVM and ANN, that the highest accuracy is achieved when a window size is equal to a forecast horizon, is reproduced for different machine learning approaches and its visibility depends on a performance of an approach.

The prediction accuracy for two class classification is higher than that for three class classification. This outcome is expected because the problem of classifying into three classes is more complicated than classifying into two classes. When the 'No Move' class was added as a possible output, the complexity of the predictive system was increased. The benefits are to avoid making trades when a predicted change in a price of an underlying stock is small. This enhancement is supposed to reduce the number of trades and to increase an average profit from a single trade.

6.2 Winning Rate

Winning rate is also known as a success rate or percentage of profitable trades, it is calculated as a ratio of a number of profitable trades to the total number of trades:

$$WinRatio = \frac{N_{WinTrades}}{N_{total}} \tag{17}$$

where $N_{WinTrades}$ is the number of winning trades that led to a profit and N_{total} is the total number of trades. For two class classification, the winning rate is equal to the prediction accuracy because the total number of trades is equal to the number of data points and therefore the number of winning trades is equal to the number of correct predictions. The winning rate achieved for three class classification is presented in Table V. When comparing results achieved by different machine learning methods with the corresponding values of the prediction accuracy, the following can be concluded: the winning rate for each combination of the window and the forecast horizon is significantly higher than the corresponding value of the prediction accuracy. Especially, the difference between the winning rate and the prediction accuracy for three class classification, averaged over all combinations of a forecast horizon and a window size, is equal to 18.7%, 16.1% and 16.2% in terms of mean values for SVM, ANN and kNN respectively. The standard deviations of the winning rates for three class classification is on average

TABLE V. Averaged winning rate in percentage (%) for three class classification obtained using different classifiers (a) SVM, (b) ANN and (c) kNN

Horizon, days	Window Size, days 3	5	7	10	15	20	25	30
(a) SVM								
1	70.3±4.3	67.2±3.7	63.9±3.7	61.7±3.1	58.0±4.4	58.1±3.5	57.0±3.6	55.3±5.7
3	77.0±4.1	78.8±4.6	73.7±4.6	71.8±4.6	68.6±4.6	67.7±8.0	65.5±7.2	64.5±7.2
5	75.5±4.7	81.1±4.4	77.9±5.1	76.5±5.2	73.4±5.5	71.2±5.2	68.9±5.4	67.0±4.6
7	72.6±5.1	80.5±4.9	80.0±4.9	79.3±5.5	77.8±5.4	74.4±5.8	73.2±6.2	70.6±6.3
10	68.0±5.8	78.3±5.7	79.8±5.8	81.5±5.9	80.8±6.3	78.9±6.7	77.4±7.3	74.7±6.3
15	66.3±8.1	73.7±7.5	77.0±6.5	80.9±6.5	82.0±6.4	81.4±6.5	80.4±7.0	78.6±6.8
20	*60.7±17.8*	69.3±13.4	72.3±13.4	78.5±7.3	82.6±6.2	82.6±6.8	81.4±7.3	80.6±7.1
25	*58.3±17.6*	*64.0±15.7*	69.5±14.0	75.4±8.6	80.1±7.3	82.6±6.2	82.5±7.0	81.7±7.4
30	*58.7±15.3*	*61.3±18.9*	*66.6±14.7*	72.4±9.8	77.9±8.3	81.0±6.8	82.1±6.7	82.2±6.8
(b) ANN								
1	64.2±7.2	61.1±9.8	59.2±4.8	*52.9±13.9*	*53.8±8.4*	55.0±3.8	*50.3±10.9*	*51.7±4.0*
3	71.6±12.5	72.6±10.4	64.8±16.2	65.2±11.0	62.8±6.4	61.0±10.1	58.4±6.5	57.7±5.0
5	69.2±15.6	73.7±17.0	72.8±13.1	68.3±17.2	65.5±15.1	67.4±5.1	61.9±11.9	*58.7±16.9*
7	66.6±15.5	76.1±9.9	75.4±7.5	74.7±10.6	71.8±11.8	69.1±8.1	65.9±12.0	63.6±8.1
10	64.0±8.2	75.7±5.4	72.2±17.2	74.3±14.6	72.7±14.0	69.5±18.6	68.9±13.5	68.8±6.3
15	*57.7±14.7*	*63.2±18.2*	72.0±8.6	74.0±17.6	76.7±9.0	75.6±8.8	73.3±16.7	71.2±13.6
20	*55.5±15.3*	*64.6±9.4*	*64.9±19.3*	70.4±17.5	78.2±9.5	76.1±12.5	71.1±17.9	68.4±19.2
25	*55.6±15.3*	*58.4±17.6*	*64.0±10.6*	*66.7±19.5*	70.6±17.7	75.8±15.3	*68.2±22.8*	74.9±10.4
30	*53.6±17.1*	*56.2±15.4*	*59.5±16.9*	*64.6±15.7*	*68.4±20.4*	74.4±12.0	77.8±8.0	76.0±8.6
(c) kNN								
1	56.5±3.5	54.7±2.8	53.7±2.6	*52.3±2.5*	*51.6±2.3*	*50.8±2.3*	*50.2±1.9*	*51.0±2.2*
3	60.9±5.5	61.3±4.5	58.3±4.2	56.4±3.7	*54.1±3.5*	*52.8±3.4*	*52.6±2.3*	*52.4±2.5*
5	59.3±5.4	62.4±6.2	61.1±5.8	58.4±5.1	*56.1±4.9*	*54.7±3.4*	*53.9±3.5*	*53.3±3.9*
7	*58.1±5.9*	61.9±6.3	61.2±6.7	59.3±6.2	*57.3±5.8*	*56.1±4.4*	*55.4±3.9*	*54.9±3.6*
10	*55.8±6.0*	*60.3±7.7*	60.7±7.3	60.8±7.2	*58.6±6.0*	*57.2±5.9*	*57.1±5.1*	*56.0±4.4*
15	*54.0±6.7*	*57.8±7.7*	*58.5±7.8*	*60.5±7.6*	*60.2±7.0*	*59.1±6.2*	*58.5±5.2*	*57.1±4.9*
20	*52.8±7.2*	*55.4±8.1*	*57.2±8.4*	*59.3±7.8*	*60.5±8.3*	*59.8±7.7*	*59.5±6.6*	*58.9±6.6*
25	*51.7±8.1*	*54.5±8.4*	*56.0±8.3*	*58.4±8.6*	*59.6±8.5*	*60.3±7.9*	*60.6±7.3*	*59.8±7.0*
30	*50.8±9.3*	*53.1±9.6*	*54.5±9.4*	*56.2±9.4*	*59.2±9.4*	*59.9±8.7*	*60.2±8.6*	*59.4±6.9*

higher than those of the prediction accuracy for SVM and ANN methods and slightly lower for the kNN method, and more values fall into the category of statistically insignificant in relation to the benchmark. Winning rate for three class classification is also higher than the winning rate (and the prediction accuracy) for two class classification which is reproducible for all approaches. In particular, the averaged difference in the mean values of winning rates between two and three class classifications is equal to 4.9%, 1.3% and 1.2% for SVM, ANN and kNN respectively. It is worth noticing that the standard deviations concurrently increased by 2.6%, 6.6% and 1.1% respectively. The results demonstrate that, when small price movements are assigned to the third ‘No Move’ class, all considered approaches better distinguish between up and down price movements however the results for different stocks show high variation around the mean value. This indicates that more noise appears in the values of the winning rate. The highest percentage of winning trades equal to 82.6% is reached for the SVM approach for three class classification when predicting for 20 days ahead with the window size equal to 15 or 20 days, and for 25 days ahead with the window size equal to 20 days. These results are comparable to the highest winning rate of 86.55% obtained by Winkowska and Marcinkiewicz in [31] for ANN. The pattern, found for the prediction accuracy, is clearly reproduced for the winning rate.

6.3 Return per Trade

Return per trade is a commonly used metric when the performance of a trading system is evaluated. When the system predicted 'Up' price movement so that an underlying stock was bought at the moment of the prediction and sold at the end of the forecast horizon, the return from this trade was calculated as:

$$R_{t,s} = (C_{t+s} - C_t)/C_t \tag{18}$$

where s is the length of the forecast horizon, C_t is the price at the moment of the prediction t, C_{t+s} is the price at the end of the forecast horizon, $R_{t,s}$ is the return from a trade. When the system predicted 'Down' price movement so that an underlying stock was sold at the moment of prediction and bought back at the end of the forecast horizon, the return from this trade was calculated as:

$$R_{t,s} = (C_t - C_{t+s})/C_t \tag{19}$$

The return was calculated for each trade made during the testing phase. Returns from single trades were averaged over the total number of trades made for each stock. Afterwards, the returns were averaged over 50 stocks for each pair {forecast horizon, window size}. The obtained results are presented in Table VI for two classes and in Table VII for three classes using SVM, ANN and kNN machine learning techniques. The results are similar to those obtained for accuracy and winning rate performance measures in terms of comparison to the benchmark. Values of returns obtained using trading strategies based on predictions from SVM and ANN are mostly significantly higher than those of the benchmark. Returns per trade made with the help of the kNN approach are mostly smaller and/or statistically insignificant than those of the benchmark with a few exceptions. The exceptions refer to short forecast horizons and window size values. The discovered pattern observed for the accuracies and the winning rates is reproduced for returns. Higher returns and smaller standard deviations for each forecast horizon are observed when a window size is set close to a forecast horizon. The pattern is getting less clear for the kNN approach that have small and/or statistically insignificant returns.

TABLE VI. Averaged return per trade in percentage (%) for two class classification obtained using different classifiers (a) SVM, (b) ANN and (c) kNN

Horizon, days	Window Size, days 3	5	7	10	15	20	25	30
	(a) SVM							
1	0.82±0.28	0.70±0.26	0.70±0.27	0.60±0.22	0.39±0.15	0.37±0.16	0.3±0.13	0.28±0.15
3	1.74±0.66	1.75±0.61	1.53±0.55	1.41±0.51	1.22±0.49	1.10±0.40	0.99±0.38	0.95±0.41
5	2.10±0.79	2.38±0.84	2.17±0.78	2.09±0.72	1.94±0.70	1.77±0.61	1.62±0.61	1.51±0.63
7	2.23±0.87	2.75±0.97	2.63±0.94	2.60±0.93	2.53±0.89	2.34±0.79	2.21±0.78	2.06±0.81
10	2.21±1.04	3.00±1.04	3.05±1.08	3.27±1.10	3.21±1.08	3.07±1.08	2.85±0.98	2.67±1.01
15	2.35±1.55	3.29±1.40	3.63±1.41	4.02±1.36	4.19±1.39	4.01±1.31	3.93±1.32	3.80±1.40
20	*2.34±2.01*	3.35±1.65	3.72±1.65	4.38±1.73	4.69±1.66	4.69±1.61	4.64±1.68	4.51±1.73
25	*2.37±2.53*	*3.24±1.93*	3.71±1.87	4.51±1.94	5.09±1.82	5.28±1.94	5.33±1.95	5.25±2.06
30	*2.61±3.12*	*3.27±2.45*	*3.68±2.25*	4.51±2.20	5.22±2.08	5.56±2.16	5.76±2.33	5.77±2.45
	(b) ANN							
1	0.66±0.35	0.59±0.26	0.46±0.24	0.35±0.17	0.25±0.15	0.20±0.16	0.20±0.14	*0.14±0.12*
3	1.68±0.71	1.61±0.69	1.27±0.58	1.15±0.55	1.01±0.51	0.83±0.44	0.69±0.41	0.65±0.37
5	1.94±0.94	2.28±0.93	1.96±0.79	1.83±0.94	1.70±0.80	1.47±0.67	1.29±0.63	1.20±0.52
7	2.04±0.98	2.59±1.10	2.57±1.11	2.51±1.08	2.17±0.99	2.00±0.89	1.85±0.87	1.55±0.82
10	1.92±1.24	2.82±1.15	2.99±1.21	3.08±1.37	2.84±1.39	2.77±1.04	2.50±1.01	2.24±1.00
15	*1.97±1.64*	3.07±1.32	3.28±1.65	3.71±1.79	3.97±1.61	3.81±1.51	3.41±1.37	3.15±1.56
20	*1.80±2.22*	2.85±2.01	3.52±1.75	4.08±1.94	4.39±2.10	4.10±2.15	4.24±1.98	3.77±2.11
25	*2.03±2.76*	*2.91±2.07*	*3.44±2.09*	4.19±2.16	4.49±2.38	4.82±2.48	4.97±2.36	4.31±2.31
30	*2.05±3.30*	*2.77±2.73*	*3.11±2.58*	*3.79±2.76*	4.89±2.56	4.77±2.88	4.95±3.08	5.10±2.99
	(c) kNN							
1	0.28±0.18	0.20±0.15	0.15±0.12	*0.10±0.10*	*0.09±0.09*	*0.04±0.09*	*0.04±0.08*	*0.04±0.08*
3	0.79±0.50	0.76±0.46	0.58±0.37	0.48±0.33	*0.33±0.28*	*0.20±0.20*	*0.20±0.19*	*0.19±0.20*
5	0.98±0.66	1.15±0.66	1.01±0.63	0.79±0.53	*0.61±0.47*	*0.42±0.33*	*0.36±0.35*	*0.31±0.32*
7	*1.00±0.77*	1.30±0.86	1.27±0.78	1.14±0.78	*0.81±0.6*	*0.64±0.49*	*0.60±0.40*	*0.51±0.46*
10	*0.96±1.04*	1.39±1.02	1.48±1.01	1.41±0.99	*1.13±0.86*	*1.02±0.76*	*0.92±0.65*	*0.76±0.57*
15	*1.05±1.50*	*1.41±1.33*	*1.54±1.31*	*1.70±1.35*	*1.68±1.32*	*1.47±1.06*	*1.44±1.05*	*1.25±0.88*
20	*1.08±2.08*	*1.43±1.75*	*1.59±1.65*	*1.91±1.72*	*1.92±1.68*	*1.85±1.43*	*1.86±1.36*	*1.69±1.22*
25	*1.08±2.72*	*1.56±2.18*	*1.57±1.91*	*1.90±1.95*	*2.05±1.95*	*2.16±1.72*	*2.15±1.74*	*1.99±1.57*
30	*1.10±3.19*	*1.39±2.51*	*1.59±2.26*	*1.86±2.25*	*2.23±2.23*	*2.43±2.23*	*2.41±2.16*	*2.08±1.83*

Note that very high returns are most likely to be obtained due to the simplified strategy that does not include transaction costs and other effects that typically reduce the profit. These high return values are unlikely to appear in practice, but they do indicate a potential arbitrage.

6.4 Sharpe Ratio

Sharpe Ratio is used to measure risk-adjusted performance of a trading system which was proposed by Sharpe and called "reward-to-variability" ratio [38]. It measures the excess return, also called a risk premium, compared with the risk free rate, in terms of their absolute values, and then compared to the overall risk measured by returns' standard deviation. The Sharpe ratio is commonly used by investment funds to measure a portfolio performance. It enables to relatively compare the performance of different portfolios including not well-diversified ones which corresponds to our case [39]. The ratio is computed by calculating an average return obtained from generated trades and its standard deviation and is required to be annualized. The commonly used formula to calculated Sharpe Ratio is:

TABLE VII. Averaged return per trade in percentage (%) for three class classification obtained using different classifiers (a) SVM, (b) ANN and (c) kNN

Horizon, days	Window Size, days							
	3	5	7	10	15	20	25	30
	(a) SVM							
1	1.09±0.27	0.96±0.24	0.82±0.23	0.71±0.21	0.55±0.22	0.55±0.20	0.45±0.21	0.40±0.20
3	2.27±0.49	2.35±0.39	2.03±0.47	1.86±0.41	1.69±0.38	1.59±0.50	1.46±0.52	1.34±0.48
5	2.74±0.56	3.20±0.49	2.94±0.52	2.84±0.45	2.65±0.44	2.41±0.44	2.24±0.49	2.10±0.46
7	2.85±0.71	3.70±0.64	3.61±0.55	3.54±0.63	3.39±0.6	3.14±0.65	3.09±0.67	2.88±0.79
10	2.83±0.91	4.12±0.65	4.23±0.68	4.45±0.70	4.37±0.7	4.20±0.63	4.06±0.62	3.79±0.66
15	3.05±1.20	4.27±1.12	4.77±1.02	5.34±0.99	5.58±1.1	5.47±0.99	5.33±0.98	5.18±1.02
20	3.06±1.60	4.31±1.65	4.8±1.54	5.78±1.32	6.41±1.18	6.37±1.31	6.25±1.36	6.2±1.33
25	*3.01±1.92*	3.87±1.93	4.71±2.05	5.78±1.65	6.73±1.65	7.20±1.61	7.22±1.66	7.10±1.62
30	*3.33±2.37*	*4.00±2.47*	4.78±2.52	5.78±2.16	7.08±2.10	7.72±1.88	7.96±1.94	8.06±2.01
	(a) SVM							
1	0.76±0.42	0.66±0.28	0.52±0.31	0.38±0.24	0.32±0.19	0.32±0.20	0.18±0.2	*0.15±0.17*
3	1.94±0.76	1.79±0.79	1.47±0.84	1.37±0.62	1.18±0.56	1.05±0.49	0.80±0.57	0.79±0.47
5	2.34±1.02	2.6±1.06	2.55±1.01	2.17±1.20	1.88±0.90	1.88±0.56	1.47±0.92	*1.14±1.92*
7	2.44±1.17	3.15±1.12	3.12±1.13	2.92±1.54	2.84±0.98	2.41±0.95	2.25±0.95	1.81±1.22
10	2.25±1.2	3.69±0.83	3.43±1.45	3.56±1.54	3.37±1.39	3.37±1.29	2.89±1.42	2.84±1.02
15	2.19±1.59	3.09±2.00	3.99±1.69	4.52±2.01	4.59±1.66	4.49±1.67	4.39±1.73	3.97±1.48
20	*1.67±2.10*	3.19±2.15	3.65±2.17	4.62±2.43	5.43±2.28	4.99±3.24	4.52±2.73	4.27±2.62
25	*2.09±2.31*	*2.88±2.30*	3.63±2.47	4.61±2.53	5.12±2.86	6.03±2.71	5.25±3.26	5.68±2.42
30	*2.04±3.13*	*2.32±3.73*	*3.29±3.16*	4.67±2.94	5.61±3.13	6.02±3.35	6.96±2.42	6.52±2.58
	(c) kNN							
1	0.37±0.20	0.29±0.18	0.22±0.12	0.15±0.11	*0.13±0.10*	*0.06±0.11*	*0.03±0.10*	*0.07±0.12*
3	1.01±0.55	0.99±0.48	0.78±0.42	0.60±0.37	*0.43±0.32*	*0.25±0.27*	*0.24±0.23*	*0.22±0.22*
5	1.15±0.65	1.43±0.74	1.22±0.65	1.05±0.61	0.76±0.54	*0.57±0.40*	*0.52±0.37*	*0.43±0.39*
7	1.22±0.77	1.63±0.82	1.52±0.88	1.36±0.89	1.05±0.72	*0.84±0.63*	*0.74±0.47*	*0.71±0.49*
10	*1.07±0.87*	1.71±1.17	1.78±1.14	1.75±1.14	1.43±0.96	*1.14±0.85*	*1.12±0.67*	*1.01±0.65*
15	*1.07±1.18*	*1.74±1.48*	*1.84±1.52*	2.12±1.52	2.10±1.49	*1.82±1.22*	*1.75±1.07*	*1.51±0.99*
20	*0.92±1.44*	*1.50±1.73*	*1.75±1.80*	*2.18±1.83*	*2.36±1.89*	*2.35±1.66*	*2.25±1.43*	*2.08±1.45*
25	*0.80±1.90*	*1.43±2.03*	*1.74±2.05*	*2.18±2.16*	*2.54±2.21*	*2.79±2.03*	*2.84±1.84*	*2.66±1.93*
30	*0.70±2.34*	*1.28±2.48*	*1.59±2.35*	*1.97±2.43*	*2.72±2.50*	*2.94±2.46*	*2.95±2.51*	*2.77±2.09*

$$S_p = \sqrt{T}\,\frac{E(R_P) - R_F}{\sigma(R_P)} \tag{20}$$

where $E(R_P)$ is an expected portfolio return, R_F is a risk free rate, $\sigma(R_P)$ is a portfolio standard deviation, T is the number of periods per year where a period corresponds to a number of days returns are computed for. In the current study, the simplified case when the risk free rate is equal to zero was considered. This choice was made based on [26], [27].

TABLE VIII. Averaged Sharpe ratio computed for two class classification obtained using different classifiers (a) SVM, (b) ANN and (c) kNN

Horizon, days	Window Size, days							
	3	5	7	10	15	20	25	30
	(a) SVM							
1	6.46±0.99	5.40±0.89	5.38±0.95	4.59±0.86	2.91±0.70	2.69±0.63	2.25±0.65	2.05±0.7
3	4.84±0.75	4.95±0.66	4.18±0.71	3.81±0.67	3.21±0.71	2.89±0.54	2.56±0.51	2.41±0.56
5	3.52±0.66	4.15±0.57	3.72±0.65	3.54±0.53	3.21±0.50	2.91±0.49	2.61±0.49	2.38±0.52
7	2.63±0.63	3.45±0.53	3.29±0.58	3.24±0.62	3.13±0.53	2.86±0.52	2.65±0.46	2.41±0.49
10	1.82±0.76	2.63±0.44	2.71±0.53	2.98±0.56	2.91±0.54	2.74±0.52	2.50±0.50	2.31±0.58
15	1.25±0.86	1.84±0.53	2.10±0.47	2.45±0.5	2.58±0.46	2.44±0.46	2.37±0.46	2.27±0.54
20	*0.94±0.90*	1.38±0.54	1.58±0.48	1.94±0.5	2.14±0.43	2.16±0.43	2.13±0.48	2.08±0.53
25	*0.75±0.92*	*1.05±0.57*	1.23±0.47	1.57±0.49	1.85±0.40	1.94±0.40	1.97±0.43	1.95±0.50
30	*0.70±0.95*	*0.88±0.63*	*1.00±0.51*	1.29±0.50	1.56±0.40	1.70±0.39	1.79±0.45	1.81±0.50
	(b) ANN							
1	5.11±1.91	4.44±1.23	3.32±1.24	2.60±0.97	1.84±0.89	1.38±0.92	1.38±0.97	*1.05±0.77*
3	4.57±0.96	4.44±1.06	3.48±1.04	2.93±0.91	2.55±0.78	2.12±0.79	1.77±0.79	1.59±0.70
5	3.12±0.98	3.89±0.75	3.34±0.88	2.96±1.06	2.69±0.75	2.32±0.78	2.03±0.7	1.89±0.55
7	2.34±0.82	3.17±0.94	3.12±0.8	3.03±0.75	2.59±0.82	2.32±0.78	2.12±0.7	1.81±0.69
10	1.52±0.92	2.43±0.64	2.61±0.64	2.7±0.78	2.42±0.88	2.40±0.51	2.1±0.56	1.84±0.55
15	*1.02±0.91*	1.72±0.55	1.86±0.68	2.18±0.82	2.36±0.62	2.23±0.5	2.01±0.67	1.77±0.79
20	*0.69±0.96*	*1.10±0.73*	1.46±0.52	1.79±0.68	1.93±0.72	1.86±0.82	1.88±0.69	1.64±0.77
25	*0.59±0.96*	*0.90±0.59*	*1.09±0.56*	1.39±0.57	1.58±0.66	1.70±0.72	1.77±0.58	1.55±0.66
30	*0.50±0.96*	*0.66±0.69*	*0.82±0.64*	*1.03±0.68*	1.41±0.57	1.41±0.71	1.48±0.76	1.55±0.70
	(c) kNN							
1	2.01±1.12	1.45±0.91	1.07±0.71	*0.83±0.76*	*0.61±0.61*	*0.31±0.70*	*0.33±0.66*	*0.23±0.59*
3	1.90±0.98	1.85±0.77	1.40±0.69	1.15±0.63	*0.76±0.57*	*0.50±0.49*	*0.49±0.50*	*0.44±0.45*
5	1.44±0.90	1.73±0.66	1.45±0.63	1.15±0.63	*0.88±0.58*	*0.64±0.47*	*0.53±0.54*	*0.44±0.46*
7	*1.06±0.87*	1.38±0.76	1.37±0.64	1.18±0.64	*0.87±0.56*	*0.69±0.45*	*0.64±0.44*	*0.54±0.42*
10	*0.72±0.92*	*1.04±0.68*	1.12±0.62	*1.07±0.66*	*0.85±0.54*	*0.77±0.51*	*0.69±0.46*	*0.56±0.34*
15	*0.53±0.91*	*0.69±0.65*	*0.77±0.61*	*0.86±0.62*	*0.85±0.57*	*0.75±0.46*	*0.71±0.47*	*0.63±0.37*
20	*0.40±0.96*	*0.52±0.67*	*0.59±0.60*	*0.72±0.62*	*0.74±0.58*	*0.71±0.50*	*0.70±0.47*	*0.64±0.35*
25	*0.30±0.98*	*0.43±0.68*	*0.45±0.57*	*0.57±0.58*	*0.61±0.55*	*0.66±0.49*	*0.67±0.5*	*0.59±0.38*
30	*0.26±0.97*	*0.31±0.66*	*0.37±0.56*	*0.46±0.59*	*0.56±0.52*	*0.62±0.51*	*0.63±0.52*	*0.54±0.37*

Tables VIII and IX show the Sharpe ratio values computed for two and three class classifications respectively. The overall performance of the predictive system in terms of Sharpe ratio is similar to that of other metrics previously presented. It corresponds to both the visibility of the pattern and the comparison to the benchmark in terms of difference in mean values, standard deviations and statistical significance of results. In [40], Sharpe ratio value, computed for the model based on the price data, varies from 2 to 8 decreasing with an increase in a forecast horizon. For the largest forecast horizon equal to 250 minutes which is approximately half of a trading day, Sharpe Ratio is close to 3. Regardless of the fact that the current research is done using not intraday but daily data, similar behaviour can be noticed: Sharpe ratio tends to be smaller for larger forecast horizons. The highest value of Sharpe ratio of 7.58 is reached for one day ahead forecasting with the window size equal to three days when classifying into three classes. The values obtained for three class classification are higher on average than the values obtained for two classes. It confirms that adding the supplementary class ‘No Move’ improves the performance of the trading system in terms of Sharpe ratio performance measure. For most forecast horizons, the highest values of Sharpe ratio are reached when a window size approximately matches a horizon. This behaviour is clearly visible for predictive systems based on the SVM and ANN techniques. For the kNN method, most of the Sharpe ratio values appear to be lower than and/or statistically insignificant from the benchmark. With a decrease in the mean values, values of standard deviations tend to increase. It is particularly visible when a short forecast horizon and a long window size, or a long forecast horizon and a short window size are used. This emphasizes the idea that when a machine learning technique is unable to infer relevant information from the input, the forecasting results are significantly affected by the noise.

TABLE IX. Averaged Sharpe ratio computed for three class classification obtained using different classifiers (a) SVM, (b) ANN and (c) kNN

Horizon, days	Window Size, days							
	3	5	7	10	15	20	25	30
	(a) SVM							
1	7.58±1.67	6.48±1.36	5.31±1.21	4.48±1.14	3.4±1.74	3.27±1.07	2.78±1.45	2.45±2.24
3	5.78±1.07	6.19±1.34	5.09±1.12	4.51±1.01	3.93±1.00	3.76±2.24	3.32±1.85	2.83±1.13
5	4.29±0.95	5.34±1.17	4.69±1.06	4.46±1.09	3.98±1.00	3.52±0.94	3.24±1.07	2.97±0.91
7	3.09±0.66	4.40±0.93	4.29±1.21	4.14±1.03	3.88±0.91	3.49±0.98	3.42±1.18	3.08±1.05
10	2.11±0.69	3.49±1.05	3.64±1.14	3.91±1.13	3.86±1.22	3.63±1.21	3.49±1.39	3.09±1.01
15	1.61±0.83	2.36±0.85	2.74±0.86	3.22±1.00	3.44±1.14	3.31±1.07	3.21±1.17	2.99±0.89
20	*1.23±1.02*	1.88±1.56	2.06±0.95	2.59±0.89	3.01±1.00	3.01±1.03	2.91±1.04	2.84±0.91
25	*0.91±0.72*	1.22±0.68	1.71±1.17	2.05±0.87	2.48±0.88	2.74±0.99	2.74±1.00	2.71±1.01
30	*0.82±0.66*	*1.05±0.87*	1.39±1.34	1.69±0.91	2.17±0.87	2.47±0.97	2.55±0.97	2.61±1.04
	(b) ANN							
1	5.35±2.43	4.49±1.65	3.44±1.91	2.51±1.54	2.10±1.35	2.11±1.62	*1.20±1.37*	*0.91±1.22*
3	5.02±1.74	4.65±2.25	3.61±2.28	3.38±1.43	2.75±1.36	2.46±1.15	1.76±1.30	1.78±1.09
5	3.53±1.31	4.32±1.68	4.09±1.58	3.40±2.00	2.93±1.30	2.96±1.11	2.2±1.28	1.95±1.50
7	2.63±1.14	3.83±1.50	3.62±1.11	3.56±1.70	3.26±1.12	2.74±1.19	2.44±1.06	1.94±1.17
10	1.74±0.92	3.16±0.86	2.94±1.32	2.92±2.56	2.95±1.36	2.80±1.10	2.42±1.18	2.28±0.68
15	*1.02±0.78*	1.64±1.00	2.22±1.03	2.69±1.15	2.77±1.10	2.61±1.00	2.59±1.04	2.23±1.09
20	*0.62±0.83*	1.27±0.88	1.59±1.02	1.94±1.00	2.72±1.88	2.41±1.26	*1.56±3.59*	1.80±1.23
25	*0.60±0.79*	*0.90±0.76*	*1.10±0.78*	1.57±0.86	1.78±0.99	2.22±1.17	1.81±1.11	2.02±0.84
30	*0.45±0.93*	*0.54±0.89*	*0.82±0.82*	*1.21±0.92*	1.61±0.90	1.82±0.99	2.15±0.83	1.99±0.75
	(c) kNN							
1	2.46±1.15	1.88±0.97	1.51±0.74	*1.07±0.78*	*0.86±0.62*	*0.43±0.82*	*0.23±0.75*	*0.43±0.77*
3	2.34±1.05	2.32±0.95	1.80±0.81	1.38±0.71	*0.96±0.68*	*0.58±0.62*	*0.57±0.55*	*0.53±0.56*
5	1.61±0.85	2.07±0.93	1.77±0.85	1.46±0.72	*1.08±0.82*	*0.82±0.53*	*0.73±0.53*	*0.64±0.64*
7	1.23±0.72	1.68±0.73	1.59±0.86	1.33±0.74	*1.10±0.82*	*0.89±0.66*	*0.74±0.45*	*0.76±0.46*
10	*0.74±0.58*	1.23±0.76	1.30±0.73	1.28±0.77	*1.06±0.65*	*0.87±0.61*	*0.84±0.53*	*0.75±0.43*
15	*0.46±0.56*	*0.79±0.65*	*0.85±0.69*	*1.04±0.72*	*1.05±0.67*	*0.90±0.53*	*0.86±0.46*	*0.74±0.41*
20	*0.28±0.54*	*0.50±0.61*	*0.63±0.66*	*0.81±0.63*	*0.89±0.64*	*0.88±0.56*	*0.84±0.47*	*0.78±0.45*
25	*0.17±0.57*	*0.37±0.59*	*0.48±0.59*	*0.63±0.62*	*0.75±0.60*	*0.84±0.58*	*0.86±0.47*	*0.78±0.46*
30	*0.10±0.61*	*0.25±0.63*	*0.34±0.59*	*0.45±0.60*	*0.67±0.60*	*0.75±0.56*	*0.76±0.55*	*0.71±0.44*

6.5 Aggregated results

For comparison purposes, results from Tables III - IX are aggregated and the highest values of performance measures achieved for each forecasting horizon by the SVM, ANN and kNN machine learning approaches and the buy-and-hold strategy are shown in Table X. The highest value of a performance metric reached for the two and three class classifications is highlighted in bold, and those values consistently refer to the SVM approach, which outperforms ANN, kNN and the baseline buy-and-hold method in terms of every considered performance measure for all horizons. In turn, ANN outperformed both kNN and buy-and-hold strategy, and kNN outperformed the buy-and-hold strategy for short horizon of 1-10 trading days and underperformed it for long horizons of 15-30 trading days. The highest prediction accuracy of 75.43% and 61.71% is obtained by SVM when predicting a price change in 15 trading days for two class classification and in 20 trading days for three class classification. Values of the winning rate are equal to those of prediction accuracy for two class classification but differ from them for three class classification, because only predictions of "Up" price movements were regarded as a signal for entering into trade when computing winning rate. There is an important observation that winning rates achieved for three class classification are higher than those achieved when classifying into two classes. As discussed in Section 6.2, these results confirm that introducing the "No Move" class increases the percentage of accurate predictions of "Up" and "Down" classes and enhances the profitability of a trading system utilising those predictions in trading. When predicting for three classes, the winning rate generally increases with an increase in forecasting horizon reaching 82.64% using SVM for horizon equal to 20 trading days.

Returns obtained per simulated trade are complicated to compare across different forecasting horizons because investment horizons of the simulated trades differ and therefore a trade for a shorter period is more likely to lead to a smaller return. The benefit of trading for shorter horizons is that once the trade is completed, money/assets can be reinvested/used in further trading to gain extra profit. When trading for long periods, money/assets are locked within the trade for the duration of the investment period. Therefore, to compare the return obtained for different horizons, they should be adjusted for the period of investment. Additionally, the transaction costs introduce more complications into the adjustment process. These costs depend on many factors such as the exchanges where trades are settled and the financial intermediary used to access exchanges. Financial institutions identified as market makers are able to trade with lower transaction costs than individual market participants. Therefore, the adjustment made to account for transaction costs should differ for different market participants. Accordingly, taking into account these complications, returns in this paper are not compared across different forecast horizons.

TABLE X. The highest prediction accuracy, return per trade, winning rate and Sharpe ratio achieved for multiple forecasting horizons by the SVM, ANN and kNN classifiers. Results are aggregated from Tables III – IX.

Step, days	2 classes classification				3 classes classification			
	SVM	ANN	kNN	Buy&Hold	SVM	ANN	kNN	Buy&Hold
	Accuracy, %							
1	**67.45%**	63.65%	55.85%	51.68%	**52.62%**	48.29%	42.22%	35.10%
3	**72.84%**	71.00%	58.93%	53.97%	**58.83%**	55.20%	44.63%	36.83%
5	**74.42%**	72.91%	60.26%	55.03%	**60.91%**	56.31%	44.98%	37.82%
7	**74.21%**	72.20%	59.78%	56.37%	**60.17%**	56.60%	45.30%	38.75%
10	**74.46%**	71.95%	59.03%	57.58%	**60.22%**	56.71%	43.94%	39.77%
15	**75.43%**	73.21%	58.47%	59.45%	**61.20%**	56.53%	42.53%	41.26%
20	**74.64%**	71.52%	58.53%	60.85%	**61.71%**	56.99%	42.23%	42.35%
25	**74.44%**	71.81%	57.74%	61.66%	**60.74%**	57.02%	42.02%	43.17%
30	**74.36%**	71.17%	57.82%	62.32%	**60.89%**	56.68%	41.86%	43.62%
	Winning rate, %							
1	**70.31%**	64.17%	56.48%	51.68%	**70.31%**	64.17%	56.48%	51.68%
3	**78.78%**	72.56%	61.28%	53.97%	**78.78%**	72.56%	61.28%	53.97%
5	**81.09%**	73.67%	62.39%	55.03%	**81.09%**	73.67%	62.39%	55.03%
7	**80.52%**	76.07%	61.91%	56.37%	**80.52%**	76.07%	61.91%	56.37%
10	**81.47%**	75.73%	60.75%	57.58%	**81.47%**	75.73%	60.75%	57.58%
15	**82.04%**	76.72%	60.51%	59.45%	**82.04%**	76.72%	60.51%	59.45%
20	**82.64%**	78.23%	60.46%	60.85%	**82.64%**	78.23%	60.46%	60.85%
25	**82.55%**	75.82%	60.56%	61.66%	**82.55%**	75.82%	60.56%	61.66%
30	**82.23%**	77.83%	60.18%	62.32%	**82.23%**	77.83%	60.18%	62.32%
	Return per trade, %							
1	**0.82%**	0.66%	0.28%	0.11%	**1.09%**	0.76%	0.37%	0.11%
3	**1.75%**	1.68%	0.79%	0.31%	**2.35%**	1.94%	1.01%	0.31%
5	**2.38%**	2.28%	1.15%	0.51%	**3.20%**	2.60%	1.43%	0.51%
7	**2.75%**	2.59%	1.30%	0.70%	**3.70%**	3.15%	1.63%	0.70%
10	**3.27%**	3.08%	1.48%	0.98%	**4.45%**	3.69%	1.78%	0.98%
15	**4.19%**	3.97%	1.70%	1.51%	**5.58%**	4.59%	2.12%	1.51%
20	**4.69%**	4.39%	1.92%	2.05%	**6.41%**	5.43%	2.36%	2.05%
25	**5.33%**	4.97%	2.16%	2.58%	**7.22%**	6.03%	2.84%	2.58%
30	**5.77%**	5.10%	2.43%	3.11%	**8.06%**	6.96%	2.95%	3.11%
	Sharpe ratio							
1	**6.46**	5.11	2.01	0.80	**7.58**	5.35	2.46	0.80
3	**4.95**	4.57	1.90	0.80	**6.19**	5.02	2.34	0.80
5	**4.15**	3.89	1.73	0.81	**5.34**	4.32	2.07	0.81
7	**3.45**	3.17	1.38	0.81	**4.40**	3.83	1.68	0.81
10	**2.98**	2.70	1.12	0.81	**3.91**	3.16	1.30	0.81
15	**2.58**	2.36	0.86	0.83	**3.44**	2.77	1.05	0.83
20	**2.16**	1.93	0.74	0.86	**3.01**	2.72	0.89	0.86
25	**1.97**	1.77	0.67	0.87	**2.74**	2.22	0.86	0.87
30	**1.81**	1.55	0.63	0.89	**2.61**	2.15	0.76	0.89

Nevertheless, returns are useful for comparing predictive performance achieved within a forecast horizon. For instance, when the trades are simulated based on the predictions of price movements on the next trading day, average returns per single trade equal 0.82%, 0.66% and 0.28% for two classes and 1.09%, 0.76% and 0.37% for three classes respectively, using the SVM approach. When employing the buy-and-hold strategy for the same trading days, only 0.11% return per trade can be gained. Therefore, there is an obvious improvement in making one-day investments based on the designed predictive system comparing to the simple buy-and-hold strategy, and the highest results are achieved by SVM. For the 30 days forecasting the predictive system generates returns of 5.77%, 5.10% and 2.43% for two class classification, and of 8.06%, 6.96% and 2.95% for three class classification. The simple buy-and-hold approach gains the return of 3.11% which outperforms kNN but underperforms SVM and ANN. The two latter approaches showed a progressive improvement comparing to the baseline approach.

In Table X, Sharpe ratio values steadily decrease with increases in forecasting horizon approaching the buy-and-hold values. This behaviour indicates that despite the fact that promising values of forecasting accuracy are achieved by multiple horizons, the long-term trading strategy that invests resources for long horizons would yield less profit than a short-term trading strategy that follows recent changes in the market state and reinvests resources according to the newly appeared information. It is worth noting that Sharpe ratio values produced by the buy-and-hold method do not show high variation in values for different forecasting horizons and lie in a range (0.80, 0.89). The Sharpe ratio values produced by the predictive system converge to this range with an increase in horizon.

7. CONCLUSION AND FUTURE WORK

The main contribution of this research paper is the detailed investigation of the dependency of the financial forecasting system’s performance on the choice of a forecasting horizon and a window size, a parameter used for calculation of many TIs. The experiments discover a strong dependency of the system performance on the combination of the window size and the forecast horizon. The following pattern was observed: the highest prediction performance was achieved when the window size is approximately equal to the horizon which the predictive system is designed to forecast. The presence of the pattern depends on the ability of a machine learning technique to infer relevant information from the input data. It gives a simple solution for setting initial values for the window size parameter depending on the forecast horizon selected.

The pattern was investigated using a number of performance metrics. Prediction accuracy tests the pattern from a classification point of view: how well the system is able to classify data points based on the computed TIs taken as an input. Average return per trade, Sharpe ratio and winning ratio assess the performance of the predictive system in terms of the risk taken and the reward received. All the considered performance measures have demonstrated that the discovered pattern persists and its visibility depends on the overall performance of the system under the specified conditions. The goal was to predict the direction of an upcoming change in a stock price for forecast horizons from 1 to 30 trading. Three well-established machine learning techniques were employed for analysis: SVM, ANN and kNN. The pattern is clearly visible for SVM: the highest performance is obtained when the window size is approximately equal to the horizon. The ANN approach shows slightly worse prediction performance than SVM for all the measures, but the same pattern can be easily seen in the results obtained from ANN. The prediction performance of the kNN approach is low, the pattern is still visible however its occurrence is significantly affected by the low performance. Fifty stocks were analysed in this study, which is a number sufficient for conducting statistical testing. The statistical analysis has shown that the results are mostly significant for the SVM and ANN techniques, and these techniques showed significantly higher results than those of kNN and the benchmark. This proves that the discovered pattern does not appear by chance. The prediction accuracy for two class classification is higher than for three class classification, however the returns per trade, Sharpe ratios and winning rates are higher for three class classification than for two classes. It proves the fact that inclusion of the ‘No Move’ class improves the prediction performance of a system in terms of trading profitability by avoiding “noisy” trades that do not lead to significant gains; it reduces the number of false trading alarms and increases an average return per trade.

In summary, the proposed research discovers a correlation between the window size and the horizon, which suggests that selecting the proper window size for calculating TIs helps improve the accuracy significantly when creating a financial forecasting system based on TA. The highest system performance for each forecast horizon value is reached when the window size is approximately equal to the horizon, and the visibility of the pattern depends on the ability of the applied machine learning technique to extract relevant information from the input data. The revealed pattern can be utilized for selecting parameter values of the TIs when developing a predictive approach. Presumably, the optimal values of the window sizes for different indicators are likely to be different from each other. Setting all window parameters to the value of a forecast horizon may give a good initial starting point from which a distinct algorithm may adjust a window size for each of the TIs separately. The process of the subsequent adjustment of indicators' parameters is a direction of further research. Within the framework of the further research, the reproduction of the pattern and other window size effects can be explored further for predicting future values of stock prices. This may provide a better insight into the nature of the pattern. Additionally, verifying whether the pattern is reproducible for other financial assets such as currencies or commodities can shed light on the question whether the pattern can be applied to those markets.

8. APPENDIX

The following 50 randomly selected stocks were analysed: AA, AET, ALXN, AMT, AVY, BBT, BK, CA, CAM, CCE, CNX, COF, COH, COL, D, DHR, DVA, ESV, FCX, GIS, GPS, HAR, HPQ, IBM, IP, IR, KMB, KMX, LLY, MAC, MMC, MO, MSFT, MYL, NTAP, PCAR, PDCO, PEP, PKI, PNW, POM, PRGO, ROST, RSG, SJM, SLB, SNDK, TER, TGT, TRV.

9. ACKNOWLEDGEMENT

This research is supported by the companies and organizations involved in the Northern Ireland Capital Markets Engineering Research Initiative (InvestNI, Citi, First Derivatives, Kofax, Fidessa, NYSE Technologies, ILEX, Queen's University of Belfast and Ulster University).

REFERENCES

[1] C.-Y. Yeh, C.-W. Huang, and S.-J. Lee, "A multiple-kernel support vector regression approach for stock market price forecasting," *Expert Systems with Applications*, vol. 38, no. 3, pp. 2177–2186, Mar. 2011.

[2] G. S. Atsalakis and K. P. Valavanis, "Surveying stock market forecasting techniques – Part II: Soft computing methods," *Expert systems with applications*, vol. 36, no. 3, pp. 5932–5941, Apr. 2009.

[3] D. J. Bodas-Sagi, P. Fernández-Blanco, J. I. Hidalgo, and F. J. Soltero-Domingo, "A parallel evolutionary algorithm for technical market indicators optimization," *Natural computing*, vol. 12, no. 2, pp. 195–207, Sep. 2012.

[4] F. Andrade de Oliveira, L. Enrique Zarate, M. de Azevedo Reis, and C. Neri Nobre, "The use of artificial neural networks in the analysis and prediction of stock prices," in *Proceedings of the IEEE International conference on systems, man, and cybernetics*, 2011, pp. 2151–2155.

[5] E. Fama, "The behavior of stock-market prices," *Journal of business*, vol. 38, no. 1, pp. 34–105, 1965.

[6] A. W. Lo, "Reconciling efficient markets with behavioral finance: the adaptive markets hypothesis," *Journal of Investment Consulting*, vol. 7, no. 2, pp. 21–44, 2005.

[7] G. Friesen and P. a. Weller, "Quantifying cognitive biases in analyst earnings forecasts," *Journal of Financial Markets*, vol. 9, no. 4, pp. 333–365, Nov. 2006.

[8] A. Urquhart and R. Hudson, "Efficient or adaptive markets? Evidence from major stock markets using very long run historic data," *International Review of Financial Analysis*, vol. 28, pp. 130–142, Jun. 2013.

[9] C.-H. Park and S. H. Irwin, "What Do We Know About the Profitability of Technical Analysis?," *Journal of Economic Surveys*, vol. 21, no. 4, pp. 786–826, Sep. 2007.

[10] J. H. Kim, A. Shamsuddin, and K.-P. Lim, "Stock return predictability and the adaptive markets hypothesis: Evidence from century-long U.S. data," *Journal of Empirical Finance*, vol. 18, no. 5, pp. 868–879, Dec. 2011.

[11] N. Taylor, "The rise and fall of technical trading rule success," *Journal of Banking & Finance*, vol. 40, pp. 286–302, Mar. 2014.

[12] P. Fernández-Blanco, D. J. Bodas-Sagi, F. J. Soltero-Domingo, and J. I. Hidalgo, "Technical market indicators optimization using evolutionary algorithms," in *Proceedings of the 10th annual genetic and evolutionary computation onference,*, 2008, pp. 1851–1857.

[13] D. J. Bodas-Sagi, P. Fernández-Blanco, J. I. Hidalgo, F. J. Soltero-Domingo, and J. L. Risco-Martin, "Multiobjective optimization of technical market indicators," in *Proceedings of the 11th annual genetic and evolutionary computation onference*, 2009, pp. 1999–2004.

[14] J. J. Murphy, *Technical analysis of the financial markets*. Prentice Hall, London, 1998.

[15] R. Lee, "iJADE Stock Advisor: An Intelligent Agent-Based Stock Prediction System Using the Hybrid RBF Recurrent Network," *IEEE Transactions on Systems, Man and Cybernetics-Part A: System and Humancs*, vol. 34, no. 3, pp. 421–428, 2004.

[16] F. E. H. Tay and L. Cao, "Application of support vector machines in financial time series forecasting," *Omega*, vol. 29, pp. 309–317, 2001.

[17] L. J. Cao and F. E. H. Tay, "Support vector machine with adaptive parameters in financial time series forecasting," *IEEE Transactions on Neural Networks*, vol. 14, no. Nov. 2003, pp. 1506–1518, 2003.

[18] K. Kim, "Financial time series forecasting using support vector machines," *Neurocomputing*, vol. 55, no. 1–2, pp. 307–319, Sep. 2003.

[19] W. Huang, Y. Nakamori, and S.-Y. Wang, "Forecasting stock market movement direction with support vector machine," *Computers & operations research*, vol. 32, no. 10, pp. 2513–2522, Oct. 2005.

[20] Y. Kara, M. Acar Boyacioglu, and Ö. K. Baykan, "Predicting direction of stock price index movement using artificial neural networks and support vector machines: the sample of the Istanbul Stock Exchange," *Expert systems with applications*, vol. 38, no. 5, pp. 5311–5319, May 2011.

[21] J. Arroyo and C. Maté, "Forecasting histogram time series with k-nearest neighbours methods," *International Journal of Forecasting*, vol. 25, no. 1, pp. 192–207, Jan. 2009.

[22] L. A. Teixeira and A. L. I. de Oliveira, "A method for automatic stock trading combining technical analysis and nearest neighbor classification," *Expert Systems with Applications*, vol. 37, no. 10, pp. 6885–6890, Oct. 2010.

[23] R. Khemchandani and S. Chandra, "Regularized least squares fuzzy support vector regression for financial time series forecasting," *Expert Systems with Applications*, vol. 36, no. 1, pp. 132–138, Jan. 2009.

[24] Y. Wang and I. Choi, "Market Index and Stock Price Direction Prediction using Machine Learning Techniques: An empirical study on the KOSPI and HSI," *Elsevier, arXiv preprint arXiv:1309.7119*, pp. 1–13, 2013.

[25] J. Stanković, I. Marković, and M. Stojanović, "Investment Strategy Optimization Using Technical Analysis and Predictive Modeling in Emerging Markets," *Procedia Economics and Finance*, vol. 19, no. 15, pp. 51–62, 2015.

[26] G. Armano, M. Marchesi, and a. Murru, "A hybrid genetic-neural architecture for stock indexes forecasting," *Information Sciences*, vol. 170, no. 1, pp. 3–33, Feb. 2005.

[27] M. Kumar and M. Thenmozhi, "Forecasting stock index movement: A comparison of support vector machines and random forest," in *Proceedings of ninth Indian institute of capital markets conference*, 2006, pp. 1–16.

[28] S. Achelis, *Technical analysis from A to Z*. 2001.

[29] X. Cai, S. Hu, and X. Lin, "Feature extraction using Restricted Boltzmann Machine for stock price prediction," in *IEEE International conference on computer science and automation engineering*, 2012, pp. 80 – 83.

[30] L. Bucur and A. Florea, "Techniques for prediction in chaos – a comparative study on financial data," *U.P.B. Scientific bulletin, series C*, vol. 73, no. 3, pp. 17–32, 2011.

[31] D. Witkowska and E. Marcinkiewicz, "Construction and Evaluation of Trading Systems: Warsaw Index Futures," *International Advances in Economic Research*, vol. 11, no. 1, pp. 83–92, Mar. 2005.

[32] C. Chang and C. Lin, "LIBSVM: a library for support vector machines," *ACM Transactions on intelligent systems and technology*, vol. 2, no. 3, pp. 1–27, 2011.

[33] C. Hsu, C. Chang, and C. Lin, "A practical guide to support vector classification," *Department of Computer Science, National Taiwan University*, 2010.

[34] M. Hagenau, M. Hauser, M. Liebmann, and D. Neumann, "Reading All the News at the Same Time: Predicting Mid-term Stock Price Developments Based on News Momentum," in *Proceedings of the 46th Hawaii International Conference on System Sciences*, 2013, pp. 1279–1288.

[35] S. S. Groth and J. Muntermann, "An intraday market risk management approach based on textual analysis," *Decision Support Systems*, vol. 50, no. 4, pp. 680–691, Mar. 2011.

[36] E. Tsang, "Directional Changes, Definitions," *Working Paper WP050-10, Centre for Computational Finance and Economic Agents, University of Essex*, 2010.

[37] A. Duran and M. J. Bommarito, "A profitable trading and risk management strategy despite transaction costs," *Quantitative Finance*, vol. 11, no. 6, pp. 829–848, Jun. 2011.

[38] W. F. Sharpe, "Mutual Fund Performance," *The Journal of Business*, vol. 39, no. 1, Part 2: Supplement on Security Prices, pp. 119–138, 1966.

[39] N. Amenc and V. Le Sourd, *Portfolio Theory and Performance Analysis*. 2003.

[40] R. Luss and A. D'Aspremont, "Predicting abnormal returns from news using text classification," *Quantitative Finance*, vol. 15, no. 6, pp. 999–1012, 2015.