Knowledge-Based Systems 225 (2021) 107119

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Technical analysis strategy optimization using a machine learning approach in stock market indices

Jordan Ayala [a], Miguel García-Torres [b,*], José Luis Vázquez Noguera [a], Francisco Gómez-Vela [b], Federico Divina [b]

[a] Polytechnic School, Universidad Nacional de Asunción, P.O.Box: 2111 SL, Paraguay
[b] Division of Computer Science, Universidad Pablo de Olavide, ES-41013, Seville, Spain

ARTICLE INFO

Article history:
Received 7 September 2020
Received in revised form 29 April 2021
Accepted 30 April 2021
Available online 3 May 2021

Keywords:
Stock market prediction
Machine learning
Technical analysis

ABSTRACT

Within the area of stock market prediction, forecasting price values or movements is one of the most challenging issue. Because of this, the use of machine learning techniques in combination with technical analysis indicators is receiving more and more attention. In order to tackle this problem, in this paper we propose a hybrid approach to generate trading signals. To do so, our proposal consists of applying a technical indicator combined with a machine learning approach in order to produce a trading decision. The novelty of this approach lies in the simplicity and effectiveness of the hybrid rules as well as its possible extension to other technical indicators. In order to select the most suitable machine learning technique, we tested the performances of Linear Model (LM), Artificial Neural Network (ANN), Random Forests (RF) and Support Vector Regression (SVR). As technical strategies for trading, the Triple Exponential Moving Average (TEMA) and Moving Average Convergence/Divergence (MACD) were considered. We tested the resulting technique on daily trading data from three major indices: Ibex35 (IBEX), DAX and Dow Jones Industrial (DJI). Results achieved show that the addition of machine learning techniques to technical analysis strategies improves the trading signals and the competitiveness of the proposed trading rules.

1. Introduction

Predicting stock prices is a growing area of interest in both academic and financial economy fields. Despite the efforts made to develop new techniques, strategies and measures, none of them have proven to be particularly effective. Stock market prediction is a challenging problem since it is affected by different factors (many of which are unknown) and the market volatility that is difficult to capture in a model. Furthermore, this kind of data is very hard to predict since it presents non-linear relationships that are non-stationary with high heteroscedasticity [1–5].

Such difficulties have led to the efficient-market hypothesis [6] (EMH), which states that asset prices already take into account the information based both on past and future events. According to EMH, it is not possible to predict future prices based on historical data since for such purpose it is necessary to possess privileged information. Some critics to EMH point to the psychological biases that investors exhibit under uncertainty, leading to irrational and unpredictable behaviors [7]. Nowadays there is no consensus about EMH and the debate is still ongoing.

In a more recent work, the adaptive markets hypothesis [8] (AMH) has been proposed to overcome the behavioral critics made to EMH arguing that markets are not rational, but are rather driven by fear and greed. AMH tackles the stock market from a biological perspective within an evolutionary framework in which prices evolve according to competition, adaptation, and natural selection to financial interactions. According to AMH, predictable patterns may appear over time for short periods.

Traditionally, the two most widely used approaches to analyze stock market data are fundamental and technical analysis [9]. Fundamental analysis relies on the concept of *intrinsic value*, which means that the current price is based on quantitative and qualitative information. This approach adopts the EMH in the long-term. However, in the short term it assumes that there may be some inefficiencies. Technical analysis, on the other hand, is based on historical data to find patterns and predicts the future price movements of a stock. In contrast to fundamental analysis, this approach is mainly focused on the short term [10].

The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

* Corresponding author.
E-mail addresses: jmayalag@pol.edu.py (J. Ayala), mgarciat@upo.es (M. García-Torres), jlvazquez@pol.una.py (J.L.V. Noguera), fgomez@upo.es (F. Gómez-Vela), fdivina@upo.es (F. Divina).

https://doi.org/10.1016/j.knosys.2021.107119
0950-7051/

In addition to technical and fundamental analysis, many researchers formulate the problem of stock price prediction as a problem of time series forecasting. Within this approach, there are basically two categories of techniques: conventional and machine learning methods. Conventional strategies include, among others, statistical analysis, smoothing and regression-based techniques. For example, in [11] regression-based methods have been evaluated on the top 4 stock exchanges—New York, London, NASDAQ and Karachi stock exchange. They also evaluated them on the top 3 companies—Apple, Microsoft, and Google. The autoregressive integrated moving average (ARIMA) is used in [12] for short-term prediction of New York Stock Exchange (NYSE) and Nigeria Stock Exchange (NSE), while in [13], it is also used for short-term prediction of Amman Stock Exchange (ASE).

In recent years, there has been a growing interest in machine learning based techniques. The main reason is that in contrast to conventional methods, these techniques are more suitable to handle complex data with non-linear relationships. Within this field, artificial neural networks (ANNs) are a very popular approach and have been applied in numerous works. In [14], several ANN models are applied to forecast daily NASDAQ data. This method has also been applied to tick data from Indian stock index in [15]. An ANN with a different optimization function is proposed in [16] and is tested on daily data from seven stock indices. In more recent works, Deep Learning (DL) is gaining popularity. For example, [17] and [18] use a DL strategy to forecast daily data from Dow 30 companies and National Stock Exchange of India (NSEI) and the NYSE, respectively. Other popular techniques are Support Vector Regression [19], tree-based algorithms [20], etc.

In this work we propose a novel hybrid trading strategy that combines machine learning techniques with technical analysis indicators to generate Profitable trades. For such purpose, the trading rules are designed taking into account the asymmetric return distribution [21] to avoid false signals and achieve successful trading transactions. Results suggest that the integration of the information about the predicted trend (using machine learning) to the technical analysis leads to more robust signals.

The machine learning techniques analyzed in this paper are: Multivariate Linear Regression or Linear Model (LM), Artificial Neural Network (ANN), Random Forests (RF) and Support Vector Regression (SVR). As technical analysis strategies, the Triple Exponential Moving Average (TEMA) crossover, a strategy based on the Exponential Moving Average (EMA) indicator, and the Moving Average Convergence Divergence (MACD) are used. The proposed strategy was tested on three major indices – Ibex35 (IBEX), DAX and Dow Jones Industrial (DJI) – from 2011 to 2019. We can summarize the contributions of this work as follows:

- Analyze the performance of the predictive models induced with LM, ANN, RF and SVR.
- Study the return of the technical analysis-based strategies TEMA and MACD.
- Propose a hybrid trading strategy that combines machine learning and technical analysis.
- Develop a workflow to calculate the performance of the proposed strategy and compare it with technical analysis-based strategies.

The rest of the paper is organized as follows. Section 2 introduces the problem studied. Then, in Section 3, the machine learning methods, the technical indicators and strategies are presented. The data used in order to assess the effectiveness of our proposal is described in Section 4. The experimental results are presented in Section 5, and finally, we draw the main conclusions and discuss possible future developments in Section 6.

2. Stock market forecasting

Stock market values consist of a discrete sequence of time-ordered data points measured at equal time intervals. Given $\mathcal{E}$ a set of n samples characterized by T real values $x_1, \ldots, x_T$, $(1 \leq i \leq T)$ so that x_i represents the recorded value at time i, let w be the historical window and h the prediction horizon so that $w + h \leq T$. Then, the associated time series forecasting problem can be formulated as the problem of predicting the values of $x_{w+1}, \ldots, x_{w+h}$, given $x_1, \ldots, x_w (w + h \leq T)$, with the objective of minimizing the error between the predicted value $\hat{x}_{w+i}$ and the actual value $x_{w+i} (1 \leq i \leq h)$. A more extensive introduction to time series analysis can be found in [22].

2.1. Related work

As previously mentioned, the application of machine learning techniques to the stock market forecasting problem has gained popularity in recent years. This is due, among others, for the suitability of such techniques to handle complex relations and the advances in computer technology that allow to process massive amounts of data.

In an early work, Yao et al. [23] investigate the performance of the autoregressive integrated moving average (ARIMA) and the ANNs techniques for forecasting the Kuala Lumpur Composite Index (KLCI). The indicators moving average (MA), momentum (M), Relative Strength Index (RSI), and stochastics %K and moving average of stochastics %D (KD) are used as the inputs of the ANN. The trading rules, which are based on such predictions, are tested on daily data collected from January 3, 1984 to October 16, 1991. In another work, Pérez-Rodríguez et al. [24] propose a combination of the filter techniques [25] trading strategy with smooth transition autoregression (STAR) models and ANNs. The study is conducted on daily data from the Spanish IBEX stock index return gathered over the period going from December 30, 1989 to February 10, 2000. In [26] Chang et al. the authors propose a combination of ANN with Piecewise Linear Representation (PLR) model to make trading decisions. This strategy receives as input, a set of technical indicators — MA, Bias (BIAS), RSI, KD, MACD, Williams %R (WR), Transaction Volume (TV) and Differences of technical indexes (Δ), which are processed to produce trading signals. This proposal is tested on nine different stocks – AU Optronics (AUO), Epistar Corporation (EPISTAR), GP, Silicon Integrated System Corporation (SiS), SENAO International Corporation (SENAO), D-Link Corporation (D-LINK), Foxlink Corporation (FOXLINK), Compal Corporation (COMPAL) and UMC Corporation (UMC) – collected from January 2, 2004 to April 12, 2006.

Teixeira and Oliveira [27] combine technical analysis and k nearest neighbor (kNN) classifier for automatic trading. The indicators MA, RSI, KD, and Bolinger Bands (BB) are calculated and used as input of the kNN model. Authors compare the results of the proposed trading rule with a buy-and-hold strategy on fifteen stocks from São Paulo Stock Exchange. The data collected covered a period going from April 1, 1990 to March 9, 2009. In another work R. Dash and K. Dash [28] propose a decision support system that integrates technical analysis and a computational efficient functional link ANNs (CEFLANNs). First, the proposed workflow learns the trends from data computed from the technical indicators MA, MACD, KD, RSI and WR. This information is then applied to the trading rules. The same workflow is compared using the machine learning techniques Naive Bayes, Support Vector Machine (SVM), kNN and Decision Tree (DT). In this work five years of historical stock index price values from BSE SENSEX and S&P500 is used. The data was collected in the period comprised between January 2010 and December 2014.

Sang and Di Pierro [29] propose the use of an ANN to improve technical analysis trading strategies. For each strategy considered

Table 1
Summary of the related works described. In the Tasks column, F refers to Forecasting, C to classification and T to trading.

Year	Indicators	Methods	Tasks	Stocks	Period	Ref.
1999	MA, M, RSI, KD	ARIMA, ANN	F, T	KLCI	03/01/1984–16/10/1991	[23]
2005	–	STAR, ANN	F, T	IBEX35	30/12/1989–10/02/2000	[24]
2009	MA, BIAS, RSI, KD, MACD, WR, TV, Δ	PLR, ANN	F, T	AUO, EPISTAR, GP, SiS, SENAO, D-LINK, FOXLINK, COMPAL, UMC	02/01/2004–12/04/2006	[26]
2010	MA, RSI, KD, BB	kNN	C, T	BM&FBovespa	01/04/1998–09/03/2009	[27]
2016	MA, MACD, KD, RSI, WR	ANN, NB, SVM, kNN, DT	C, T	BSE SENSEX, S&P500	01/01/2010–01/12/2014	[28]
2019	SMA, RSI, MACD	ANN	F, T	S&P500	01/01/2014–31/12/2015	[29]
	RSI, MA, KD, WD, EMA, MACD	DL, SVM, LR	C, T	NSEI	16/11/2016–15/11/2018	[30]
2020	MACD	GA	F, T	NASDAQ	01/01/2013–31/12/2019	[31]
	SMA, EMA, RSI, CMO, CCI, MACD, PPO, TMA, KD, CAD, BB, WR	DL, SVR, LSTM, CNN, MLP, MLP+BiLSTM	F	S&P500, NASDAQ, Russell2000, DJ	2/11/2008–12/07/2019	[32]

— Simple MA (SMA), RSI and MACD, an ANN model is induced in order to determine whether the strategy will produce a Profit or a loss. The data used consisted of nine indexes representing entire sectors under S&P500 recorded in the period 2014–2015. Deep Learning has also been combined with stock technical indicators in [30] by Agrawal et al. First, new input features are created from the RSI, MA, KD, WR, Exponential MA (EMA) and MACD indicators. Then, the price trend is estimated using DL and such prediction is incorporated in the trading rule. The performance of DL is compared with SVM and Linear Regression (LR) on data from 3 banks listed in the National Stock Exchange of India. The data encompassed the trading days of 2 years: from November 16, 2016 to November 15, 2018.

In a recent work Aguirre et al. [31] propose to hybridize MACD with a Genetic Algorithm (GA) to optimize the parameters that generate the buy–sell signals. The proposal is compared with MACD and Buy & Hold strategies on data from NASDAQ stock index over a seven-year period: from January 1, 2013 to December 31, 2019. The article from Chen et al. [32] introduce a novel hybrid DL model that integrates Attention Mechanism (AM), Multi-Layer Perceptron (MLP), and Bidirectional Long–Short Term Memory (BiLSTM) ANN. First, the strategy creates a knowledge base comprised of 31 features obtained from historical prices of stocks, technical indicators and natural resources prices and historical data of the Google index. Then, the dimensionality is reduced by applying Principal Components Analysis (PCA). After this phase, the hybrid DL approach is applied to forecast the closing price. The technical indicators used in this work are MA, EMA, RSI, Chande Momentum Oscillator (CMO), Commodity Channel Index (CCI), MACD, Percentage Price Oscillator (PPO), Triangular Moving Average (TMA), KD, Chaiking A/D Oscillator (CAD), BB and WR. The model is compared with Support Vector Regression (SVR), LSTM, Convolutional Neural Network (CNN), MLP and MLP+BiLSTM. The robustness of the proposed model was proven through testing on the stock indexes S&P 500, Dow Jones (DJ), NASDAQ, and Russell 2000 (Russell2000).

New hybrid models that integrate sentiment analysis data are also becoming popular in recent works (see [33–35]). Despite the potential advantage of incorporating the market sentiment, it also presents challenging difficulties such as misspelling, shortcuts and information duplication in text data. Furthermore that may led to low efficiency [36]. Ensemble approaches have also obtained good results when applied to the stock market prediction problem. Examples of such strategies are, for instance, [37,38,38–41].

We refer the reader to [42] for a further description of additional works using machine learning strategies. A summary of the hybrid models described in this section is shown in Table 1. The first column indicates the year of the publication followed by the technical indicators considered in the work. Then, the strategy used are listed followed by the tasks addressed: F for forecasting, C for classification and T for trading signals. Next, the stock data source and the period of time collected are shown, and, finally, the relative reference is reported.

2.2. Methods

In this section, we will briefly describe the methods used to produce the predictions of the experiments performed in this work. As mentioned before, in this paper we use four learning strategies that are independently trained, named Linear Model, Artificial Neural Networks, Random Forest and Support Vector Regression. These methods were selected since they have proven their good performance in the field. Due to this, they are widely-used in literature for regression tasks in stock market prediction, e.g., [43–46].

Linear Model (LM) [47], also called linear regression, is a statistical approach that is typically used to model the relationship between two variables and also for time series forecasting. The main idea behind this approach is to find the relationship between two variables using a linear equation, $Y = a + bX$, for representing the association between the independent variable (X) and the dependent one (Y), i.e., the variable to be predicted. This approach can also use multiple independent variables to determine the final value of the dependent one, which is called multiple linear regression and it is represented by the equation $Y = a + b_1X_1 + \cdots + b_nX_n + \epsilon$, where ϵ is the residual (difference between the predicted and the observed value) and X_i, $1 \leq i \leq n$ are the n explanatory variables. In this work, we model different dependent variables from the same input dataset so, we selected the multi-output regression. Multi-output regression, also known in the literature as multi-target, aims to simultaneously predict multiple output/target variables.

In this paper, we have used the implementation provided by R caret package [48].

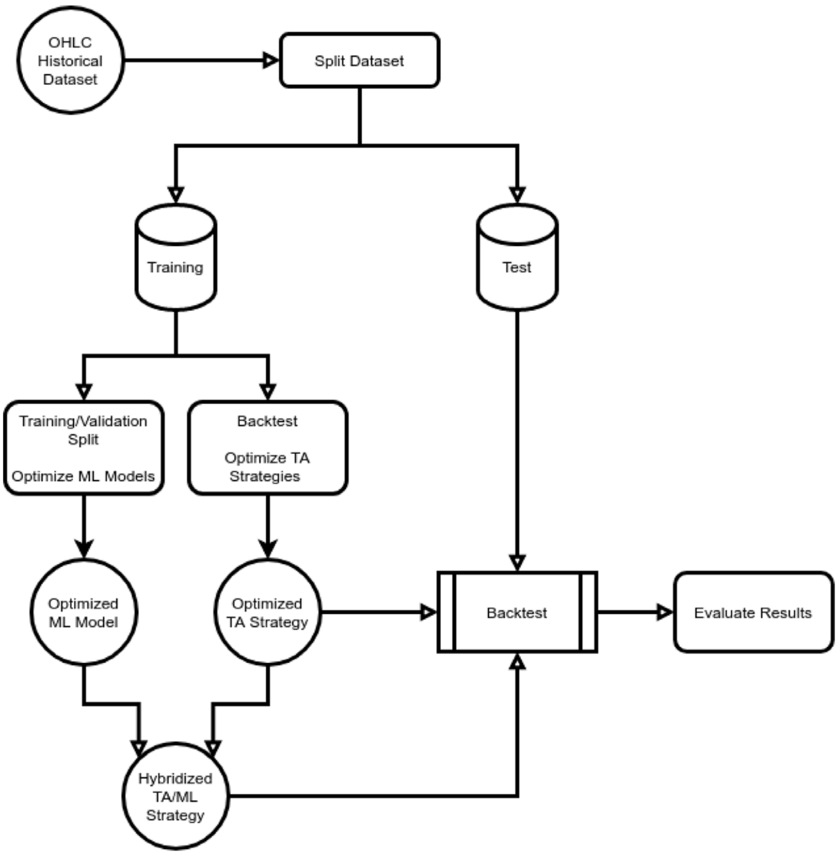


Fig. 1. General scheme for backtesting the proposed trading strategy.

Table 2
Summary of the indices data.

Symbol	Start	End	#observations
IBEX	2011-01-02	2019-12-31	2800
DAX	2011-01-02	2019-12-31	2799
DJI	2011-01-03	2019-12-31	2713

Table 3
Optimal parameters for each learning scheme.

A	Parameter			Description
LM	–			–
ANN	#It.	=	500	Maximum number of iterations
	Size	=	1	Number of units in the hidden layer
	Decay	=	0	The weight decay
RF	#Trees	=	100	Number of trees to grow
	#Nodes	=	100	Maximum number of terminal nodes
SVR	kernel	=	linear	Kernel type
	ε	=	0.1	Insensitive-loss function
	tolerance	=	0.001	Tolerance for stopping criterion

Artificial Neural Networks (ANN) [49], are computational models for classification and regression. They are inspired by the human brain neural networks. An ANN is formed by a set of connected nodes, called (*neurons*), that are interconnected with each other simulating the connections of the brain. The information is transmitted from neuron to neuron and the learning is achieved through training data. ANNs are composed by different layers: an input layer, one or more hidden layers, and an output layer. Connections among nodes of different layers are weighted. Thus, the information is carried from one layer to another using an *activation function* that accounts for the non-linearity in the data. This process, which is called feed-forward propagation, defines how the data are fed into the next layers. The model learns by minimizing a *loss* function that calculates the data prediction error by adjusting the weight of the connections. The final prediction is computed in the output layer, where a transformation function is used in the final step. The number of neurons in the input layer varies and depends on the dimensionality of the input data, while in the output layer this number depends on the expected output. In the case of regression tasks, a single neuron that offers a final numerical value is used.

In this work we use a feed-forward ANN that consists of an input layer, one hidden layer, and an output layer. No feedback or lateral connections were used. The algorithm developed in the R package `nnet` [50] was used.

Random Forests (RF) is an ensemble method that may be applied for both classification and regression tasks. The prediction model induced by RF consists of a set of decision trees, and the final prediction is computed considering the predictions of each tree, e.g., with a majority vote in the case of classification. For regression tasks, the final prediction consists of the average of the trees' predictions. The trees induced are independently trained with a bootstrap sample of the training data (*Bagging* ensemble method) selecting a random number of features. RF was first proposed by Ho [51] and improved by Breiman [52], combining the random sub-set method with its bagging method. RFs offer a way of averaging multiple trees with a low variance for predictions, since each tree is formed with a random subset of data and features. In this paper we used the implementation provided by the R package [53], which is based on the Breiman's algorithm.1ausugum

Support Vector Regression (SVR) [54], is a variant of support vector machines (SVM), adapted for being used for regression and forecasting tasks. SVR applies the same criteria as SVM for classification. SVR is characterized by the use of kernels, sparse

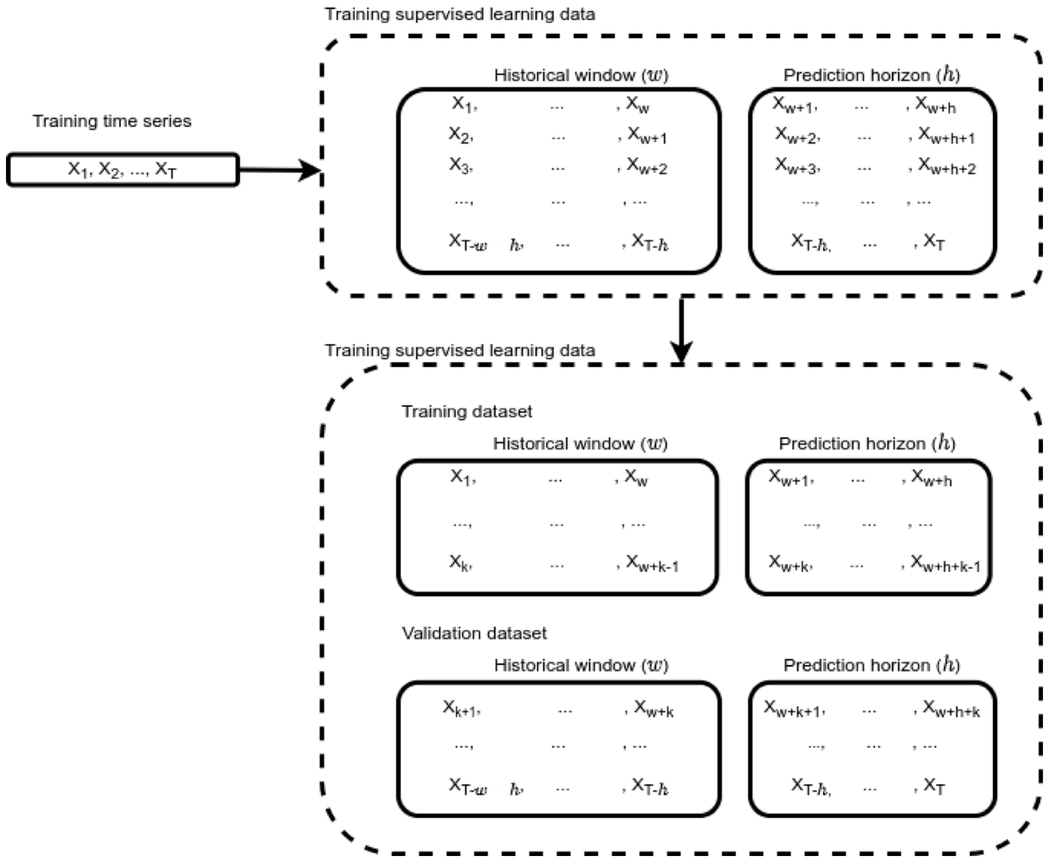


Fig. 2. First, the training time series is converted to a supervised learning problem. Next, the data is split into training and validations datasets. w refers to the amount of the historical data used, while h determines the prediction horizon.

solution, and VC control of the margin and the number of support vectors. In spite of their similarities, SVR presents some minor differences with SVM.

Since the output is a real number, it is difficult to predict an exact value, since there are infinite possibilities. However, the main idea is the same: minimize the error, finding the hyper plane that maximizes the margin by taking into account that part of the error is tolerated.

In this work, the implementation provided by the R caret package was used [48].

2.3. Performance metrics

In this section, we introduce the metrics used to assess the quality of the learning methods used in this paper. In particular, we use four measures that are commonly used in regression: the mean absolute error (MAE), the root mean squared error (RMSE), the mean absolute percentage error (MAPE) and the symmetric mean absolute percentage error (sMAPE).

Given the sample size n, the actual observation y_t at time t, $1 \leq t \leq T$, and the prediction $\overline{y}_t$, the metrics are defined as follows:

- **Mean absolute error (MAE)** measures the average over a sample of the absolute differences between the predicted and actual observation. It is defined by the following formula:

$$MAE = \frac{1}{n}\sum_{t=1}^{n}|y_t - \overline{y_t}|$$

  The MAE is a linear score. This means that all the single differences are considered equally in the average. It follows that larger errors will contribute linearly to the total error. This represents the main drawback of this measure, as outliers may affect its meaning. An advantage of using MAE is that it is an intuitive and easy to interpret measure.

- **Root mean squared error (RMSE)** also measures the average magnitude of the errors, and it is defined as:

$$\sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \overline{y_t})^2}$$

  Notices that this measure is a quadratic scoring rule. It follows that it measures the average magnitude of the error. In this sense outliers have a greater impact on this measure.

- **Mean absolute percentage error (MAPE)** measures how accurate, as a percentage, a forecasting system is. It is calculated as follows:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{y_t - \hat{y}_t}{y_t}\right|$$

  We can state that MAPE is basically the percentage equivalent of MAE. MAE is the average magnitude of error produced by a model, while MAPE represent how far, on average, the predictions are from the real values. As for MAE, MAPE is not affected too much by the presence of outliers, as it is a linear function. MAPE have a problem that is the denominator is zero, then its value would be undefined.

- **Symmetric mean absolute percentage error** [55] **(sMAPE)** is a modified version of MAPE to fix the issues of being infinite or undefined due to zeros in the denominator [56],

Table 4
Performance of each learning algorithm averaged over the historical window w and the prediction horizon h. For each metric and algorithm, the average results together with its standard deviation is reported.

w	Metric	LM	ANN	RF	SVR
IBEX	MAE	175.76 ± 2.88	180.40 ± 2.59	204.08 ± 9.84	177.35 ± 5.07
	RMSE	226.20 ± 2.64	231.60 ± 2.79	259.08 ± 12.24	228.33 ± 4.61
	sMAPE[a]	18.45 ± 0.35	18.92 ± 0.32	21.50 ± 1.06	18.58 ± 0.58
	MAPE[a]	18.41 ± 0.34	18.86 ± 0.30	21.43 ± 1.00	18.51 ± 0.56
DAX	MAE	257.97 ± 3.70	260.00 ± 4.92	625.47 ± 144.60	259.11 ± 4.98
	RMSE	326.63 ± 6.05	331.99 ± 7.81	749.97 ± 148.81	329.75 ± 8.09
	sMAPE[a]	21.15 ± 0.31	21.28 ± 0.40	51.30 ± 12.13	21.22 ± 0.41
	MAPE[a]	21.10 ± 0.29	21.15 ± 0.37	53.19 ± 12.95	21.11 ± 0.37
DJI	MAE	487.69 ± 10.09	476.52 ± 9.65	4614.25 ± 63.61	493.45 ± 12.14
	RMSE	638.27 ± 11.25	635.91 ± 11.70	4900.81 ± 93.46	643.72 ± 14.44
	sMAPE[a]	19.46 ± 0.39	18.99 ± 0.36	200.62 ± 3.74	19.69 ± 0.46
	MAPE[a]	19.44 ± 0.40	18.92 ± 0.35	226.00 ± 5.03	19.66 ± 0.46

[a] $\times 10^{-3}$.

and is defined in the following way:

$$sMAPE = \frac{1}{n}\sum_{t=1}^{n}\frac{2\left|y_t - \bar{y}_t\right|}{\left|y_t + \bar{y}_t\right|}$$

As already stated, sMAPE is a modified MAPE in which the divisor is half of the sum of the actual and predicted values.

In following we report the values of MAPE and sMAPE multiplied by 10^3, in order to avoid presenting extremely small values.

Table 5
Parameter ranges considered for TEMA and MACD trading strategies to find the optimal combination.

A	Parameter	Description
TEMA	Fast	[1, 25]
	Medium	[5, 50]
	Slow	[10, 75]
MACD	Fast	[1, 25]
	Slow	[5, 75]
	Signal	[5, 25]

3. Stock market strategies

In this section, we first introduce the technical analysis indicators used in this paper, namely the Exponential Moving Average, the Moving Average Convergences/Divergences and the Triple Exponential Moving Average crossover strategy. We also briefly describe the trading strategies associated to them. After that, we continue by describing the strategy proposed in this paper.

3.1. Technical indicators

In our proposal, we use two commonly used indicators in trading strategy, i.e., the Exponential Moving Average and the Moving Average Convergence/Divergence indicators.

Exponential Moving Average (EMA). This is one of the most popular technical indicators. EMA is used to gauge the trend of a financial asset. To this aim, EMA smooths the price by filtering out the noise from random price fluctuations by averaging the price over a given period of time m. EMA is based on past prices and, so, it is a lagging indicator. This means that EMA cannot predict new trends, but it can confirms the direction of the trend.

EMA assigns more weight to recent prices, and is defined as in the following formula:

$$EMA_t^m(S_t) = \begin{cases} S_1 & \text{if } t = 1, \\ \alpha \cdot S_t + (1-\alpha)\cdot EMA_{t-1}^m & \text{if } t > 1. \end{cases}$$

In the above formula, S_t refers to the current price, m to the number of observations and α is a smoothing factor, $0 \le \alpha \le 1$, that is calculated as $\alpha = \frac{2}{m+1}$.

Moving Average Convergence/Divergence (MACD). This indicator was first proposed by Gerald Appel in [57] and used to identify the trend direction and duration by calculating the relationship between two EMAs. MACD consists of two series: the MACD line ($MACD_t$) and the MACD signal series ($Signal_t$). The MACD line is obtained as the difference between the faster and the slower EMAs. The signal is the Moving Average (MA) of the MACD series. Given the time periods m, n and p so that $m < n$, then

$$MACD_t = EMA_t^m(S_t) - EMA_t^n(S_t),$$
$$Signal_t = EMA_t^p(MACD_t).$$

3.2. Technical analysis strategies

As trading strategies, two simple strategies based on MACD and EMA indicators are selected. The description of each strategy is given below.

MACD strategy. A very common strategy based on the MACD indicator is the following:

$$Strategy_{t+1} = \begin{cases} Buy & \text{if } MACD_t > Signal_t, \\ Sell & \text{if } MACD_t < Signal_t. \end{cases}$$

Therefore, a potential buy signal is generated when $MACD_t$ crosses above the $Signal_t$ line and, similarly, we have a potential sell signal in the opposite scenario.

Triple Exponential Moving Average (TEMA) crossover strategy. This strategy is used to identify trends in the market and to deal with false market signals. It is based on three EMAs, for short, mid and long term periods, respectively. The EMA for short period (fast EMA) is the first one to detect a possible shift in the trend that is confirmed once it crosses both the medium and the slow EMAs. So, giving three periods of time m, n and p so that $m < n < p$, the strategy is defined as follows:

$$Strategy_{t+1} = \begin{cases} Buy & \text{if } EMA_t^m > EMA_t^n \text{ and } EMA_t^m > EMA_t^p, \\ Sell & \text{if } EMA_t^m < EMA_t^n \text{ and } EMA_t^m < EMA_t^p. \end{cases}$$

Therefore, the buy signal is generated once the fast MA crosses above the medium and slow MAs, while the sell signal is produced in the other case.

Table 6
The performance of the top 5 combination of parameters for TEMA and MACD.

Strategy	Index					
	IBEX		DAX		DJI	
	Parameters	PF	Parameters	PF	Parameters	PF
TEMA	(2, 6, 10)	0.943	(14, 30, 74)	1.364	(18, 34, 54)	3.900
	(2, 5, 10)	0.940	(14, 31, 74)	1.364	(18, 35, 54)	3.900
	(2, 7, 10)	0.918	(3, 25, 35)	1.341	(18, 36, 54)	3.900
	(2, 9, 10)	0.916	(19, 26, 64)	1.330	(21, 22, 50)	3.845
	(1, 8, 12)	0.913	(19, 27, 64)	1.330	(21, 23, 50)	3.845
MACD	(9, 19, 6)	1.056	(3, 11, 6)	1.629	(3, 28, 9)	2.003
	(5, 12, 17)	1.047	(5, 8, 5)	1.629	(2, 34, 11)	1.998
	(6, 19, 9)	1.043	(3, 7, 11)	1.621	(2, 25, 13)	1.988
	(5, 13, 16)	1.038	(3, 6, 11)	1.619	(2, 27, 11)	1.987
	(2, 30, 13)	1.037	(4, 5, 11)	1.619	(2, 29, 12)	1.985

Table 7
Comparison of the performance of the trading strategies with and without hybridization using the optimal parameter values found for TEMA and MACD indicators.

Index	Strategy	Parameters	#T	PF	NT	$\overline{T}$	D_{max}	PP
IBEX	TEMA	(2, 6, 10)	19	0.705	−403.6	−21.2	−1052.5	42.11
	hTEMA		2	∞	866.1	433.1	−1033.1	100.00
	MACD	(9, 19, 6)	19	0.518	−807.6	−42.5	−1627.0	31.58
	hMACD		2	∞	300.5	150.3	−1228.0	100.00
DAX	TEMA	(14, 30, 74)	2	∞	1350.0	675.0	−1072.0	100.00
	hTEMA		1	∞	2211.8	2211.8	−1260.6	100.00
	MACD	(3, 11, 16)	33	1.117	303.9	9.2	−1083.5	36.36
	hMACD		3	32.991	1817.1	605.7	−1260.6	66.67
DJI	TEMA	(18, 34, 54)	4	1.478	470.5	117.6	−2402.3	25.00
	hTEMA		1	∞	2678.0	2678.0	−2151.5	100.00
	MACD	(3,28, 9)	20	1.307	1046.1	52.3	−2108.2	40.00
	hMACD		2	∞	2738.9	1369.5	−2151.5	100.00

3.3. Hybridization of machine learning with technical analysis

The proposed strategy aim at improving the technical analysis trading signals by incorporating the machine learning techniques in the trading rules. The general scheme of the proposal is presented in Fig. 1. First, stock data is collected from the server. The data is then split into training and test sets. The training data is used to optimize the predictive models and the technical analysis based strategies. The so obtained hybrid schema is then built and its performance is assessed on the test dataset. Finally, the results are analyzed to establish the quality of the hybrid strategy. The main steps of this workflow are:

(a) Build the optimal learning model.
(b) Optimize the technical analysis strategies.
(c) Backtest the hybrid technical analysis strategy

In the following we describe each step.

(a) Build the optimal learning model. The time series is preprocessed as in Divina et al. [58] using a strategy often referred to as Walk Forward Validation. In a nutshell, this strategy requires to set the number of historical observations w and the size of the prediction horizon h. Given the training time series observations and the values of w and h, the training supervised learning data is created by considering a sliding window of size $w + h$. The first instance consists of the first $w + h$ observations. For the next instance, such a window slides forward one value from x_2 to x_{w+h+1} and so on. Then, the training data is divided into training and validation datasets. This process is graphically described in Fig. 2.

Before building the final predictive model, a hyperparameter tuning phase is carried out. For each combination of hyperparameters, the model is built on the training dataset and tested on the validation dataset. As a final step, the model is induced on the full training data using the optimal hyperparameters found.

(b) Optimize the technical analysis strategies. In this step, the training data is used for backtesting the technical analysis strategies. For this purpose, a grid search is applied on the entire training data. The parameters with the best performance will be selected for the hybridization.

(c) Backtest the hybrid technical analysis strategy. In this last step, the hybrid strategy is backtested on the test dataset. The proposed trading rules is applied and the performance evaluated.

3.4. Proposed trading rule

MACD and EMA, which belong to the family of lagged indicators, are measurable factors that trail behind the current market price. Although they are used to create trading signals, actually they are useful to confirm the strength of a long-term trend since they look retrospectively at past data [59]. Furthermore, this kind of indicators may generate false signals due to many factors such as short-term market fluctuations. Therefore, the integration of predicted price may lead to a decrease in the number of false signals.

The proposed rule uses the predictive model to generate trading signals. For a given day t, Buy_t and $Sell_t$ refer to the buy and sell signals given only by the technical analysis strategies. The learning model provides information about the evolution of the price. In order to decide about to enter in the market, it is useful to know the trend on the stock values. For this purpose, although larger values of h allow us to estimate the price farther in the future, the prediction error increases with h. So, in order to balance both issues, a prediction horizon of just over a month was set in this study.

Table A.8
Results achieved by all the methods on IBEX averaged over the prediction horizon h for each historical window w.

w	Metric	LM	NN	RF	SVR
6	MAE	174.36 ± 49.57	179.08 ± 52.89	199.44 ± 51.36	174.05 ± 49.35
	RMSE	225.64 ± 63.71	230.46 ± 66.82	254.26 ± 63.98	226.11 ± 63.78
	sMAPE[a]	18.19 ± 5.19	18.69 ± 5.54	20.93 ± 5.43	18.15 ± 5.16
	MAPE[a]	18.17 ± 5.18	18.65 ± 5.52	20.89 ± 5.41	18.09 ± 5.13
12	MAE	174.10 ± 49.65	178.44 ± 52.81	196.79 ± 50.40	173.74 ± 49.28
	RMSE	225.24 ± 63.75	229.59 ± 66.62	249.40 ± 62.78	225.69 ± 63.57
	sMAPE[a]	18.17 ± 5.20	18.63 ± 5.53	20.64 ± 5.32	18.12 ± 5.15
	MAPE[a]	18.15 ± 5.19	18.60 ± 5.52	20.63 ± 5.32	18.06 ± 5.12
24	MAE	174.44 ± 49.74	178.52 ± 52.77	196.00 ± 49.68	173.48 ± 49.11
	RMSE	225.69 ± 63.65	229.64 ± 66.36	250.25 ± 62.06	225.17 ± 63.10
	sMAPE[a]	18.23 ± 5.22	18.65 ± 5.53	20.59 ± 5.25	18.10 ± 5.14
	MAPE[a]	18.21 ± 5.21	18.62 ± 5.52	20.58 ± 5.24	18.05 ± 5.11
48	MAE	173.95 ± 47.63	178.41 ± 50.85	200.48 ± 50.06	172.74 ± 47.14
	RMSE	224.69 ± 60.98	229.42 ± 64.22	254.41 ± 61.54	223.94 ± 60.68
	sMAPE[a]	18.21 ± 5.02	18.66 ± 5.35	21.10 ± 5.32	18.06 ± 4.95
	MAPE[a]	18.17 ± 5.00	18.61 ± 5.32	21.09 ± 5.30	17.98 ± 4.91
72	MAE	172.47 ± 47.21	177.78 ± 51.10	198.23 ± 48.34	172.00 ± 47.15
	RMSE	222.42 ± 60.56	227.95 ± 64.30	251.60 ± 59.91	222.57 ± 60.53
	sMAPE[a]	18.11 ± 5.00	18.65 ± 5.39	20.90 ± 5.15	18.03 ± 4.97
	MAPE[a]	18.07 ± 4.98	18.59 ± 5.37	20.86 ± 5.13	17.96 ± 4.94
96	MAE	172.91 ± 47.49	178.51 ± 51.35	194.93 ± 49.09	174.23 ± 48.28
	RMSE	223.38 ± 61.34	229.76 ± 65.41	247.73 ± 60.50	225.09 ± 62.20
	sMAPE[a]	18.18 ± 5.02	18.74 ± 5.41	20.57 ± 5.22	18.28 ± 5.08
	MAPE[a]	18.14 ± 5.01	18.69 ± 5.39	20.53 ± 5.19	18.21 ± 5.04
120	MAE	175.19 ± 49.46	179.48 ± 52.61	197.99 ± 51.07	176.78 ± 50.02
	RMSE	225.41 ± 62.66	230.98 ± 66.44	251.17 ± 62.63	227.48 ± 63.25
	sMAPE[a]	18.43 ± 5.23	18.86 ± 5.54	20.88 ± 5.40	18.56 ± 5.26
	MAPE[a]	18.40 ± 5.22	18.80 ± 5.52	20.84 ± 5.37	18.48 ± 5.22
168	MAE	180.96 ± 52.97	184.31 ± 55.41	210.29 ± 53.42	184.55 ± 54.35
	RMSE	231.11 ± 65.38	235.62 ± 68.39	265.79 ± 64.37	234.60 ± 66.61
	sMAPE[a]	19.06 ± 5.60	19.39 ± 5.85	22.21 ± 5.62	19.38 ± 5.71
	MAPE[a]	19.02 ± 5.59	19.33 ± 5.81	22.12 ± 5.57	19.29 ± 5.65
192	MAE	180.36 ± 51.95	184.36 ± 54.85	213.81 ± 57.38	184.78 ± 54.36
	RMSE	230.32 ± 64.65	235.52 ± 68.08	270.95 ± 68.08	234.71 ± 66.68
	sMAPE[a]	18.99 ± 5.49	19.38 ± 5.78	22.55 ± 6.02	19.42 ± 5.71
	MAPE[a]	18.92 ± 5.46	19.29 ± 5.73	22.40 ± 5.93	19.31 ± 5.66
216	MAE	176.65 ± 50.26	182.05 ± 54.28	210.87 ± 55.26	181.24 ± 53.10
	RMSE	226.94 ± 62.45	234.04 ± 67.30	267.32 ± 66.30	232.49 ± 65.97
	sMAPE[a]	18.58 ± 5.30	19.13 ± 5.71	22.24 ± 5.79	19.03 ± 5.58
	MAPE[a]	18.51 ± 5.27	19.03 ± 5.66	22.08 ± 5.70	18.93 ± 5.52
240	MAE	177.99 ± 52.21	183.45 ± 56.34	226.01 ± 56.01	183.31 ± 55.06
	RMSE	227.35 ± 64.05	234.64 ± 69.01	287.01 ± 67.86	233.75 ± 67.75
	sMAPE[a]	18.76 ± 5.53	19.32 ± 5.95	23.85 ± 5.87	19.29 ± 5.80
	MAPE[a]	18.71 ± 5.51	19.23 ± 5.91	23.65 ± 5.76	19.21 ± 5.75

[a] $\times 10^{-3}$.

In this context, to estimate the trend of the market, the strategy compares the price x_t at day t with the predicted value $\hat{x}_{t+h}$ at day $t+h$. The tendency is considered as an uptrend (UT_{t+h}) if $\hat{x}_{t+h} > x_t$ and a downtrend (DT_{t+h}) if $\hat{x}_{t+h} < x_t$.

It is a well-known fact that the very largest movements in the market usually correspond to decrements rather than to increments [21]. This fact may favor the number of false sell signals. Therefore in our proposal the buy signal at day $t+1$ is generated if it is confirmed by the technical indicator at day t or by the estimated trend. On the other hand, the sell signal is generated if the technical indicator and the trend identify a down market movement. So, let D_{t+1} be the trading decision at day $t+1$, the trading rule is defined as follows:

$$D_{t+1} = \begin{cases} \text{Buy if} & Buy_t \text{ or } UT_{t+h} \\ \text{Sell if} & Sell_t \text{ and } DT_{t+h} \end{cases}$$

The use of a predictive model improves the estimation of the price trend in a near future. So, despite the fact that its inclusion increases the complexity of the strategy, it provides valuable and relevant information that can yield better trading signals.

3.5. Metrics

The metrics used to evaluate the performance of the trading strategies associated are:

- **Profit factor (PF)**. Relation between Profits and Losses:

$$\mathrm{PF} = \left| \frac{\mathrm{Profits}}{\mathrm{Losses}} \right|$$

  It follows that if PF < 1, then Losses > Profits.
- **Number of trades (#T)**. The number of trades done. A trade is composed of a buy and a sell order.
- **Maximum drawdown ($\mathbf{D}_{max}$)**. The Worst decline-from-peak observed in the test period.
- **Net Profit (NP)**. Sum of Profits and Losses:

$$\mathrm{NP} = \mathrm{Profits} - \mathrm{Losses}$$

- **Average Profit per trade** ($\overline{T}$). This metric is calculated by dividing the net Profit by the number of trades:

$$\overline{T} = \frac{NP}{\#T}$$

Table A.9
Results achieved by all the methods on DAX averaged over the prediction horizon h for each historical window w.

w	Metric	LM	NN	RF	SVR
6	MAE	253.58 ± 82.98	253.82 ± 83.91	788.23 ± 36.07	252.58 ± 82.40
	RMSE	318.27 ± 98.99	321.03 ± 100.76	917.72 ± 31.98	318.43 ± 98.56
	sMAPE[a]	20.77 ± 6.78	20.77 ± 6.84	65.02 ± 3.07	20.68 ± 6.73
	MAPE[a]	20.77 ± 6.79	20.68 ± 6.80	67.90 ± 3.25	20.63 ± 6.72
12	MAE	253.56 ± 83.39	253.76 ± 84.23	784.00 ± 36.86	252.70 ± 82.71
	RMSE	318.40 ± 99.73	320.98 ± 101.40	913.47 ± 32.54	318.59 ± 99.18
	sMAPE[a]	20.77 ± 6.82	20.76 ± 6.87	64.66 ± 3.14	20.69 ± 6.75
	MAPE[a]	20.76 ± 6.83	20.68 ± 6.83	67.50 ± 3.32	20.64 ± 6.74
24	MAE	255.37 ± 83.51	260.38 ± 86.87	774.25 ± 35.18	253.77 ± 82.53
	RMSE	320.70 ± 100.31	331.16 ± 104.65	903.04 ± 30.91	319.77 ± 99.44
	sMAPE[a]	20.94 ± 6.83	21.27 ± 7.06	63.82 ± 3.00	20.79 ± 6.74
	MAPE[a]	20.94 ± 6.84	21.11 ± 6.98	66.59 ± 3.17	20.75 ± 6.73
48	MAE	257.37 ± 83.30	258.24 ± 84.79	739.87 ± 25.10	257.41 ± 83.48
	RMSE	325.21 ± 101.56	328.68 ± 104.12	866.68 ± 20.63	326.82 ± 102.16
	sMAPE[a]	21.10 ± 6.82	21.15 ± 6.92	60.88 ± 2.14	21.08 ± 6.82
	MAPE[a]	21.06 ± 6.80	21.04 ± 6.87	63.40 ± 2.24	20.99 ± 6.77
72	MAE	256.26 ± 81.82	257.37 ± 83.24	693.30 ± 19.57	257.38 ± 82.27
	RMSE	325.34 ± 100.65	329.07 ± 103.09	818.95 ± 15.05	329.07 ± 101.84
	sMAPE[a]	21.01 ± 6.69	21.08 ± 6.80	56.93 ± 1.68	21.08 ± 6.72
	MAPE[a]	20.95 ± 6.66	20.96 ± 6.73	59.15 ± 1.73	20.96 ± 6.65
96	MAE	254.73 ± 82.71	256.05 ± 83.87	660.71 ± 19.36	257.72 ± 83.00
	RMSE	324.30 ± 101.15	327.52 ± 103.14	785.09 ± 14.64	330.12 ± 102.27
	sMAPE[a]	20.89 ± 6.76	20.98 ± 6.85	54.18 ± 1.66	21.11 ± 6.77
	MAPE[a]	20.81 ± 6.73	20.86 ± 6.78	56.19 ± 1.70	20.96 ± 6.69
120	MAE	258.13 ± 84.36	259.22 ± 85.52	598.70 ± 21.27	260.72 ± 84.12
	RMSE	327.14 ± 102.52	330.78 ± 104.98	723.96 ± 15.20	331.86 ± 102.77
	sMAPE[a]	21.19 ± 6.91	21.25 ± 6.99	48.95 ± 1.82	21.37 ± 6.87
	MAPE[a]	21.13 ± 6.89	21.13 ± 6.92	50.63 ± 1.86	21.25 ± 6.81
168	MAE	259.88 ± 83.64	261.09 ± 85.15	526.24 ± 22.22	262.09 ± 83.84
	RMSE	332.81 ± 104.96	336.39 ± 107.47	650.74 ± 16.26	335.96 ± 104.70
	sMAPE[a]	21.31 ± 6.83	21.39 ± 6.94	42.91 ± 1.91	21.47 ± 6.83
	MAPE[a]	21.22 ± 6.78	21.25 ± 6.86	44.22 ± 1.93	21.31 ± 6.74
192	MAE	262.54 ± 85.69	265.74 ± 88.87	459.13 ± 25.10	264.54 ± 85.86
	RMSE	334.06 ± 105.36	341.63 ± 110.57	580.37 ± 19.47	338.50 ± 105.97
	sMAPE[a]	21.51 ± 6.99	21.74 ± 7.23	37.35 ± 2.15	21.66 ± 6.99
	MAPE[a]	21.41 ± 6.94	21.56 ± 7.12	38.32 ± 2.16	21.50 ± 6.90
216	MAE	263.62 ± 84.50	268.11 ± 88.78	433.33 ± 32.48	266.29 ± 85.06
	RMSE	333.45 ± 102.87	343.10 ± 109.36	551.45 ± 27.46	338.95 ± 103.90
	sMAPE[a]	21.62 ± 6.90	21.94 ± 7.22	35.24 ± 2.75	21.81 ± 6.93
	MAPE[a]	21.54 ± 6.88	21.75 ± 7.12	36.07 ± 2.78	21.66 ± 6.85
240	MAE	262.60 ± 83.28	266.17 ± 86.80	422.39 ± 37.95	265.03 ± 84.20
	RMSE	333.21 ± 102.40	341.50 ± 107.70	538.23 ± 32.97	339.21 ± 104.59
	sMAPE[a]	21.54 ± 6.80	21.78 ± 7.06	34.36 ± 3.21	21.70 ± 6.84
	MAPE[a]	21.49 ± 6.79	21.62 ± 6.97	35.12 ± 3.25	21.55 ± 6.75

[a] $\times 10^{-3}$.

- **Percent Profitable (PP)**. This metric is also known as the probability of winning and is calculated by dividing the number of winning trades by the total number of trades:

$$PP = \frac{\#T_W}{\#T}$$

4. Data

In this work we focus on daily stock exchange rates from three major indices: IBEX, DAX and DJI. The IBEX is the major stock exchange of Spain. It comprehends the 35 most liquid Spanish stocks traded in the Madrid Stock Exchange General Index. The DAX is the German stock index which measures the performance of the 30 largest companies according to order book volume and market capitalization. Finally, DJI is the stock exchange of industrial companies of the United States. It measures the stock performance of 30 large companies.

Data was collected from IG Group and covers a period going from January 1, 2011 to December 31, 2019. Each observation is described by five features. In particular, the features used are the date, the opening price, the closing price, the highest price and the lowest price. A summary of the data is shown in Table 2, where for each index, the starting and end days is given followed by the total number of observations.

5. Experiments

This section describes the experiments conducted to assess the quality of the proposed strategy. Such experiments can be summarized as follows:

(a) in Section 5.1, a comparison of the performances of the machine learning techniques used is proposed.
(b) the optimization of the technical analysis strategies TEMA and MACD is reported in Section 5.2.
(c) Section 5.3 presents the backtesting performed with the proposed hybrid trading strategy and its results compared with TEMA and MACD strategies.

In the experiments, each buy or sell order is executed the next day after a signal is generated. The order size is the same for all entrances. In order to fix the spread values, we have considered the values set by IG Group at regular trading hours in Europe. Thus, the spread of each trade is fixed to 5, 2, and 2.4 for *IBEX*, *DAX* and *DJI* respectively.

Table A.10
Results achieved by all the methods on DJI averaged over the prediction horizon h for each historical window w.

w	Metric	LM	NN	RF	SVR
6	MAE	483.86 ± 165.36	466.21 ± 154.70	4735.42 ± 7.68	488.15 ± 168.60
	RMSE	629.67 ± 190.43	625.06 ± 187.09	5060.00 ± 10.30	632.33 ± 191.88
	sMAPE[a]	19.36 ± 6.59	18.61 ± 6.14	207.42 ± 0.86	19.54 ± 6.73
	MAPE[a]	19.37 ± 6.62	18.56 ± 6.12	235.00 ± 1.15	19.56 ± 6.77
12	MAE	483.05 ± 164.29	466.93 ± 154.67	4680.76 ± 8.19	487.04 ± 166.35
	RMSE	628.65 ± 189.12	624.41 ± 186.13	5005.78 ± 10.99	631.06 ± 189.82
	sMAPE[a]	19.32 ± 6.55	18.64 ± 6.14	204.68 ± 0.88	19.49 ± 6.64
	MAPE[a]	19.33 ± 6.57	18.58 ± 6.12	231.55 ± 1.17	19.50 ± 6.67
24	MAE	482.09 ± 163.42	467.92 ± 154.92	4606.75 ± 6.39	486.23 ± 163.96
	RMSE	626.82 ± 187.64	623.31 ± 185.19	4930.32 ± 9.66	628.91 ± 187.72
	sMAPE[a]	19.27 ± 6.51	18.68 ± 6.15	200.95 ± 0.81	19.44 ± 6.54
	MAPE[a]	19.27 ± 6.53	18.61 ± 6.13	226.83 ± 1.07	19.45 ± 6.56
48	MAE	482.71 ± 162.10	469.09 ± 155.29	4654.86 ± 10.12	486.30 ± 160.86
	RMSE	628.25 ± 187.22	625.28 ± 185.52	4964.39 ± 12.78	630.84 ± 186.12
	sMAPE[a]	19.29 ± 6.46	18.72 ± 6.17	203.09 ± 1.01	19.44 ± 6.41
	MAPE[a]	19.28 ± 6.46	18.65 ± 6.14	229.36 ± 1.32	19.43 ± 6.42
72	MAE	482.15 ± 158.29	479.23 ± 157.60	4652.18 ± 11.89	484.41 ± 156.52
	RMSE	636.86 ± 187.59	636.76 ± 187.50	4951.54 ± 14.05	639.31 ± 184.80
	sMAPE[a]	19.22 ± 6.28	19.10 ± 6.25	202.76 ± 1.10	19.31 ± 6.21
	MAPE[a]	19.17 ± 6.26	19.03 ± 6.23	228.82 ± 1.42	19.27 ± 6.21
96	MAE	475.83 ± 154.16	474.28 ± 152.88	4633.87 ± 16.26	478.95 ± 151.32
	RMSE	631.39 ± 182.94	631.28 ± 182.74	4924.34 ± 17.84	634.76 ± 180.41
	sMAPE[a]	18.96 ± 6.12	18.90 ± 6.06	201.66 ± 1.31	19.09 ± 6.00
	MAPE[a]	18.90 ± 6.10	18.83 ± 6.04	227.34 ± 1.67	19.03 ± 5.99
120	MAE	479.78 ± 153.93	478.22 ± 153.35	4560.91 ± 16.41	488.52 ± 154.87
	RMSE	635.39 ± 183.19	635.38 ± 183.17	4845.72 ± 17.98	642.95 ± 183.80
	sMAPE[a]	19.11 ± 6.11	19.05 ± 6.08	197.93 ± 1.29	19.46 ± 6.14
	MAPE[a]	19.05 ± 6.09	18.98 ± 6.06	222.60 ± 1.64	19.40 ± 6.13
168	MAE	490.49 ± 155.99	474.29 ± 146.65	4584.66 ± 12.42	497.15 ± 153.16
	RMSE	644.57 ± 183.44	641.00 ± 180.86	4846.56 ± 14.56	651.35 ± 181.48
	sMAPE[a]	19.58 ± 6.22	18.90 ± 5.82	198.65 ± 1.11	19.82 ± 6.09
	MAPE[a]	19.56 ± 6.22	18.81 ± 5.79	223.24 ± 1.42	19.79 ± 6.08
192	MAE	496.02 ± 156.12	482.11 ± 149.13	4577.59 ± 16.49	503.18 ± 153.28
	RMSE	648.49 ± 183.05	645.82 ± 181.12	4829.97 ± 18.10	656.27 ± 181.69
	sMAPE[a]	19.78 ± 6.21	19.19 ± 5.91	198.11 ± 1.30	20.05 ± 6.09
	MAPE[a]	19.77 ± 6.22	19.11 ± 5.88	222.46 ± 1.65	20.02 ± 6.09
216	MAE	498.66 ± 157.25	485.03 ± 149.32	4544.00 ± 19.48	509.97 ± 157.85
	RMSE	649.19 ± 181.69	646.69 ± 179.51	4788.90 ± 20.16	661.68 ± 183.22
	sMAPE[a]	19.87 ± 6.26	19.30 ± 5.92	196.31 ± 1.44	20.30 ± 6.27
	MAPE[a]	19.86 ± 6.27	19.21 ± 5.88	220.13 ± 1.81	20.27 ± 6.28
240	MAE	509.90 ± 161.99	498.42 ± 156.53	4525.80 ± 23.90	518.01 ± 158.82
	RMSE	661.66 ± 188.15	660.06 ± 187.03	4761.43 ± 23.94	671.47 ± 185.66
	sMAPE[a]	20.31 ± 6.44	19.82 ± 6.21	195.22 ± 1.64	20.60 ± 6.31
	MAPE[a]	20.30 ± 6.45	19.74 ± 6.17	218.67 ± 2.04	20.58 ± 6.32

[a] $\times 10^{-3}$.

5.1. Analysis of the learning schemes

In this section the machine learning techniques used in this paper, i.e., LM, ANN, RF and SVR, are compared. The hyperparameter optimization for each learning scheme was performed using a grid search. To facilitate the readability of this section, only the parameters values found for each technique are shown in Table 3. Tables reporting the average results over the prediction horizon h, are presented in Appendix A. In order to select the optimal parameters, the best values, averaged over w and h, found for each technique were selected. These values are shown in Table 4.

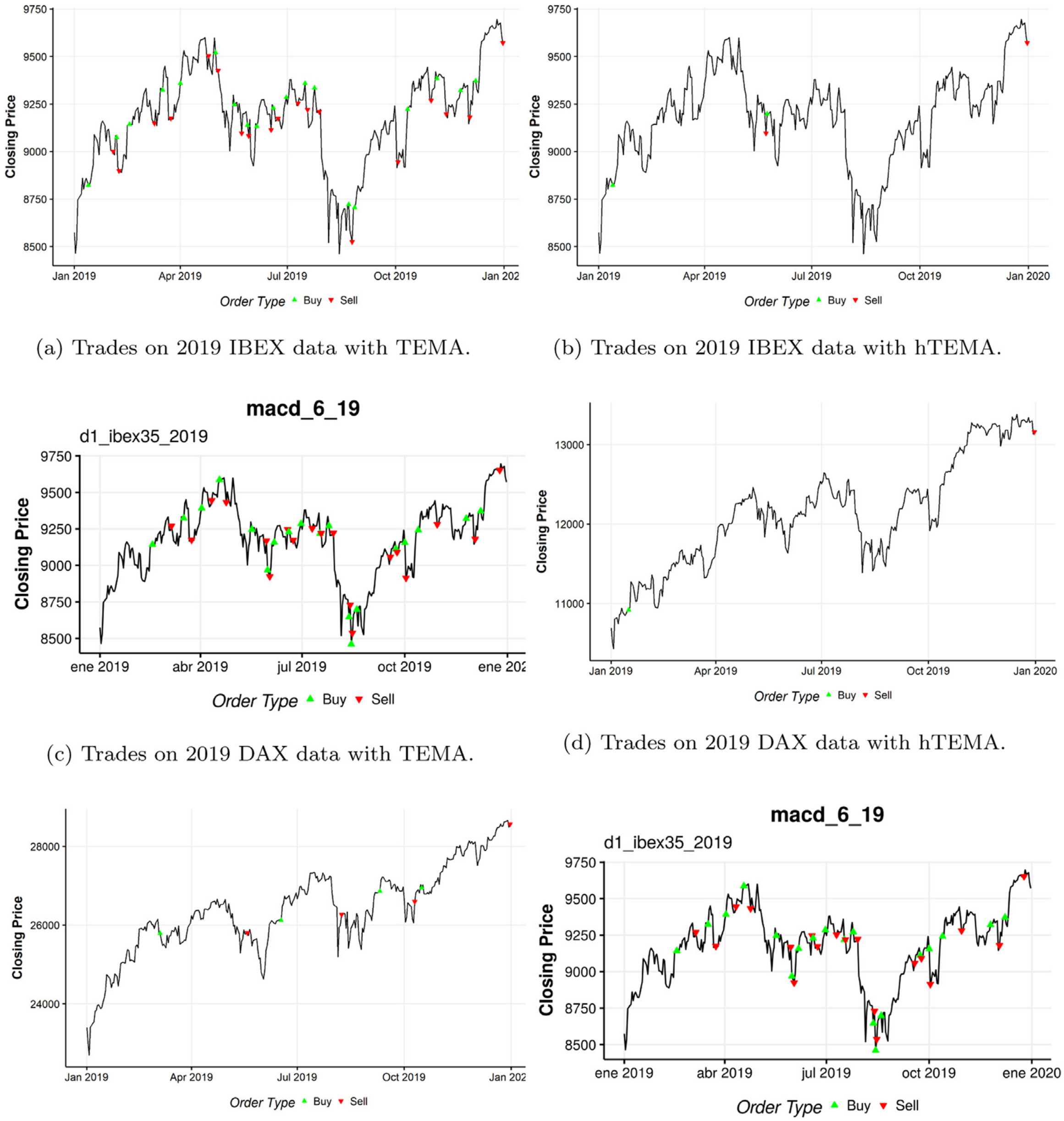
Table 4 reports the performance of each method on the test set. For each method and stock market we report the average over all historical windows w and prediction horizons h of the measures MAE, RMSE, sMAPE and MAPE, together with the standard deviation. As it can be noticed, the best results are achieved by LM and ANN, followed closely by SVR. RF models are, by far, the worse ones. It can be seen that the more the stock values vary, the higher the error is.

5.2. Optimize technical analysis strategies

Throughout this section, the performance of MACD and TEMA strategies will be analyzed. The optimal combination of parameter values are sought for with a grid search on data ranged from January 1, 2011 to December 31, 2018.

Table 5 presents the range of values considered for each strategy. For TEMA and MACD, the combination of parameter values were restricted to follow the rules *Fast* $<$ *Medium* $<$ *Slow* and *Fast* $<$ *Slow*, respectively.

The top 5 best combinations of parameters are presented in Table 6. We can notice that the behavior of both strategies differ when applied to the indices. For both trading strategies, the best results are achieved with DJI index, followed by DAX and IBEX. MACD outperforms TEMA on IBEX and DAX while it underperforms on DJI data. On IBEX, TEMA is a non Profitable strategy since it reaches values below 1. Despite MACD is Profitable, the values on test data may be unProfitable due to taxes and, furthermore, the expected return does not compensate the risk taken. On DAX, only in the case of MACD the return expected is interesting enough. However, if we take into account that the performance of the strategies are lower on new data, we can discard both strategies for a real trading. Finally, on DJI, the achieved results



(a) Trades on 2019 IBEX data with TEMA.

(b) Trades on 2019 IBEX data with hTEMA.

(c) Trades on 2019 DAX data with TEMA.

(d) Trades on 2019 DAX data with hTEMA.

(e) Trades on 2019 DJI data with TEMA.

(f) Trades on 2019 DJI data with hTEMA.

Fig. B.3. Buy and sell signals found by TEMA and hTEMA trading strategies.

can be considered competitive, especially for the case of TEMA, which reaches PF values close to 4. To sum things up, we can state that only on DJI the TEMA and MACD strategies are competitive and are expected to be Profitable according to the Profit factor.

5.3. Backtest the trading strategies

Finally, the trading strategies are backtested on 2019 data using the optimal parameter values of TEMA and MACD strategies. According to the results achieved in Section 5.1, LM and ANN are the best strategies. Following Occam's razor principle, we selected LM. In order to set the size of the historical windows w Tables A.8, A.9, and A.10 are taken into account. For each stock, the w with which LM achieved the lowest error was selected. The LM models were built setting w to 72, 12 and 96 for IBEX, DAX and DJI, respectively. To generate the trading signal, the model considered $h = 24$ since it allows to better capture the trend of the price.

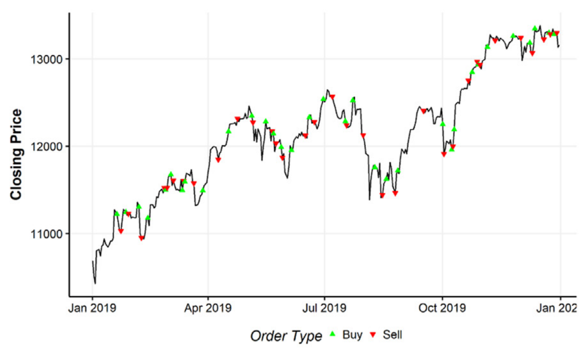
The results on test data are shown in Table 7, where, for each strategy and index, we report the best combination of parameters found on the training set, the total number of trades (#T), the Profit factor (PF), the net Profit (NP), the average Profit per trade ($\overline{T}$) and the maximum drawdown (D_{max}). Finally, in the last column, the percent Profitable (PP) is shown. The proposed strategies are denoted as hybrid TEMA (hTEMA) and hybrid MACD (hMACD). The values associated to the metrics NT, $\overline{T}$ and D_{max} are given in points.

As it can be seen, the hybridization yield an improvement of the Profit of each strategy for all indices as well as a reduction of the number of trades. As previously seen, on IBEX, the hybridization allow to have Profitable strategies. On DAX and DJI, despite having strategies positive NT, hTEMA and hMACD improved the Profitability. Furthermore, the proposed strategy does not present any lost trade.

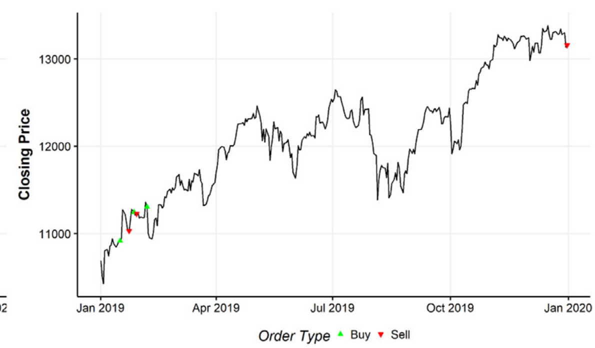
It can also be noted that only in the case of IBEX, the drawdown exceeds the net Profit. hMACD outperforms hTEMA on DAX

(a) Trades on 2019 IBEX data with MACD.

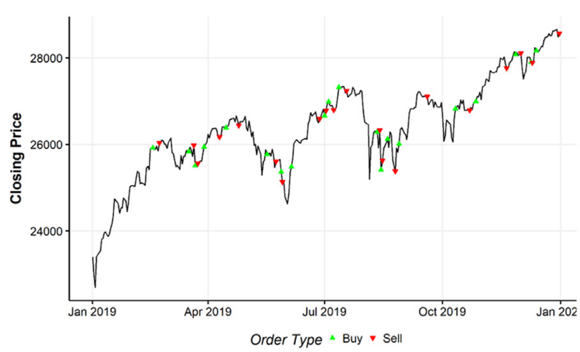
(b) Trades on 2019 IBEX data with hMACD.



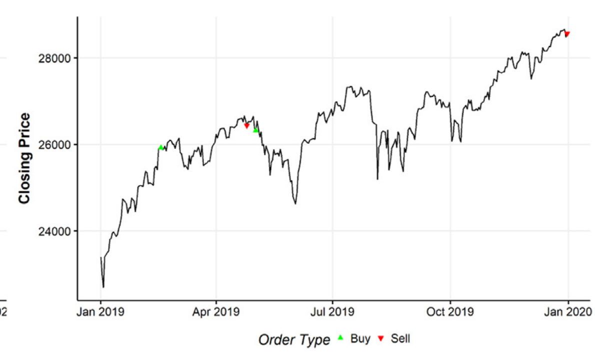
(c) Trades on 2019 DAX data with MACD.



(d) Trades on 2019 DAX data with hMACD.



(e) Trades on 2019 DJI data with MACD.



(f) Trades on 2019 DJI data with hMACD.

Fig. B.4. Buy and sell signals found by MACD and hMACD trading strategies.

and DJI while it underperforms on IBEX. If, for example we have a portfolio of $10000, and each point is supposed to be equivalent to $1, then the total Profit on IBEX would be of $866.1 and $300.5 for hTEMA and hMACD respectively. For DAX, the return would be of $2211.8 and $1817.1 and, finally, for DJI, of $2678 and $2738.9.

The trades with TEMA and hTEMA on IBEX, DAX and DJI are shown in Fig. B.3 while for MACD and hMACD are in Fig. B.4.

6. Conclusions and future work

In this work a novel trading decision making workflow has been proposed to generate effective buy or sell signals. In this proposal, the trading rules are based on hybridizing technical analysis rules with the predictive power of machine learning models.

From among all the machine learning techniques tested, LM and ANN were the ones that performed best. The good performance of ANN is well known in stock market prediction and, for that reason, has been extensively used in previous work. The good performance of LM could suggest that for smaller period of time the linear model is suitable for prediction purposes.

We have tested our proposal with TEMA and MACD trading strategies, and we have proved the our strategy helped in obtaining superior results. Hybridization not only has improved the Profit but it also has decreased the number of trades as well as the risk of losses.

It is also worth noticing that, for each algorithm used in the proposed workflow, the optimal parameters depended on the index analyzed. Despite the good results achieved, more research is necessary to enhance the understanding of the proposed workflow and trading rules.

As it has been shown in this work, the hybrid trading strategies achieve good results. However the prediction horizon length *h* can be optimized. Therefore, as future work, we intend to carry out the optimization of h and validate it on new real data from other stock market indices, as well as on foreign exchange market data. Another possible improvement of the strategy is to incorporate trend information by addressing the problem as a classification task. We also intend to address the exploration of more technical analysis strategies and new trading rules. Finally, in order to gain more insight about the predictive models, in future works more metrics will be included in the study, such as precision, recall and F1-score [60] among many others available.

CRediT authorship contribution statement

Jordan Ayala: Software, Visualization, Investigation. **Miguel García-Torres:** Conceptualization, Methodology, Investigation, Writing, Supervision. **José Luis Vázquez Noguera:** Reviewing. **Francisco Gómez-Vela:** Writing, Supervision. **Federico Divina:** Writing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Part of the computer time was provided by the Centro Informático Científico de Andalucía (CIC).

Appendix A. Results

The tables shown in this section report, for each metric, the average results together with its standard deviation. Such values are computed over the prediction horizon h for each historical window w.

Appendix B. Trades

See Figs. B.3 and B.4.

Appendix C. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.knosys.2021.107119.

References

[1] P.F. Pai, C.S. Lin, A hybrid ARIMA and support vector machines model in stock price forecasting, Omega 33 (6) (2005) 497–505.
[2] L.Y. Wei, A hybrid model based on ANFIS and adaptive expectation genetic algorithm to forecast TAIEX, Econ. Model. 33 (2013) 893–899.
[3] K. Kanjamapornkul, R. Pinčák, Kolmogorov space in time series data, Math. Methods Appl. Sci. 39 (15) (2016) 4463–4483.
[4] B. Al-hnaity, M. Abbod, Intelligent Systems and Applications. Studies in Computational Intelligence, 650, Springer, 2016, pp. 19–41 (Chapter Predicting financial time series data using hybrid mode).
[5] A.M. Awajan, M.T. Ismail, S.A. Wadi, Improving forecasting accuracy for stock market data using EMD-HW bagging, PLoS One 13 (7) (2018) e0199582.
[6] H.V. Roberts, Stock-market "patterns" and financial analysis: Methodological suggestions, J. Finance 14 (1) (1959) 1–10.
[7] A.W. Lo, The New Palgrave: A Dictionary of Economics, second ed., Palgrave Macmillan Ltd., 2007 (Chapter efficient markets hypothesis).
[8] A.W. Lo, The adaptive markets hypothesis, J. Portfolio Manag. 30 (5) (2004) 15–29.
[9] E. Ahmadi, M. Jasemi, L. Monplaisir, M.A. Nabavi, A. Mahmoodi, P.A. Jam, New efficient hybrid candlestick technical analysis model for stock market timing on the basis of the support vector machine and heuristic algorithms of imperialist competition and genetic, Expert Syst. Appl. 94 (2018) 21–31.
[10] N. Petrusheva, I. Jordanoski, Comparative analysis between the fundamental and technical analysis of stocks, J. Process Manag. New Technol. Int. 4 (2) (2016) 26–31.
[11] U. Khan, F. Aadil, M.A. Ghazanfar, S. Khan, N. Metawa, K. Muhammad, I. Mehmood, Y. Nam, A robust regression-based stock exchange forecasting and determination of correlation between stock markets, Sustainability 10 (2018) 3702–3722.
[12] A.A. Ariyo, A.O. Adewumi, C.K. Ayo, Stock price prediction using the ARIMA model, in: 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, 2014, pp. 106–112.
[13] S.A. Wadi, M. Almasarweh, A.A. Alsaraireh, Predicting closed price time series data using ARIMA model, Modern Appl. Sci. 12 (11) (2018) 181–185.
[14] A.H. Moghaddam, M.H.M.M. Esfandyari, Stock market index prediction using artificial neural network, J. Econ. Finance Adm. Sci. 21 (41) (2016) 89–93.
[15] D. Selvamuthu, V. Kumar, A. Mishra, Indian stock market prediction using artificial neural networks on tick data, Financ. Innov. 5 (16) (2019) 1–12.
[16] S.C. Nayak, B. Misra, H. Behera, Artificial chemical reaction optimization of neural networks for efficient prediction of stock market indices, Ain Shams Eng. J. 8 (3) (2017) 371–390.
[17] O.B. Sezer, M. Ozbayoglu, E. Dogdu, A deep neural-network based stock trading system based on evolutionary optimized technical analysis parameters, Procedia Comput. Sci. 114 (2017) 473–480.
[18] M. Hiransha, E.A. Gopalakrishnan, V.K. Menon, K.P. Soman, NSE stock market prediction using deep-learning models, Procedia Comput. Sci. 132 (2018) 1351–1362.
[19] B.M. Henrique, V.A. Sobreiro, H. Kimura, Stock price prediction using support vector regression on daily and up to the minute prices, J. Finance Data Sci. 4 (3) (2018) 183–201.
[20] S. Basak, S. Kar, S. Saha, L. Khaidem, S.R. Dey, Predicting the direction of stock market prices using tree-based classifiers, North Am. J. Econ. Finance 47 (2019) 552–567.
[21] J. Chen, H. Hong, J.C. Stein, Forecasting crashes: trading volume,past returns, and conditional skewness in stock prices, J. Financ. Econ. 61 (2001) 345–381.
[22] P.J. Brockwell, R.A. Davis, Introduction to Time Series and Forecasting, Springer, 2002.
[23] J. Yao, C. Tan, H.-L. Poh, Neural networks for technical analysis: A study on KLCI, Int. J. Theor. Appl. Finance 2 (2) (1999) 221–241.
[24] J.V. Pérez-Rodríguez, S. Torra, J. Andrada-Félix, STAR and ANN models: forecasting performance on the Spanish Ibex-35 stock index, J. Empir. Financ. 12 (2005) 490–509.
[25] S.S. Alexander, Price movements in speculative markets: trends or random walks, Ind. Manag. Rev. 2 (1961) 7–26.
[26] P. Chang, C. Fan, C. Liu, Integrating a piecewise linear representation method and a neural network model for stock trading points prediction, IEEE Trans. Syst. Man Cybern. C 39 (2009) 80–92.
[27] L.A. Teixeira, A.L.I. de Oliveira, A method for automatic stock trading combining technical analysis and nearest neighbor classification, Expert Syst. Appl. 37 (2010) 6885–6890.
[28] R. Dash, P.K. Dash, A hybrid stock trading framework integrating technical analysis with machine learning techniques, J. Finance Data Sci. 2 (2016) 42–57.
[29] C. Sang, M.D. Pierro, Improving trading technical analysis with TensorFlow Long Short-Term Memory (LSTM) Neural Network, J. Finance Data Sci. 5 (2019) 1–11.
[30] M. Agrawal, A.U. Khan, P.K. Shukla, Stock price prediction using technical indicators: A predictive model using optimal deep learning, Int. J. Recent Technol. Eng. 8 (2) (2019) 2297–2305.
[31] A.A. Aguirre, R.R. Medina, N.D. Méndez, Machine learning applied in the stock market through the Moving Average Convergence Divergence (MACD) indicator, Invest. Manag. Financ. Innov. 17 (4) (2020) 44–60.
[32] Q. Chen, W. Zhang, Y. Lou, Forecasting stock prices using a hybrid deep learning model integrating attention mechanism, multi-layer perceptron, and bidirectional long-short term memory neural network, IEEE Access 8 (2020) 117365–117376.
[33] A. Picasso, S. Merello, Y. Ma, L. Oneto, E. Cambria, Technical analysis and sentiment embeddings for market trend prediction, Expert Syst. Appl. 135 (2019) 60–70.
[34] T.T. Souza, T. Aste, Predicting future stock market structure by combining social and financial network information, Physica A 535 (2019) 122343.
[35] X. Li, P. Wu, W. Wang, Incorporating stock prices and news sentiments for stock market prediction: A case of Hong Kong, Inf. Process. Manage. 57 (5) (2020) 102212.
[36] R. Talib, M.K. Hanif, S. Ayesha, F. Fatima, Text mining: Techniques, applications and issues, Int. J. Adv. Comput. Sci. Appl. 7 (11) (2016).
[37] H. Liu, B. Song, Stock trends forecasting by multi-layer stochastic ANN bagging, in: 2017 IEEE 29th International Conference on Tools with Artificial Intelligence, ICTAI, 2017, pp. 322–329.
[38] I.K. Nti, A.F. Adekoya, B.A. Weyori, A comprehensive evaluation of ensemble learning for stock-market prediction, J. Big Data 7 (20) (2020) 1–40.
[39] Y. Nojima, Stacked deep learning structure with bidirectional long-short term memory for stock market prediction, in: Proceedings of the Neural Computing for Advanced Applications: First International Conference, NCAA 2020, 1265, 2020, p. 447.
[40] I. Ghosh, T.D. Chaudhuri, FEB-stacking and FEB-DNN models for stock trend prediction: A performance analysis for pre and post Covid-19 periods, Decis. Making: Appl. Manag. Eng. 4 (1) (2021) 51–84.

[41] F. Zhou, Q. Zhang, D. Sornette, L. Jiang, Cascading logistic regression onto gradient boosted decision trees for forecasting and trading stock indices, Appl. Soft Comput. 84 (2019) 105747.
[42] T.J. Strader, J.J. Rozycki, T.H. Root, Y.-H. Huang, Machine learning stock market prediction studies: Review and research directions, J. Int. Technol. Inf. Manag. 28 (4) (2020) 63–83.
[43] R. Choudhry, K. Garg, A hybrid machine learning system for stock market forecasting, World Acad. Sci. Eng. Technol. 39 (3) (2008) 315–318.
[44] S. Shen, H. Jiang, T. Zhang, Stock Market Forecasting Using Machine Learning Algorithms, Department of Electrical Engineering, Stanford University, Stanford, CA, 2012, pp. 1–5.
[45] P.D. Yoo, M.H. Kim, T. Jan, Machine learning techniques and use of event information for stock market prediction: A survey and evaluation, in: International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, CIMCA-IAWTIC'06, 2, 2005, pp. 835–841.
[46] J. Patel, S. Shah, P. Thakkar, K. Kotecha, Predicting stock market index using fusion of machine learning techniques, Expert Syst. Appl. 42 (4) (2015) 2162–2172.
[47] J. Neter, M.H. Kutner, C.J. Nachtsheim, W. Wasserman, Applied Linear Statistical Models, Irwin, Chicago, 1996.
[48] M. Kuhn, Building predictive models in R using the caret package, J. Stat. Softw. 28 (5) (2008) 1–26.
[49] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice Hall, 1994.
[50] W. Venables, B.D. Ripley, Modern Applied Statistics with S, fourth ed., Springer, New York, 2002.
[51] T.K. Ho, Random decision forests, in: Proceedings of 3rd International Conference on Document Analysis and Recognition, 1, 1995, pp. 278–282.
[52] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[53] A. Liaw, M. Wiener, Classification and regression by randomforest, R News 2 (3) (2002) 18–22.
[54] M. Awad, R. Khanna, Support vector regression, in: Efficient Learning Machines, Springer, 2015, pp. 67–80.
[55] S. Makridakis, Accuracy measures: theoretical and practical concerns, Int. J. Forecast. 9 (4) (1993) 527–529.
[56] C. Chen, J. Twycross, J.M. Garibaldi, A new accuracy measure based on bounded relative error for time series forecasting, PLoS One 12 (3) (2017) e0174202.
[57] G. Appel, The Moving Average Convergence-Divergence Method, Technical Report, Signalert.
[58] F. Divina, A. Gilson, F. Goméz-Vela, M. García Torres, J.F. Torres, Stacking ensemble learning for short-term electricity consumption forecasting, Energies 11 (4) (2018).
[59] L.B. Mendelsohn, J.J. Murphy, Trend Forecasting with Technical Analysis: Unleashing the Hidden Power of Intermarket Analysis to Beat the Market, Marketplace Books, 2000.
[60] L. Torgo, R. Ribeiro, Precision and recall for regression, in: Discovery Science, in: Lecture Notes in Computer Science, vol. 5808, 2009, pp. 332–346.