IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received November 17, 2020, accepted December 14, 2020, date of publication December 24, 2020, date of current version January 15, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3047160

Stock Forecasting Using Local Data

GERARDO ALFONSO[1,2], A. DANIEL CARNERERO[2], DANIEL R. RAMIREZ[2], AND TEODORO ALAMO[2]

[1]Shenwan & Hongyuan Securities Company Ltd. (SWS), Shanghai 200031, China
[2]Departamento de Ingeniería de Sistemas y Automática, Universidad de Sevilla, 41092 Sevilla, Spain

Corresponding author: Gerardo Alfonso (ga284@cantab.net)

This work was supported by the Ministerio de Ciencia e Innovación of Spain under Project PID2019-106212RB-C41.

ABSTRACT Stock price forecasting is a relevant and challenging problem that has attracted a lot of interest from engineers and scientists. In this paper we apply two techniques for stock price and price intervals forecasting. Both techniques, derived from previous works by the authors, are based on the use of local data extracted from a database. These data are those that correspond to similar market states to the current one. The first technique uses these local data to compute a price forecast by finding an optimal combination of past states that equals the current state. The price forecast is then obtained by combining the past actual prices associated to the past market states. The second technique can be used to forecast prices but its main use is to forecast price intervals that will contain the real future price with a guaranteed probability. This is accomplished by building a probability distribution for the forecasted price and then setting the intervals by a choice of desired percentiles. Thus, this technique can be used in financial risk management. Both techniques are purely data driven and do not need a theoretical description or model of the price trend being forecasted. The proposed techniques adapt very easily to market changes because they use only the subset of the database that it is closer to the current state. Furthermore, the database can be updated as new data is available. Finally, both approaches are highly parallelizable, thus making possible to manage large data sets. As a case study, the proposed approaches have been applied to the k-step forecasting of the Dow Jones Industrial Average index. The results have been validated in relation with some baseline approaches, such as martingale and neural network predictors and quantile regression for the interval forecasting.

INDEX TERMS Stock forecasting, probabilistic interval forecasting, direct weight optimization, data driven methods.

I. INTRODUCTION

Stock price forecasting is a challenging field that has attracted researchers from different fields including engineers and scientists. It is likely fair to say that there is not yet an approach for stock forecasting that is accepted as superior, with the existing approaches having their strengths and weaknesses. There are two major approaches regarding stock forecasting [1]. In the traditional financial approach, the objective is to estimate the intrinsic value of a security [2]–[4]. For instance, in the case of a stock, the intrinsic value from a financial traditional approach is the time discounted value of the free cash flow of the company [5], [6]. There are several issues regarding the intrinsic value approach, such as the need to forecast the cash flows of the company for several years into the future [7] which makes estimates rather challenging. Another issue, from an investor point of view, is that the intrinsic value of the stock and the actual value of the stock, i.e., its price, are not the same.

Another investing approach can be defined as quantitative investment [8]. In this type of investment strategy the objective is not to estimate the intrinsic value (true value) of a security but the price (or price trend) that the stock will follow over a certain time horizon, which tends to be short to midterm investment [9], [10]. The assumption on this type of investment strategy is that the price of a security is dictated by the demand and supply forces and that, as such, statistical and learning methods can be used to forecast the price. This approach implicitly assumes that stock prices do not follow a perfect random walk. If they did, then historical price information would be useless when trying to determine future stock prices [11]. This assumption can be related to the efficient market hypothesis [11] which is a long-standing topic of debate in the financial academic literature. The strong

The associate editor coordinating the review of this manuscript and approving it for publication was Grigore Stamatescu.

form of this hypothesis assumes that all public information, both private and public, is already priced in stock levels and it is therefore impossible to develop any trading strategy even for company insiders. There are other approaches to financial theory that suggest that the market are not completely efficient because investors are subject to biases and do not have access to all the information that can potentially impact the price of the stock of a company [12]. Thus, forecasting the stock price using historical data would be a difficult but possible task. Due to the financial relevance of stock price forecasting, many different techniques have been applied to the problem. The almost random nature of the market has made Brownian motion [13] and martingale models [14], [15] one of the first choices. Since the efficient market hypothesis is not proved, more elaborate techniques have been used trying to exploit the market inefficiencies. Among these techniques, in the literature can be found applications with linear models [16], support vector machines [17], genetic algorithms [18] or more frequently neural networks [19]–[21] and deep learning methods [22]–[24] (for a recent survey on this topic see [25] or [26] for a more general survey).

In this paper, we present two quantitative techniques that as far as we know have not been covered in the existing literature in the field of stock forecasting. One is an approach derived from the predictive control strategy presented by the authors in [27], [28], which in turn can be related to the direct weight optimization approach [29]–[31]. Direct weight optimization uses linear estimators and convex optimization [32] and has been applied in different fields like predictive control [33], nonlinear system identification [34] or electron density analysis [35]. The proposed approach uses local data, that is, only a subset of the whole data available, chosen among those past stock market states that are close to the current state. With such a subset, the approach computes an optimal linear combination of past states that equals the current state, using such combination then to compute the price forecast. Unlike other methods, like neural networks, the proposed approach does not use a training phase as the subset and the linear combination is computed each time a forecast is needed. This allows an easy adaptation to different market situations and also the updating of the database as new data are available without having to retrain the estimator. Furthermore, the use of local data results in a lower computational burden, as the cardinality of the subset will be much lower than that of the whole data set.

The other technique proposed in the paper is a probabilistic price interval strategy previously presented in a more general context in [36]. This strategy can be used to forecast stock prices but its main application is to provide price intervals with a guaranteed probability of containing the real price. In this sense is complementary to the first approach as it provides the guarantees that the previous one lacks. On the other hand, although the algorithm is highly parallelizable, the computational burden is higher, thus it does not replace the first approach if no guarantees are required. This approach uses dissimilarity functions evaluated on local data to build an empirical probability distribution of the predicted price. Thus using such distribution it is possible to build price intervals using the desired percentiles and also predict the forecast using the median of the distribution. This can be very useful for risk management purposes [37], a field of increasing importance in finance.

Finally, as a case study, the techniques have been applied to the task of predicting future values of the Dow Jones Industrial Average Index up to 5 days (i.e., a full week), validating the results in relation to two baseline approaches, a persistence (martingale) predictor and a neural network based predictor. Furthermore, quantile regression [38], [39] has been used as a third baseline approach to validate the predicted price intervals. The results prove that the proposed techniques are a valuable tool that can be added to the portfolio of existing techniques for stock price forecasting.

The paper is organized as follows: section II presents the first strategy for stock price forecasting using local data. The probabilistic price interval strategy is shown in section III. Section IV presents the results of these two strategies when used to forecast the Dow Jones Average Industrial Index. Finally, the conclusions are presented in section V.

II. STOCK PRICE FORECASTING USING LOCAL DATA

The first approach used in this paper to forecast stock prices is based on the technique presented by the authors for predictive control in [27], [28].

Consider the evolution of the price of a stock as a time series $p(t) \in \mathcal{P}$, being t the time index expressed in the proper time unit, usually days in the case of the daily market, and $\mathcal{P}$ the possible range of values for the stock price. The state of the price time series is described as the value at time t of series of technical indicators, i.e.,

$$z(t) = (Z_1(t), Z_2(t), \dots, Z_{nz}(t)) \in \Re^{nz}.$$

These technical indicators can be past values of $p(t)$, stock price returns or more complex metrics like moving averages or the relative strength indicator amongst others [40]. The objective is to be able to predict k-steps ahead the price of a stock, that is, to obtain $\hat{p}(t+k)$ at time t in such a way that it is as close as possible to $p(t+k)$.

The approach presented here uses a database of past values of $z(t)$ and k-step ahead stock prices. The database DB_k for predicting k-steps ahead the price of $p(t)$ will be implemented as a table with N_{DB} entries (i.e., rows), in which each entry contains a past value of $z(t)$ and the corresponding $p(t+k)$, as shown in table 1.

TABLE 1. Structure of the database DB_k.

$z(i_1)$	$p(i_1+k)$
$z(i_2)$	$p(i_2+k)$
$z(i_3)$	$p(i_3+k)$
$\vdots$	$\vdots$

Note that the time indexes i_j of the past states do not have to be ordered in any way or be consecutive, thus they are not required to form a proper time series. The only requisite is that the price associated with $z(i_j)$ is the one corresponding to k steps after time i_j.

The proposed approach does not use the database to train or fit a predictor, as in the training of a neural network. Thus, the database is not considered a training set (except for the tuning of a reduced number of hyperparameters). Instead, it is used every time a prediction is needed in an oracle fashion [41]. Furthermore, the approach considered in this section uses only a subset of the database, denoted as $\Omega(z(t))$, to compute the prediction $\hat{p}(t+k)$. In this sense, it is also different from techniques like neural networks in which the estimator is fitted using the whole training set. More precisely, given a distance measurement function $d(z(t), z(i_j))$, and cardinality parameter N, the elements of $\Omega(z(t))$ are obtained selecting the N states $z(i_j)$ closer to $z(t)$. Thus, the prediction is computed using only local data.

Once the data that are to be included in $\Omega(z(t))$ are selected, the proposed approach proceeds to compute an optimal combination of all the $z(i_j)$ in $\Omega(z(t))$ that matches $z(t)$, using the weights of such combination to compute $\hat{p}(t+k)$ as the corresponding combination of all the $p(i_j+k)$ in $\Omega(z(t))$. Furthermore, a regularization term, weighted by a scalar $\gamma \geq 0$, is included in the computation of the optimal combination. Algorithm 1 gives a formal description of the proposed approach.

Remark 1: The distance $d(\cdot,\cdot)$ can be any measure of how close are the states $z(i_j)$ stored in DB_k to $z(t)$. A typical choice would be the Euclidean distance, i.e., $d(z(t), z(i_j)) = \|z(t) - z(i_j)\|$, but also could consider other aspects like the time span between states, i,e.,

$$d(z(t), z(i_j)) = \|z(t) - z(i_j)\| + \rho|t - i_j|,$$

where the non-negative scalar ρ would be a weighting factor. In this way, recent data would be prioritized in the selection process of step 3 in algorithm 1. Other aspects like seasonality could also be taken into account using the modulus operator, and, in general, many of the resemblance measures used in cluster analysis [42].

The optimization problem in step 4 of algorithm 1 can be easily solved, specially when $\gamma = 0$ as it results in a QP problem with equality constraints whose solution is that of a system of linear equations. Moreover, steps 1 and 2 can be easily parallelized, thus efficient implementations of algorithm 1 can be obtained.

Note that the fact that only local data is used to compute $\hat{p}(t+k)$ makes the strategy adaptive, being the definition of $d(\cdot,\cdot)$ the way to change how the strategy adapts to the current price variations. Finally, the proposed approach does not require a training phase (except for the possible tuning of the hyperparameter $\gamma \geq 0$), thus new data can be included in the database as they are available, without having to retrain the predictor. As in Lasso approaches [43], larger values of γ tend to make a larger fraction of the weights equal to zero, providing an enhanced local approach approximation. Thus,

Algorithm 1 k-Step Ahead Stock Forecasting Using Local Data

Input: DB_k, $z(t)$, N and γ.

Output: $\hat{p}(t+k)$ (estimation of price at $t+k$).

1: Compute the distance $d(z(t), z(i_j))$ for all $z(i_j)$ in the database DB_k.

2: Create a list of the entries in DB_k sorted according to the distances $d(z(t), z(i_j))$. Denote as z_l and $p_{l,k}$ the state $z(i_j)$ and k-step ahead price $p(i_j+k)$ of the l-th entry in this ordered list.

3: Build $\Omega(z(t))$ using the first N entries in the ordered list, that is,

$$\Omega(z(t)) \triangleq \left\{(z_l, p_{l,k})\right\} \quad \forall l \in \{1, \ldots, N\}.$$

4: Solve the following QP problem:

$$\begin{aligned} \min_{\lambda_1, \lambda_2, \ldots, \lambda_N} \quad & \sum_{l=1}^{N} \lambda_l^2 + \gamma|\lambda_l| \\ \text{s.t.} \quad & \sum_{l=1}^{N} \lambda_l = 1, \\ & \sum_{l=1}^{N} \lambda_l z_l = z(t). \end{aligned}$$

5: Compute $\hat{p}(t+k)$ as:

$$\hat{p}(t+k) = \sum_{l=1}^{N} \lambda_l p_{l,k}.$$

γ is an hyperparameter that potentially improves the quality of the predictions.

III. PROBABILISTIC PRICE INTERVAL FORECASTING

The price forecasting approach presented in the previous section provides an easy and convenient way of forecasting stock prices k-steps ahead. However, this approach does not provide any measure on how the real price could deviate from the forecasted one and also it does not have any guarantee on that deviation. In this section, the approach presented by the authors in [36] is adapted for stock price interval forecasting with probabilistic guarantees. The proposed methodology computes an interval prediction for the price $p(t+k)$ in which the lower and upper bound of the interval are computed taking into account given probabilistic specifications. The use of local data is introduced here in the strategy to better handle large databases. The reader is referred to [36] for a full description of the procedures involved in the original strategy. Here it will be shown the main concepts and implementation details.

The proposed strategy is based on building an empirical conditional probability distribution for $p(t+k)$ subject to $z(t)$. Let $p_{\underline{\alpha}}$ be the $\underline{\alpha}$-th percentile and $p_{\overline{\alpha}}$ the $\overline{\alpha}$-th percentile of such distribution. Then, the interval $[p_{\underline{\alpha}}, p_{\overline{\alpha}}]$ will contain the price with a probability of $\frac{\overline{\alpha}-\underline{\alpha}}{100}$. Thus, finding the intervals amounts to compute the lowest and highest percentile that define the desired interval for a given probability, e.g., for a probability of 0.8 the desired interval will be $[p_{10}, p_{90}]$. On the other hand, if a forecast $\hat{p}(t+k)$ is also needed, it can be chosen as the 50th percentile of the distribution.

The key concept in this approach is that of dissimilarity function, a generalization of the optimization problem in algorithm 1. A dissimilarity function measures how similar the given pair $(z(t), p)$ is to the set of pairs $(z_l, p_{l,k})$ of $\Omega \subseteq DB_k$. The formal definition of dissimilarity function is given in the following.

Definition 1: Given $\Omega \subseteq DB_k$, a scalar $\gamma \geq 0$, $z(t)$ and price p then the dissimilarity function $J_\gamma(\cdot,\cdot,\cdot)$ is defined as:

$$\begin{aligned} J_\gamma(z(t), p, \Omega) = \quad & \min_{\lambda_1,\lambda_2,\ldots,\lambda_N} \sum_{l=1}^{N} \lambda_l^2 + \gamma|\lambda_l| \\ & \text{s.t.} \sum_{l=1}^{N} \lambda_l = 1, \\ & \sum_{l=1}^{N} \lambda_l z_l = z(t), \\ & \sum_{l=1}^{N} \lambda_l p_{l,k} = p, \end{aligned}$$

where, as in section II, the N pairs $(z_l, p_{l,k}) \in \Omega$ denote the state $z(i_j)$ and k-step ahead price $p(i_j+k)$ of the l-th entry in Ω.

The dissimilarity function J_γ has a lower value when it is easy to represent $(z(t), p)$ as a combination of the N pairs $(z_l, p_{l,k})$ of Ω and a higher value otherwise (for further details see [36]). Notice that given $z(t)$, the value of p that minimizes the dissimilarity function $J_\gamma(z(t), p, \Omega)$ is equal to the k-step ahead forecast of Algorithm 1.

The other key concept in the approach is the empirical conditional probability density function (ecp) [36], that uses the dissimilarity function of definition 1, and approximates the real distribution of $p(t+k)$ conditioned to $z(t)$.

Definition 2: For a given $\Omega \subseteq DB_k$, $\gamma \geq 0$, $c > 0$, $z(t)$ and price p, the empirical conditional probability density function (pdf) ecp is defined as:

$$\text{ecp}_{\gamma,c}(z(t), p, \Omega) = \frac{e^{-cJ_\gamma(z(t),p,\Omega)}}{\int_{\mathcal{P}} e^{-cJ_\gamma(z(t),\breve{p},\Omega)} d\breve{p}},$$

where $\mathcal{P}$ is the set of all possible values of $p(t+k)$ for all $t+k$.

Note that according to this definition, given $z(t)$, the probability of $p(t+k)$ being in a certain interval $[p_a, p_b]$ is approximately equal to

$$\int_{p_a}^{p_b} \text{ecp}_{\gamma,c}(z(t), p, \Omega) dp.$$

To obtain the interval prediction $[p_{\underline{\alpha}}, p_{\overline{\alpha}}]$, the hyperparameters γ and c are chosen to make the approximation as sharp as possible for $p_a = p_{\underline{\alpha}}$ and $p_b = p_{\overline{\alpha}}$.

The hyperparameter c affects how the prices are distributed around its expected values. Higher values of c yield a more narrow pdf. Lower values of γ are appropriate when the real pdf is close to a normal distribution whereas higher values of γ are to be used if the distribution is flat and close to a uniform distribution. Thus, the family of empirical distributions parameterized with c and γ encompasses a broad range of distributions [36].

Note however that, in practice, the integral in definition 2 should be computed numerically over a finite set of possible values of $p(t+k)$, denoted as $\mathcal{P}_s \subset \mathcal{P}$ obtained from a grid of N_p values $\bar{p}_i$ in the interval $[p_{min}, p_{max}]$[1] with $\bar{p}_1 = p_{min}$ and $\bar{p}_{N_p} = p_{max}$. Denote the increment between two successive prices $\bar{p}_i \in \mathcal{P}_s$ as:

$$\Delta\bar{p} = \bar{p}_{i+1} - \bar{p}_i$$

Then, the approximation of the ecp will be computed as

$$\text{ecp}_{\gamma,c}(z(t), p, \Omega) \approx \frac{e^{-cJ_\gamma(z(t),p,\Omega)}}{I_S}, \tag{1}$$

where the approximation of the integral can be computed using the trapezoidal rule[2] obtaining

$$I_S = \Delta\bar{p} \sum_{i=1}^{N_p-1} \frac{e^{-cJ_\gamma(z(t),\bar{p}_{i+1},\Omega)} + e^{-cJ_\gamma(z(t),\bar{p}_i,\Omega)}}{2}.$$

Once the empirical distribution of $p(t+k)$ is obtained, computing the desired percentiles requires to find the value $p_{\overline{\alpha}}$ for which

$$\int_{p_{min}}^{p_{\overline{\alpha}}} \text{ecp}_{\gamma,c}(z(t), p, \Omega) dp = \frac{\overline{\alpha}}{100}.$$

holds and repeating the operation for $\underline{\alpha}$ to obtain $p_{\underline{\alpha}}$. As in the previous case, these integrals should be computed numerically. In the case of finding $p_{\overline{\alpha}}$ and using the trapezoidal approximation, it reduces to solve:

$$\begin{aligned} i_{\overline{\alpha}} = \quad & \arg\min_i \; i \\ & \text{s.t.} \sum_{j=1}^{i} \frac{\varphi_{j+1} + \varphi_j}{2} \geq \frac{\overline{\alpha}}{100\Delta\bar{p}}, \end{aligned} \tag{2}$$

where $\varphi_j = \text{ecp}(z(t), \bar{p}_j, \Omega)$ is computed as in (1), and then

$$p_{\overline{\alpha}} = \bar{p}_{i_{\overline{\alpha}}+1} \in \mathcal{P}_s. \tag{3}$$

[1]The choice of p_{min} and p_{max} can be done arbitrarily conservative, as the only requisite is that with a high probability any $p(t+k)$ verifies that $p(t+k) \in [p_{min}, p_{max}]$. However, it is better to use reasonably tight bounds that require a lower N_p to sample the interval correctly.

[2]More accurate methods can be used instead of the trapezoidal rule, which has been chosen here because of its simplicity and low requirements on the function to be integrated.

On the other hand, finding the lower percentile $p_{\underline{\alpha}}$ requires to solve:

$$i_{\underline{\alpha}} = \arg\max_{i} \; i \quad \text{s.t.} \sum_{j=1}^{i} \frac{\varphi_{j+1} + \varphi_j}{2} \leq \frac{\underline{\alpha}}{100\Delta\bar{p}}, \tag{4}$$

and then

$$p_{\underline{\alpha}} = \bar{p}_{i_{\underline{\alpha}}+1} \in \mathcal{P}_s. \tag{5}$$

The procedure to obtain the empirical distribution and price intervals can be outlined as follows. First, it is assumed that some values for γ and c denoted as γ^* and c^* have been chosen previously. Also, the value of the current market state $z(t)$ and the desired percentiles $\underline{\alpha}$ and $\overline{\alpha}$ are known. The procedure starts by computing the value of the dissimilarity function for the given γ^* and $z(t)$ for all the possible values of $p(t+k)$ (i.e., $\forall \bar{p}_i \in \mathcal{P}_s$). These values of J_{γ^*} are then used to compute the empirical probability density function for the given γ^*, c^* and $z(t)$ for all the possible values of $p(t+k)$. Using these computations it is possible to build the aforementioned empirical distribution of $p(t+k)$, and then find the desired percentiles to build the price interval, and also the median to be used as the price forecast. These steps are formally described in algorithm 2.

Algorithm 2 k-Step Probabilistic Price Interval Forecasting

Input: DB_k, $\mathcal{P}_s$, γ^*, c^*,$z(t)$, $\underline{\alpha}$ and $\overline{\alpha}$.
Output: $\hat{p}(t+k)$ and the price interval $[p_{\underline{\alpha}}, p_{\overline{\alpha}}]$.
1: Build $\Omega(z(t))$ as in Algorithm 1.
2: Compute the dissimilarity function of definition 1 $J_{\gamma^*}(z(t), \bar{p}_i, \Omega(z(t)))$ for all $\bar{p}_i \in \mathcal{P}_s$.
3: Using the previously computed values of the dissimilarity function, build the empirical distribution by computing $\text{ecp}_{\gamma^*,c^*}(z(t), \bar{p}_i, \Omega(z(t)))$ for all $\bar{p}_i \in \mathcal{P}_s$ using the approximation given in (1).
4: Find the desired upper percentile $p_{\overline{\alpha}} \in \mathcal{P}_s$ using the approximations given in (2) and (3) with $z(t)$, $\Omega(z(t))$,γ^* and c^*. In the same way, find the desired lower percentile $p_{\underline{\alpha}} \in \mathcal{P}_s$ using (4) and (5). Finally, using both methods with $\overline{\alpha} = \underline{\alpha} = 50$ obtain $p_{\overline{50}}$ and $p_{\underline{50}}$ and compute the median as $p_{50} = \frac{p_{\overline{50}}+p_{\underline{50}}}{2}$.
5: Return the desired interval $[p_{\underline{\alpha}}, p_{\overline{\alpha}}]$ and the price forecast $\hat{p}(t+k) = p_{50}$.

Note that the database can be updated as new market data is available. This, together with the use of local data, makes this strategy adaptive as in section II.

There are different ways of choosing the values γ^* and c^*. A possibility could be to implement some form of local search that would find the values of c and γ that minimize the prediction error in a validation set or even maximize the revenue when using the forecast and price intervals in a trading strategy. However, these strategies would not give the desired probabilistic guarantees on the computed price intervals. Thus here it is proposed to use a maximum likelihood estimation procedure presented by the authors in [36] and modified to use local data.

The algorithm needs the sets of possible values of γ and c, denoted as Γ and C. The sets can be chosen as sets of N_γ and N_c numbers from a grid in the intervals $[\gamma_{min}, \gamma_{max}]$ and $[c_{min}, c_{max}]$ where the extreme points of these intervals can be chosen directly as tuning parameters (e.g, they could be chosen using cross-validation with a test set). On the other hand, N_γ and N_c should be set in relation to the computing power available.

The procedure starts by computing the dissimilarity function for all the possible combinations of values of Γ and $\mathcal{P}_s$ and for every entry in the database using local data. Then, with these values of the dissimilarity function, the ecp is used to compute the empirical distribution for all the combinations of γ, c and market states in DB_k. After this, the desired percentiles are computed for each of the previously built distributions. Then, for every combination of $\gamma \in \Gamma$ and $c \in C$, the number of prices in DB_k that fall outside the quantiles of its distribution (that is, the number of quantile violations) are computed. These numbers are used to associate to every $\gamma \in \Gamma$ the greatest $c_\gamma \in C$ for which the percentage of violations of both lower and upper quantile is less than $\underline{\alpha}$ and greater than $\overline{\alpha}$ respectively. Finally, the optimal γ^*, c^* are chosen as the one combination among all the previously computed (γ, c_γ) that maximizes the likelihood ratio. Algorithm 3 describes formally this procedure.

There are other ways to compute the optimal γ^* and c^* with probabilistic guarantees, such as algorithm 2 in [36] in which the interval length is penalized aiming to smaller intervals.

Finally, note that although algorithms 2 and 3 have a higher computational burden than algorithm 1, both are highly parallelizable as many of the operations performed on every combination of data and parameters are independent of each other. In this way, large data sets, which are readily available by the stock market data providers, can be used. Furthermore, the computation of the optimal γ^* and c^* does not have to be repeated if the database is updated until the amount of updated data becomes a significant fraction of the database.

IV. CASE STUDY: FORECASTING THE DOW JONES INDUSTRIAL AVERAGE INDEX

The proposed approaches have been used in the problem of predicting the daily closing prices and price intervals for the Dow Jones Industrial Average index. The dataset was obtained from the data provider Bloomberg and is composed of the daily closing price of the Dow Jones Index from 2005 to mid-2016. The data was divided into a database DB, from 2005 to 2014, and a testing dataset, from 2015 to mid-2016. This latter period has been chosen because there is not a clear market trend (bullish or bearish) that would make forecasting easier. To lower the noise, all the raw prices in the database have been smoothed using a 5-day Exponential

Algorithm 3 Computation of γ^* and c^*

Input: $DB_k, \mathcal{P}_s, \Gamma, C, \underline{\alpha}, \overline{\alpha}$.

Output: γ^*, c^*.

1: For all the possible combinations of $\gamma \in \Gamma$, $z_l \in DB_k$ and $\bar{p}_i \in \mathcal{P}_s$ compute the dissimilarity function $J_\gamma(z_l, \bar{p}_i, \Omega(z_l))$.

2: For all possible combinations of $\gamma \in \Gamma$, $c \in C$ and $z_l \in DB_k$ build its associated empirical distribution computing $\text{ecp}_{\gamma,c}(z_l, \bar{p}_i, \Omega(z_l))$ for all $\bar{p}_i \in \mathcal{P}_s$ using the approximation given in (1).

3: For all $\gamma \in \Gamma$, $c \in C$, $z_l \in DB_k$ and their associated empirical distribution find the desired percentiles as in algorithm 2, i.e, for every combination find $p_{\overline{\alpha}} \in \mathcal{P}_s$ using (2) and (3) and $p_{\underline{\alpha}} \in \mathcal{P}_s$ using (4) and (5). Save the $p_{\underline{\alpha}}$ values in a vector denoted as $\underline{\phi}_{\gamma,c} \in \Re^{N_{DB}}$ and the values $p_{\overline{\alpha}}$ in a vector denoted $\overline{\phi}_{\gamma,c} \in \Re^{N_{DB}}$.

4: For every γ, c compute the number of prices $p_l \in DB_k$ falling outside the interval defined by $[\underline{\phi}_{\gamma,c}(l), \overline{\phi}_{\gamma,c}(l)]$. Denote such numbers as $\underline{v}_{\gamma,c}$ and $\overline{v}_{\gamma,c}$.

5: For each $\gamma \in \Gamma$, select the greatest $c_\gamma \in C$ for which

$$\frac{100\underline{v}_{\gamma,c_\gamma}}{N_{DB}} \leq \underline{\alpha} \text{ and } \frac{100\overline{v}_{\gamma,c_\gamma}}{N_{DB}} \geq \overline{\alpha}.$$

6: Compute

$$\gamma^* = \arg\max_{\gamma} \sum_{l=1}^{N_{DB}} \log\left(\text{ecp}_{\gamma,c_\gamma}(z_l, p_l, \Omega(z_l))\right).$$

using for every γ considered the c_γ selected in the previous step. The optimal value of c^* is the c_γ selected in step 3 for γ^*.

Moving Average (EMA), which can be computed as:

$$p^d_{EMA}(t) = \frac{2}{d+1}p(t) + (1 - \frac{2}{d+1})p^d_{EMA}(t-1),$$

with $EMA_D(0) = p(0)$ and being $d = 5$ in this case. Note that the smoothing applied here is very light as the usual values of d for short-term forecasting are the 12 and 26 day EMA [44]. This would preserve fast price fluctuations although makes forecasting more difficult.

The market state $z(t)$ has been chosen to be composed of the last ten days prices smoothed using the 5-day EMA approach, as well as the 5-day and 10-day relative difference percentage of unsmoothed prices (RDP) [45] i.e.,

$$RDP_d(t) = 100\frac{p(t) - p(t-d)}{p(t)},$$

being d equal to 5 and 10 respectively.

The approach of section II has been applied to the case study to forecast the closing prices for up to 5 days, that is a full week of market sessions. The size of $\Omega(z(t))$ was $N = 250$ and $\gamma = 0$. The forecast and real prices are seen in figures 1 and 2. It can be seen that the forecast is quite accurate for the first sessions and, as expected, it becomes worse as k grows.

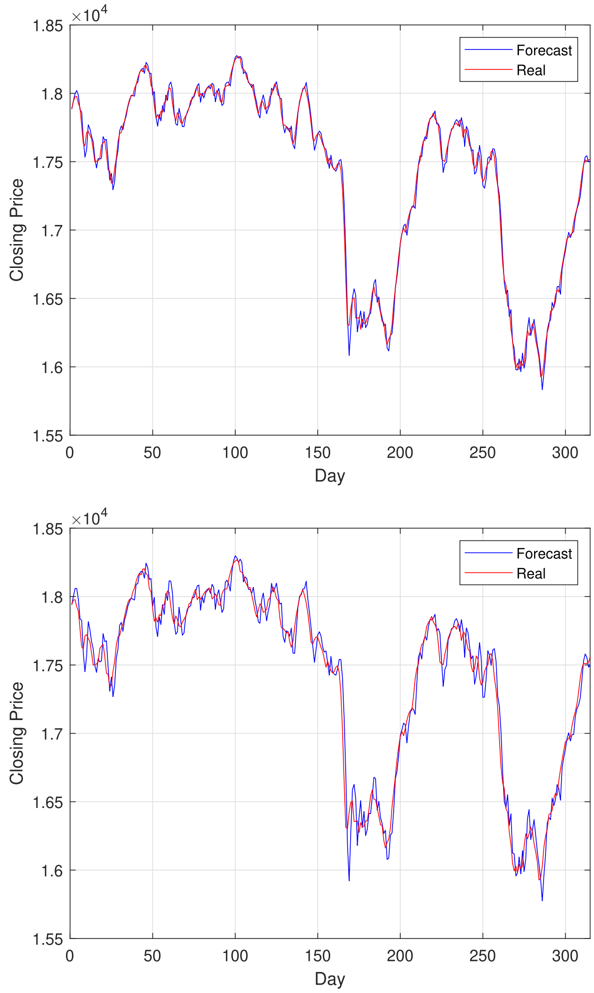

FIGURE 1. Forecasted and real prices (5-day EMA) for 1 day and 2 days forecasting.

The approach for price interval forecasting has been also applied to the case study. The parameters have been $N = 250$, $N_p = 1000$, $p_{min} = 6684.3$, $p_{max} = 19445$, $N_\gamma = 10$, $\gamma_{min} = 0$, $\gamma_{max} = 5$, $N_c = 60$, $c_{min} = 0.25$, $c_{max} = 15$. The 10-th and 90-th percentiles were chosen for the price intervals, thus the probabilistic specification is that the intervals contain the real price is 0.8. The results obtained are shown in figures 3 and 4 in which the price intervals are represented as envelopes. It can be seen that although the price intervals are quite tight for $k = 1$, they grow as k rises. This is congruent with the fact that for farther prediction horizons the uncertainty on the forecasting is greater. Note also that sometimes the real price is not inside the computed price interval. This is also congruent to the fact that it should fall outside of the interval about 20% of the times.

Even if the results obtained seem correct at a glance, it is necessary some form of validation. Thus, the results have been validated in relation to a persistence predictor, i.e., martingale, that has been used as a baseline approach forecasting the prices as $\hat{p}(t+k) = p(t)$. Furthermore, a multilayer perceptron (MLP) with 20 neurons in the hidden layer

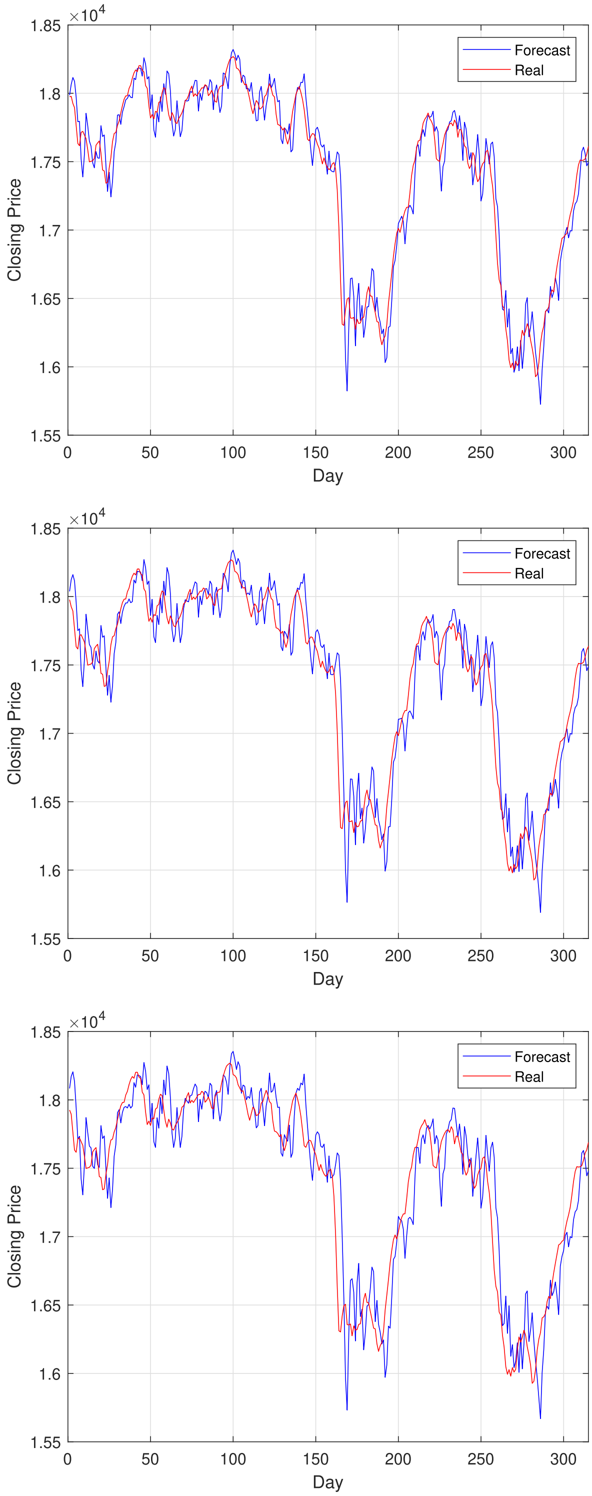

FIGURE 2. Forecasted and real prices (5-day EMA) for 3 to 5 days forecasting.

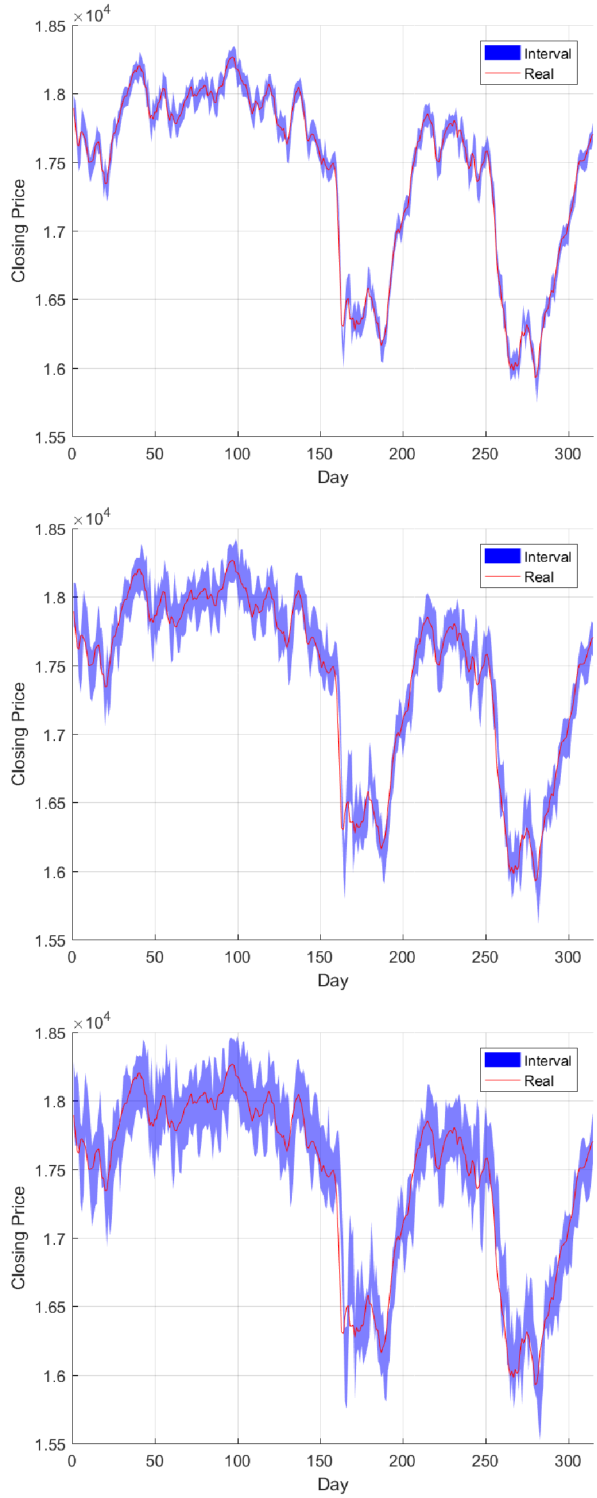

FIGURE 3. Price intervals for 1 to 3 days (5-day EMA).

and trained with the Levenberg-Marquardt rule has been also used as a baseline approach. Table 2 shows the mean squared errors (MSE) for proposed and baseline approaches. It can be seen that the approach proposed in section II presents the lower MSE of all approaches and that the forecast using the strategy of section III is the second best for k up to 4. Another parameter to be studied is the dispersion of errors. Table 3 shows the standard deviations of the errors for all approaches. It can be seen that, as in the case of the MSE, the approach of section II has the tighter errors in all cases and

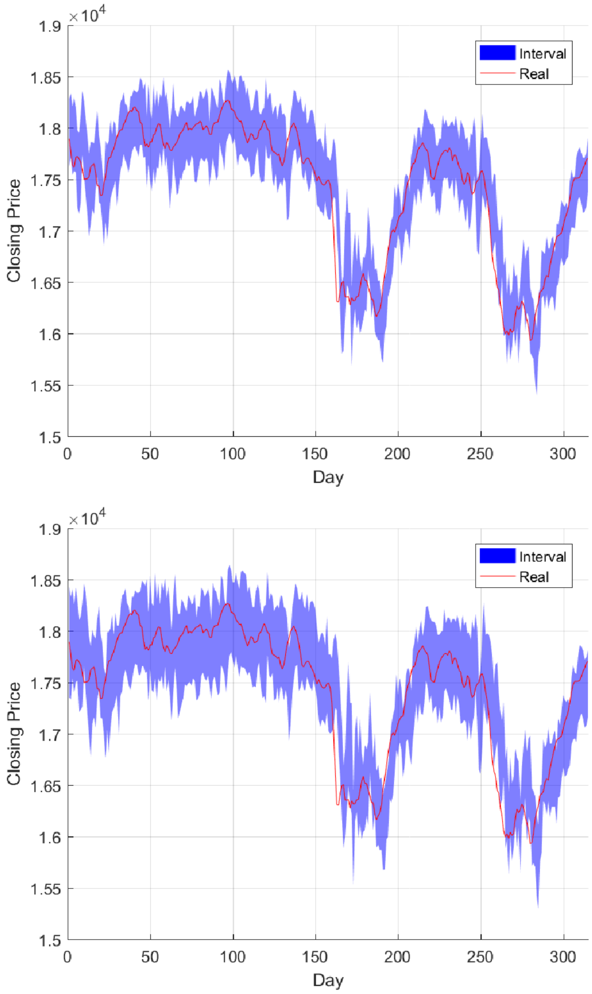

FIGURE 4. Price intervals for 4 and 5 days (5-day EMA).

TABLE 2. MSE obtained using the proposed and baseline approaches (5-day EMA).

k	Proposed Section II	Proposed Section III	MLP	Persistence
1	3,294.8	3,520.5	3,577.0	5,296.4
2	12,433.2	13,607.3	14,190.8	16,867.4
3	26,659.0	28,574.7	30,829.6	31,975.0
4	45,475.9	46,931.0	50,794.9	49,072.4
5	66,859.6	71,300.9	77,017.5	66,943.4

TABLE 3. Standard deviation of the errors (σ); 5-day EMA.

k	Proposed Section II	Proposed Section III	MLP	Persistence
1	57.4	59.4	59.9	72.8
2	111.5	116.4	119.3	130.0
3	163.1	168.2	175.6	178.9
4	212.9	215.7	224.9	221.6
5	257.8	266.4	276.0	258.8

TABLE 4. MSE obtained using the proposed and baseline approaches (15-day EMA).

k	Proposed Section II	Proposed Section III	MLP	Persistence
1	469.2	486.8	606.8	1,531.7
2	2,129.4	2,313.2	2,492.3	5,642.4
3	5,386.7	5,907.5	7,538.6	11,944.6
4	10,596.6	11,554.2	13,629.2	20,151.2
5	17,745.7	19,856.5	24,768.5	29,953.1

TABLE 5. Standard deviation of the errors (σ); 15-day EMA.

k	Proposed Section II	Proposed Section III	MLP	Persistence
1	21.6	22.1	24.6	42.1
2	45.9	47.9	49.9	78.7
3	72.8	76.0	85.6	113.2
4	101.8	105.9	114.2	145.9
5	131.4	138.5	152.8	177.1

that the forecasting using the median of the price distribution is the second best for k up to 4. Thus, the errors are expected to be smaller with the proposed strategies and more close to their mean values. This also results in lower uncertainty on the quality of the prediction. Furthermore, the results show that the approach of section II is complementary to that of section III producing better forecasts with a much lower computational burden. In fact, when implemented in Matlab on an Intel Core i7-4790 CPU computer, the computation time for the strategy of section II was 0.0037 seconds. On the other hand, the implementation of Algorithm 2 took 0.3997 seconds on the same computer, whereas the implementation of algorithm 3 required 4.5 hours to find γ^* and c^*. Note, however, that γ^* and c^* are computed only once provided that the database does not suffer major changes.

Although the MSE and standard deviation is better for the approach of section II, it is practically equal to the MSE of the persistence predictor for $k = 5$. The reason for this is that as the prediction horizon k grows, the price time series becomes more similar to a random walk, making persistence predictors a good choice for price forecasting. This is more evident when the smoothing of the prices is quite light, like in the previous simulations. More typical periods in the EMA smoothing (12 and 26 days are common in stock trading for short term forecasting) show how the proposed approaches make a better job predicting the price trend than persistence predictors. Tables 4 and 5 show the MSE and standard deviations for all the approaches when using a 15-day EMA smoothing. These tables show that in this case the proposed approaches have always lower MSE and tighter errors. Furthermore, the MLP is also better than the persistence for all k, whereas in the case of the lighter smoothing it was worse for k equal to 4 and 5.

The proposed strategies can also be used to forecast intraday stock prices. The Dow Jones Industrial Average prices from 06/03/2020 to 11/03/2020 have been considered as an example. Tables 6 and 7 show the MSE and standard deviation values when forecasting half-hourly prices from this period. In these tests the longest forecasting horizon was 3.5 hours, thus k varies from 1 to 7. A 2.5-hour EMA has been used to smooth the prices and the structure of market state $z(t)$ is the same as before but changing daily prices and RDP for their half-hourly counterparts. It can be seen that the strategy of section II performs as expected and that, in this case, there is no need for further smoothing to keep the performance better than the persistence predictor. On the other hand, the

TABLE 6. MSE obtained using the proposed and baseline approaches with intraday half-hourly prices (2.5-hour EMA).

k	Proposed Section II	Proposed Section III	MLP	Persistence
1	1,034.9	1,153.1	1,037.7	2,103.9
2	4,406.9	4,816.6	4,498.1	7,335.2
3	10,233.0	11,527.9	10,589.7	14,584.5
4	17,423.8	19,859.2	18,599.3	22,933.7
5	25,453.4	29,444.1	27,880.3	31,796.9
6	34,002.5	39,619.5	37,676.9	40,947.9
7	43,086.0	49,990.4	49,331.7	50,124.3

TABLE 7. Standard deviation of the errors (σ); 2.5-hour EMA.

k	Proposed Section II	Proposed Section III	MLP	Persistence
1	32.2	33.5	32.1	45.9
2	66.4	68.2	66.9	85.7
3	101.2	104.6	102.6	120.9
4	132.0	136.8	135.9	151.6
5	159.5	165.8	166.4	178.5
6	184.3	191.6	193.3	202.5
7	207.4	215.4	221.2	224.1

TABLE 8. Empirical probability of the real price to be contained in the computed intervals (the theoretical one is 0.8) and average width of the price intervals using the proposed approach and quantile regression.

	Empirical probability		Average interval width	
k	Proposed	Q. regression	Proposed	Q. regression
1	0.8679	0.6006	162.4802	99.8091
2	0.8459	0.6101	315.7218	190.5443
3	0.8553	0.6321	477.8004	287.5774
4	0.8648	0.6761	638.8747	397.3780
5	0.8805	0.6950	813.5661	489.7816

strategy of section III has a higher MSE than the MLP (but lower than the persistence predictor). Note, however, that the strategy of section III is used to produce interval forecasts rather than price forecasts.

On the other hand, the previous results and baseline approaches are not useful to validate the price interval forecasting obtained using the strategy of section III. In order to do so, the well-known quantile regression has been chosen as a baseline approach to validate the interval forecasting. Table 8 shows the empirical probabilities and interval width of the proposed strategy and quantile regression using the same data and theoretical probability (0.8). It can be seen that the intervals computed using the proposed approach contains the real price with a higher probability than the specified one, whereas the quantile regression produces tighter intervals that do not meet the specified probability for any k. Thus, the quantile regression fails in this case. However, in the case of the intraday dataset both strategies work well (see table 9), being the proposed strategy a bit more conservative. Note that some form of tightening could be used with the proposed approach to make the intervals narrower while meeting the probability in practice, but the resulting intervals will not have the probabilistic guarantee of the approach.

Finally, the growing values of the forecasting error and its dispersion together with the effect of smoothing the prices suggest, that although not a perfect random walk for very

TABLE 9. Empirical probability and average width of the price intervals using the proposed approach and quantile regression (intraday dataset).

	Empirical probability		Average interval width	
k	Proposed	Q. regression	Proposed	Q. regression
1	0.8820	0.8614	88.5662	74.3958
2	0.8846	0.8373	178.6064	159.0279
3	0.8635	0.8279	275.3878	238.7693
4	0.8571	0.8214	378.3202	313.8665
5	0.8358	0.8119	443.9647	389.4412
6	0.8293	0.8084	516.9591	457.2965
7	0.8348	0.8138	576.9186	516.4660

short term forecasting, as the prediction horizon grows, the price time series becomes more difficult to forecast, gradually approaching to a random walk.

V. CONCLUSION

Two strategies have been proposed to be used in the open problems of stock price and price interval forecasting. The first is related to direct weight optimization techniques and obtains the forecast by using local data close to the current market state. The computational burden is quite low and does not require a training phase, except for the tuning of γ. Moreover, its results when applied to a well known case study have been validated in relation to two well-known techniques. The second approach computes the price intervals using a probabilistic approach in which the empirical conditional probability density function of the forecasted price is computed using local data. The algorithm for doing this is easily parallelizable making its computational burden manageable. This approach has been also validated and compared favourably to the well-known quantile regression approach. Both techniques have been proved to be useful for the investors in terms of accuracy, and have other advantages like adaptation to the current market situation. Thus, the proposed techniques have proved that they can be added to the toolbox of stock market traders.

Among the open questions that could be considered as future work, one would be the study of which technical indicators work best as market state, i.e., a feature selection study tailored for the proposed approaches. In this regard, the number of possible technical indicators is quite high, and clearly, some of them are redundant. From the results of [40] it can be inferred that among the most promising technical indicators would be moving averages, either simple or exponential, the daily return on capital, the traded volume and the proprietary JKHL index. Nevertheless, the suitability of these technical indicators with the proposed techniques is something that needs to be studied in future works. Finally, the proposed techniques could be used to devise a trading strategy based on forecasting prices and intervals that could be validated against well-known trading strategies like buy-and-hold or those based on time-weighted average price (TWAP) or volume-weighted average price (VWAP).

REFERENCES

[1] E. F. Fama, "Random walks in stock market prices," *Financial Analysts J.*, vol. 51, no. 1, pp. 75–80, Jan. 1995.

[2] C. M. C. Lee, J. Myers, and B. Swaminathan, "What is the intrinsic value of the dow?" *J. Finance*, vol. 54, no. 5, pp. 1693–1741, Oct. 1999.

[3] P.-F. B. Lai and W. K. Wong, "An empirical study of relationship between share price and intrinsic value of company," *Financial Stud.*, vol. 19, no. 4, pp. 1–28, 2015.

[4] R. Tiwari, "Intrinsic value estimates and its accuracy: Evidence from Indian manufacturing industry," *Future Bus. J.*, vol. 2, no. 2, pp. 138–151, Dec. 2016.

[5] R. E. Shrieves and J. M. Wachowicz, "Free cash flow (FCF), economic value added (EVA^{TM}), and net present value (NPV): A reconciliation of variations of discounted-cash-flow (DCF) valuation," *Eng. Economist*, vol. 46, no. 1, pp. 33–52, Jan. 2001.

[6] R. Lundholm and T. O'keefe, "Reconciling value estimates from the discounted cash flow model and the residual income model," *Contemp. Accounting Res.*, vol. 18, no. 2, pp. 311–335, 2001.

[7] T. Sougiannis and T. Yaekura, "The accuracy and bias of equity values inferred from Analysts' earnings forecasts," *J. Accounting, Auditing Finance*, vol. 16, no. 4, pp. 331–362, Oct. 2001.

[8] R. A. DeFusco, D. W. McLeavey, J. E. Pinto, and D. E. Runkle, *Quantitative Investment Analysis Workbook*. Hoboken, NJ, USA: Wiley, 2015.

[9] G. S. Atsalakis and K. P. Valavanis, "Forecasting stock market short-term trends using a neuro-fuzzy based methodology," *Expert Syst. Appl.*, vol. 36, no. 7, pp. 10696–10707, Sep. 2009.

[10] H. Ince and T. B. Trafalis, "Short term forecasting with support vector machines and application to stock price prediction," *Int. J. Gen. Syst.*, vol. 37, no. 6, pp. 677–687, Dec. 2008.

[11] E. F. Fama, "The behavior of stock-market prices," *J. Bus.*, vol. 38, no. 1, pp. 34–105, 1965.

[12] R. C. Shiller, "Irrational exuberance," *Philosophy Public Policy Quart.*, vol. 20, no. 1, pp. 18–23, 2000.

[13] M. F. M. Osborne, "Brownian motion in the stock market," *Oper. Res.*, vol. 7, no. 2, pp. 145–173, Apr. 1959.

[14] J.-P. Danthine, "Martingale, market efficiency and commodity prices," *Eur. Econ. Rev.*, vol. 10, no. 1, pp. 1–17, Jan. 1977.

[15] W. A. Barnett and A. Serletis, "Martingales, nonlinearity, and chaos," *J. Econ. Dyn. Control*, vol. 24, nos. 5–7, pp. 703–724, Jun. 2000.

[16] C. Zheng and J. Zhu, "Research on stock price forecast based on gray relational analysis and ARMAX model," in *Proc. Int. Conf. Grey Syst. Intell. Services (GSIS)*, Aug. 2017, pp. 145–148.

[17] Y. Lin, H. Guo, and J. Hu, "An SVM-based approach for stock market trend prediction," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Aug. 2013, pp. 1–7.

[18] S. Mahfoud and G. Mani, "Financial forecasting using genetic algorithms," *Appl. Artif. Intell.*, vol. 10, no. 6, pp. 543–566, Dec. 1996.

[19] N. Baba and M. Kozaki, "An intelligent forecasting system of stock price using neural networks," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, vol. 1, 1992, pp. 371–377.

[20] H. White, "Economic prediction using neural networks: The case of IBM daily stock returns," in *Proc. ICNN*, vol. 2, 1988, pp. 451–458.

[21] E. Guresen, G. Kayakutlu, and T. U. Daim, "Using artificial neural network models in stock market index prediction," *Expert Syst. Appl.*, vol. 38, no. 8, pp. 10389–10397, Aug. 2011.

[22] J. Cao and J. Wang, "Stock price forecasting model based on modified convolution neural network and financial time series analysis," *Int. J. Commun. Syst.*, vol. 32, no. 12, p. e3987, Aug. 2019.

[23] P. Yu and X. Yan, "Stock price prediction based on deep neural networks," *Neural Comput. Appl.*, vol. 32, no. 6, pp. 1609–1628, Mar. 2020.

[24] Q. Chen, W. Zhang, and 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*, vol. 8, pp. 117365–117376, 2020.

[25] O. B. Sezer, M. U. Gudelek, and A. M. Ozbayoglu, "Financial time series forecasting with deep learning: A systematic literature review: 2005–2019," *Appl. Soft Comput.*, vol. 90, May 2020, Art. no. 106181.

[26] P. S. Rao, K. Srinivas, and A. K. Mohan, "A survey on stock market prediction using machine learning techniques," in *Proc. ICDSMLA*. Singapore: Springer, 2020, pp. 923–931.

[27] J. R. Salvador, D. Muñoz de la Peña, D. R. Ramirez, and T. Alamo, "Predictive control of a water distribution system based on process historian data," *Optim. Control Appl. Methods*, vol. 41, no. 2, pp. 571–586, Mar. 2020.

[28] J. R. Salvador, D. M. de la Peña, T. Alamo, and A. Bemporad, "Data-based predictive control via direct weight optimization," *IFAC-PapersOnLine*, vol. 51, no. 20, pp. 356–361, 2018.

[29] J. M. Bravo, T. Alamo, M. Vasallo, and M. E. Gegundez, "A general framework for predictors based on bounding techniques and local approximation," *IEEE Trans. Autom. Control*, vol. 62, no. 7, pp. 3430–3435, Jul. 2017.

[30] J. Roll, A. Nazin, and L. Ljung, "Nonlinear system identification via direct weight optimization," *Automatica*, vol. 41, no. 3, pp. 475–490, Mar. 2005.

[31] W. Jian-hong, "Improvement on direct weight optimization identification," *J. Control Syst. Eng.*, vol. 3, no. 1, pp. 1–9, Mar. 2015.

[32] J. Roll, A. Nazin, and L. Ljung, "A general direct weight optimization framework for nonlinear system identification," *IFAC Proc. Volumes*, vol. 38, no. 1, pp. 178–183, 2005.

[33] Q. He, S. Tan, and D. Wanshan, "DWO based predictive control for nonlinear systems," in *Proc. Int. Conf. Measuring Technol. Mechatronics Autom.*, vol. 2, Mar. 2010, pp. 38–41.

[34] E.-W. Bai and Y. Liu, "Recursive direct weight optimization in nonlinear system identification: A minimal probability approach," *IEEE Trans. Autom. Control*, vol. 52, no. 7, pp. 1218–1231, Jul. 2007.

[35] Q. Wu and T. Van Voorhis, "Direct optimization method to study constrained systems within density-functional theory," *Phys. Rev. A, Gen. Phys.*, vol. 72, no. 2, Aug. 2005, Art. no. 024502.

[36] A. Daniel Carnerero, D. R. Ramirez, and T. Alamo, "Probabilistic interval predictor based on dissimilarity functions," Oct. 2020, *arXiv:2010.15530*. [Online]. Available: http://arxiv.org/abs/2010.15530

[37] J. Berkowitz, "Testing density forecasts, with applications to risk management," *J. Bus. Econ. Statist.*, vol. 19, no. 4, pp. 465–474, Oct. 2001.

[38] R. Koenker and K. Hallock, "Quantile regression," *J. Econ. Perspect.*, vol. 15, no. 4, pp. 143–156, 2001.

[39] D. Pradeepkumar and V. Ravi, "Forecasting financial time series volatility using particle swarm optimization trained quantile regression neural network," *Appl. Soft Comput.*, vol. 58, pp. 35–52, Sep. 2017.

[40] G. Alfonso and D. R. Ramirez, "A nonlinear technical indicator selection approach for stock Markets. Application to the Chinese stock market," *Mathematics*, vol. 8, no. 8, p. 1301, Aug. 2020.

[41] J. M. Manzano, J. M. Nadales, D. M. de la Pena, and D. Limon, "Oracle-based economic predictive control," in *Proc. IEEE 58th Conf. Decis. Control (CDC)*, Dec. 2019, pp. 4246–4251.

[42] M. R. Anderberg, *Cluster Analysis for Applications*. New York, NY, USA: Academic, 1973.

[43] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc. B, Methodol.*, vol. 58, no. 1, pp. 267–288, Jan. 1996.

[44] V. P. Upadhyay, S. Panwar, R. Merugu, and R. Panchariya, "Forecasting stock market movements using various kernel functions in support vector machine," in *Proc. Int. Conf. Adv. Inf. Commun. Technol. Comput. (AICTC)*. New York, NY, USA: Association for Computing Machinery, 2016.

[45] M. Thomason, "The practitioner methods and tool," *J. Comput. Intell. Finance*, vol. 7, no. 3, pp. 36–45, 1999.

GERARDO ALFONSO received the B.Eng. degree in electrical and electronics engineering from the University of Leicester, U.K., the M.B.A. degree from the University of Cambridge, U.K., and the Ph.D. degree in finance from the University Complutense de Madrid, Spain. He was a Postdoctoral Researcher with the University of Cambridge. He has more than a decade of experience in trading in capital markets in USA, Europe, and Asia. He is currently an Executive Director with Shenwan Hongyuan Securities, Shanghai.

A. DANIEL CARNERERO received the M.Eng. degree in industrial engineering from the University of Seville, in 2019, where he is currently pursuing the Ph.D. degree with the Department of Systems Engineering and Automation. His current research interests include GPU computing, metaheuristic optimization, model predictive control, randomized algorithms, and data-driven methods.

DANIEL R. RAMIREZ was born in Spain, in 1972. He received the M.Eng. and Ph.D. degrees in computer engineering from the University of Seville, Spain, in 1996 and 2002, respectively. From 1997 to 1999, he was a Research Assistant with the Department of System Engineering and Automatic Control. Since 1999, he has been an Assistant Professor and an Associate Professor with the Department of System Engineering and Automatic Control, University of Seville. He has authored or coauthored more than 70 technical articles in international journals and conference proceedings. His current research interests include randomized algorithms, model predictive control, data-based forecasting, and soft computing techniques.

TEODORO ALAMO was born in Spain, in 1968. He received the M.Eng. degree in telecommunications engineering from the Polytechnic University of Madrid, Madrid, Spain, in 1993, and the Ph.D. degree in telecommunications engineering from the University of Seville, Seville, Spain, in 1998. From September 1991 to May 1993, he was with the Ecole Nationale Superieure des Télécommunications (Télécom Paris), Paris, France. From 1993 to 2000, he was an Assistant Professor with the Department of System Engineering and Automatic Control, University of Seville, where he was an Associate Professor, from 2001 to 2010, where he has been a Full Professor, since March 2010. Part of his Ph.D. work was done with RWTH Aachen, Aachen, Germany, from June to September 1995. He has co-founded the spin-off company Optimal Performance (University of Seville). He is the author or coauthor of more than 200 publications, including books, book chapters, journal articles, conference proceedings, and educational books. His current research interests include decision-making, model predictive control, data-driven methods, randomized algorithms, and optimization strategies.