# Machine Learning-based Time Series Prediction at Brazilian Stocks Exchange

Ana Paula dos Santos Gularte[1,2*], Danusio Gadelha Guimarães Filho[1,2], Gabriel de Oliveira Torres[1], Thiago Carvalho Nunes da Silva[1] and Vitor Venceslau Curtis[1,2*]

[1]Department of Aerospace Science and Technology, Aeronautics Institute of Technology (ITA), Praça Marechal Eduardo Gomes, São José dos Campos, 12.228-900, São Paulo, Brazil.
[2]Department of Science and Technology, Federal University of São Paulo (UNIFESP), Avenida Cesare Mansueto Giulio Lattes, São José dos Campos, 12.247-014, São Paulo, Brazil.

*Corresponding author(s). E-mail(s): gularte@ita.br; curtis@ita.br;
Contributing authors: danusio.gadelhafilho@gmail.com; torres@ita.br; thg.cns@gmail.com;

**Abstract**

Machine learning (ML) algorithms stand out for their ability to deal with nonlinear, dynamic, and chaotic problems, characteristics in stock market price series. This paper uses the daily returns series of 10 companies listed in the Ibovespa index from 2016 to 2020 to predict returns 180 days ahead. We implement an attribute selection approach, which achieves better performance in training time, prediction accuracy, and improved ability to select a subset of variables and model ensemble for improving overall predictor efficiency. The results suggest the potential for predicting stock returns based on the most important variables selected by the algorithm. Furthermore, the evaluation metrics (Coefficient of determination $\boldsymbol{R^2}$, ***Willmott index***, Mean absolute error $\boldsymbol{MAE}$, and ***Kurtosis***) for measuring model stability were calculated outside the training sample and showed average values of $\boldsymbol{R^2}$ of 95%, ***Willmott's index*** indicated a better fit between predicted and realized values in the ML model, with a

value very close to 1. The prediction errors measured by ***MAE*** indicate 3% for the ML model, while the ARIMA model presented 32%. Finally, ***Pearson's kurtosis coefficient*** showed a value of 12, achieving superior prediction results compared to the traditional Autoregressive Integrated Moving Averages (ARIMA) model in all metrics.

**Keywords:** Machine Learning, Financial Time Series, Prediction, Hybrid Intelligent Algorithm, Ensemble

# 1 Introduction

Comparisons of forecasts and their accuracy have long played an important and essential role in evaluating economic models and financial decision-making [1], [2]. In the empirical demonstration [3] in 1983, Messe and Rogoff compare historical exchange rate series and evaluate the accuracy of out-of-sample prediction models. They prove that structural interest rate models predicted nothing better than a random walk. Their findings stimulated a huge literature and made prediction evaluation a common element in reduced-form and structural exchange rate models [4]. Several recent examples are present in chapter 2 of the Handbook of Economic Forecasting [5]; the authors explore forecasting and evaluate models such as Dynamic Stochastic General Equilibrium (DSGE) that use modern macroeconomic theory to explain and forecast co-movements of aggregate time series over the business cycle. And macro-finance models study the relationship between asset prices and economic fluctuations [5].

In addition to the forecasting methodology, variables are an essential component in stock market forecasting and play a vital role in the forecasting process [6]. The variables related to this market have various forms, such as stock prices, log returns, volatility, interest rates, and lagged indices, to name a few examples, as a result of the historical time series data set [7]. There are two well-known analytical approaches in stock market forecasting studies: fundamental and technical. Charles Henry Dow developed the original Dow Theory for technical analysis in 1884, and a modern explanation in the literature by [8] emerges as the core of the concepts around effective technical trading to support investors in making smarter and more profitable decisions. We present these findings because the final result of the forecast does not depend only on machine learning algorithms, but is also influenced by the representation of the problem's input data. In a recent literature survey, authors [9] and [6] state that the use of technical indicators in the selection of variables included during the model-building process in stock market forecasting is still limited in the literature and has been increasing its application in recent research. The authors further reinforce that the results of identifying relevant variables can improve the forecast accuracy of models and reduce computational processing time.

Time series forecasting has traditionally used statistical-based methods, such as the Autoregressive Integrated Moving Average (ARIMA) model, which is a generalization of an Autoregressive Moving Average (ARMA) model proposed by [10] due to the models' properties in learning patterns from historical data and robustness in forecasting time series. Over time, new tools and developments supplement many insights that have improved these methodologies. For example, for model selection, [11] developed the information criterion denoted as Akaike Information Criterion (AIC), which aims to minimize the Kullback-Leibler (K-L) distance as a basis for model selection by discerning how "close" a fitted model is to the true generating model. Still, [12] describe innovations of state space models that underlie exponential smoothing methods. On the other hand, the continuous evolution in Machine Learning and Artificial Intelligence, and the fact that financial market information is increasingly accessible to many investors, contribute to the development of computational intelligence approaches to forecast time series. Despite the numerous publications on forecasting with computational intelligence methods, this field has not yet reached the degree of maturity of traditional statistical methodologies [13].

In [6], [14], and [7], the authors conducted a comprehensive review of the literature devoted to machine learning and dataset techniques for stock market forecasting in recent articles from the last two decades. We highlight three relevant aspects of their research. First, the publication of articles between 2015 and 2019 accounts for more than 50%, while the remaining articles are from 15 years before 2015. These findings indicate an exponential growth in the number of studies applied to the financial market in stock forecasting using machine learning approaches. Second, the authors present the occurrence of the keywords present in the articles listed at the top of the list forecasting, data mining, feature selection, and stock price prediction or forecasting. However, they comment that South America, with Brazil as the only representative, has only seven mentions covered in the literature and, according to the authors, seems to have a minor role in the stock market forecasting literature. And finally, the authors indicate that the number of research on the variables included during the model-building process for stock market forecasting is limited; they reinforce that while there are a variety of recent techniques in the literature that apply variable selection, there is no consistent method that selects the relevant variables in forecasting stock returns.

The Brazilian market represents an interesting case study because besides being on the list of the ten largest economies in the world, it is one of the most important emerging economies and the main one in Latin America [15]. It is also a market with strong information asymmetry, presents an ambiguous nature in the movement of stock prices, and makes investments intrinsically risky, making it difficult for investors and the government to forecast its trends accurately.

The stock price series is generally dynamic, nonlinear, and non-parametric [16]; on the one hand, the poor performance of statistical time series models is

potentially greater due to the assumptions of their structure. Also, as alluded to earlier, although computational intelligence approaches are prominent and have been gaining more and more space in the financial market due to the ability to find complex patterns in large volumes of data [17], [18], ensemble methods that use various learning techniques are recent innovations. These methods present a gap for developments applied to datasets targeted at real-world problems, such as stock market forecasting [19], [20], and are proven to outperform single models in forecasting financial time series [21].

This paper uses data mining techniques to select the important variables for the prediction model. The general idea behind variable importance analysis is to calculate measures to quantify each variable's importance in the dataset concerning a particular class or concept description. Such measures include One Rule (OneR), Information Gain (IG), and Chi-squared. We also introduce a new ensemble time series method to predict the direction of the return values of ten stocks 180 days ahead, employing the Gradient Boosting Machine (GBM), K-Nearest Neighbor (KNN), and Bayesian Regularized Neural Networks (BRNN) algorithms, in addition to the traditional ARIMA model. We apply the proposed model to the data set extracted from the Ibovespa index, the most important indicator of the average performance of Brazilian stock prices traded on the B3 (acronym for Brasil, Bolsa, Balcão), formed by a hypothetical portfolio of the stocks with the highest trading volume, composed of 93 stocks from 90 companies [22]. Finally, we compare the performance of the ensemble and ARIMA in-sample and out-of-sample models.

Thus, this work aims to extend the current knowledge in stock market forecasting through a new approach that combines different machine learning methods. We believe that many of the strategies used in the proposed methodology are general enough to be applied in other contexts of time series forecasting based on computational intelligence techniques.

The remainder of this paper is structured as follows: we review the literature on stock market prediction through machine learning from the perspective of the parametric and non-parametric methods selected for this study in Section 2. We describe our methodology in detail, i.e., data description, preprocessing, data partitioning for the sliding window, evaluation metrics, and architecture of the proposed model in Section 3. We present the results at each step developed, explain data mining for selecting the important variables used in the prediction model, and evaluate the in-sample and out-of-sample performance of the ensemble and ARIMA methods in Section 4. Finally, we present our conclusions and recommend future research in Section 5.

## 2 Related Work

Recent research [23], [24], [25] shows the time series widely studied in domains abounding in areas such as economics, industrial organizations, meteorology, sales projection, production planning, labor, environmental, public and health economics are some examples evidenced by [26], [27], [28] and [29]. While

there is still some controversy about the predictability of stock returns due to concerns about spurious regressions, the prevailing tone in the literature [30], [31], [32] and [33] is that stock returns have a predictable component.

Most previous studies have applied traditional time series methodologies based on historical data to predict stock prices and returns [34], [35]. These include Simple Linear Regression models (SLR), Multiple Linear Regression (MLR), Moving Average (MA) model, and the various extensions such as Autoregressive (AR), namelly Autoregressive MA (ARMA), Autoregressive integrated MA (ARIMA) Seasonal ARIMA (SARIMA), among others. Besides these techniques, others such as Markov Chains, Autoregressive Conditional Heteroskedasticity (ARCH), Generalized Autoregressive Conditional Heteroskedasticity (GARCH), Kalman filtering, and Exponential Smoothing complete the most popular techniques [26], [34], [36], [37], [23], [38], [39], [40], [41], [42], [43]. The last few decades have witnessed the widespread use of new approaches with the introduction of artificial intelligence (AI), and these techniques have received more attention in stock market forecasting studies; however, this field has not yet reached the degree of maturity of statistical methodologies [13]. Results show that the accuracy of these artificial intelligence methods is superior to traditional statistical methods, especially concerning nonlinear, chaotic, noisy, and complex stock market patterns. These methods have been extensively studied by [13], [44], [45], [20], [46], [47], [48] and more recently, [18], [49], which present some of the state-of-the-art machine learning-based techniques on hundreds of stocks and predictors.

Thus, the literature on stock market prediction through machine learning is vast, and here we briefly review state-of-the-art applied to the parametric and non-parametric methods selected for this study. We chose three different algorithms to optimize the predictive model: 1) Gradient Boosting Machine (GBM), 2) k-Nearest Neighbor (k-NN), 3) Bayesian Regularized Neural Networks (BRNN), as well as the ensemble of the models; compared with the conventional ARIMA model proposed by George Box & Gwilym Jenkins (1976)[10], widely used for time series prediction. Each algorithm selected in this study employs a different method to fit the data and approximate the regularities or correlations according to specific thresholds that bound the hypotheses to explain a wider range of the target variable. In addition, there is an improvement in the prediction accuracy of the techniques above by using filter-based attribute selection methods. This approach is discussed in the research of [50], which makes use of this method to reduce data complexity and improve prediction accuracy.

The non-parametric k-NN method is one of the main algorithms used by the machine learning community due to its simplicity, and good predictive accuracy on varied datasets [51], [13]. However, according to [13], there are few applications of k-NN regressions in time series forecasting in a general setting. The author develops an automated method for applying k-NN in time series forecasting effectively and efficiently, whose results indicate that the selection of input variables was a key factor in forecast accuracy, and desazonization

techniques were superfluous since k-NN was able to handle seasonal patterns. Whereas in the experiments of [48], despite the optimal performance of the k-NN method, the authors performed an increment in the technique that empirically combines decomposition to forecast the stock index. The results indicate higher accuracy in predicting financial time series with the application of Ensemble Empirical Mode Decomposition (EEMD) with Multidimensional k-NN model (EEMD-MKNN) than the Empirical Mode Decomposition (EMD) with k-NN (EMD-KNN), k-NN, and ARIMA models, suggesting robustness to the method.

Furthermore, forecasting models that consider the importance of model parameter optimization and the use of recent data, such as the BRNN approach, stand out in [52] research, which proposes an increment in the algorithm to deal with generalizability and overfitting problems in forecasting.

Ensemble learning techniques are a state-of-the-art learning approach that contributes due to statistical, computational, and representational advantages. These three issues are among the most important factors for which traditional machine learning approaches fail [53]. The main characteristics of these advantages [54] explain why ensemble techniques can often perform better than any individual classifier.

1. Statistic: is associated with the combination of several hypotheses in the solution space of the function, which tends to reduce the risk of the learning algorithm choosing a wrong hypothesis.
2. Computational: comes from of local search that is run from different starting points, generating a combination that can provide a better approximation to the true unknown hypothesis reducing the risk of choosing a wrong local minimum.
3. Representational: it is related to the combination of the hypotheses that it may be possible to expand the space of representable functions, and thus the learning algorithm may be able to form a more accurate approximation to the true unknown hypothesis.

One ensemble learning approach is the boosting method explored in this paper: the gradient boosting method of [55]. In their book, [56] states that the technique is promising in computational learning and also mentions its importance due to the iterative adjustment of the sequence of weak predictors to new weighted versions of the training data, generating at the end of the process combined outputs using the mean in regression problems. It also has higher computational efficiency compared to methods such as neural networks, generates a model with higher accuracy, and can reduce both model bias and variance [53], [57], [58].

[21] and [19] stated that there are few relevant studies on applying ensemble learning to real data sets in the stock market forecasting literature. [21] found only four studies that use ensemble methods in this domain; they are: [9] perform price direction prediction of emerging markets and combine Logistic Regression (LR), Neural Networks (NN), Support Vector Machines (SVM)

and Random Forest (RF); [59] explore the indices of three emerging markets and use LR, NN, AdaBoost (AB) and RF; in the studies of [60], the direction of the Dow Jones daily movements is predicted with AB and finally, [61] study emerging markets and combined NN, SVM, and RF methods. In these studies cited above, both results proved that, in contrast to the Efficient Market Hypothesis, it is possible to develop Machine Learning models for trading in financial markets capable of performing better than a random walk.

Some studies highlight machine learning algorithms' ability to deal with nonlinear and non-stationary problems [23]. Whose results show that the accuracy of these artificial intelligence methods is superior to traditional statistical methods, even more so with ensemble learning, whose goal is to reduce the bias and variance of the prediction and obtain better predictive performance than a single algorithm. Furthermore, prediction models that consider the importance of model parameter optimization and the use of recent data, such as the BRNN approach, stand out in the research of [52], which proposes an increment in the algorithm to deal with the generalization ability and overfitting problems in prediction.

It is also important to note that the final result of the prediction is not only dependent on the prediction algorithms but also has a strong relationship with the representation of the input. Therefore, identifying important features (variables) and using only them as input instead of all features can improve the prediction accuracy of prediction models. The filter-based attribute selection approach improves prediction accuracy, training time, and the ability to select a subset of relevant attributes. This approach is discussed in the research of [50], which uses this method to reduce data complexity and improve prediction accuracy.

# 3 Methodology

The process of obtaining the database, the financial assets selected, the period evaluated, and the techniques and algorithms adopted are presented. The full code of the solution, for reproducibility purposes, has been made publicly available at: github.com/ComputerFinance/CEJOR.

## 3.1 Data description

The sample was composed of daily closing prices of ten publicly traded companies from eight different economic sectors, chronologically ordered. These companies are part of the Ibovespa index, which measures the performance of the most representative assets of the Brazilian market and that stand out for market efficiency (achieved when the allocation of resources maximizes the total surplus) and liquidity. The analysis period is from Jan/1/2016 to Dec/30/2020 consisting of 1,254 trading days.

The relationship of a small sample of the database considering the variables collected for further analysis is represented in table 1.

**Table 1** Extraction of variables with attribute type and scale for analysis (partial, for reference only).

| QN[1] Ticker | QN[1] Sector[2] | QI[1] Data | QR[1] Closing (R$) |
|---|---|---|---|
| BRKM5 | Petrochemical | Jan/05/2016 | 25.93 |
| BRML3 | Real Estate and Construction | Jan/05/2016 | 7.26 |
| CSAN3 | Oil and Gas | Jan/05/2016 | 24.12 |
| EMBR3 | Industry | Jan/05/2016 | 29.32 |
| ENBR3 | Energy and Sanitation | Jan/05/2016 | 11.69 |
| GGBR4 | Steel and Mining | Jan/05/2016 | 4.33 |
| MRFG3 | Consumer and Retail | Jan/05/2016 | 5.95 |
| VALE3 | Steel and Mining | Jan/05/2016 | 12.52 |
| VIVT3 | Telecommunications | Jan/05/2016 | 30.40 |
| WEGE3 | Industry | Jan/05/2016 | 11.81 |

[1]QN=Qualitative Nominal, QI=Quantitative Interval, QR=Quantitative Rational

[2]Source: https://www.infomoney.com.br

## 3.2 Data preprocessing

This step consisted in minimizing the influence of eventual problems in the database, such as: noise and missing values, among others. These operations included: cleaning, integration, transformations, and dimensionality reduction. Each step performed is described below:

- **Cleaning**
  Performe the filling of missing values with the immediately preceding value of the data series to preserve the variability of the continuous time series [29].
- **Integration**
  We calculate the logarithmic return of the closing values in each asset, the difference of as profitability, as follows: $RCC = ln\left(\frac{P[t+s]}{P[t+0]}\right)$, where $RCC$ is the Continuously Compounded Return, $P[t]$ is the closing price of the asset on day $t$ and $s$ is the lag (forecast window).
  We also calculate the Rate of Change (ROC) of the closing price of the stock at time $t$, defined as $ROC_t = \frac{P[t]-P[t-n]}{P[t-n]}$, to provide the percentage difference of the series over two observations, and in this way we measure the ROC of the price $n$ days ago.
  Typically, our choice of variables is in line with previous research that has used at least a subset of these features, such as lagged index data and technical indicators [20], [7]. Following this research, the composition of the variables in this study comprises eighteen technical indicators calculated from the daily closing stock price information. They are segmented as follows: four technical, two markets, two of the stock, three artificial prices, and six artificial ROC, in addition to the target variable, briefly described in table A1 in section A.

- **Transformations**
  The z-score normalization is applied to variables $x$ given the standardized vector $\tilde{X} = \frac{x-\bar{x}}{s_x}$, where $\bar{x}$ is the mean and $s_x$ is the sample standard deviation of training variables, so that variables with large scale do not prevail in the prediction process. Precisely, we use the `preProcess` function of the caret package [62] (version 6.0-93) with the argument `method = c("center=0", "scale=1")`. We perform both the type conversions and the variable scales' normalization using the information obtained from the training set.
- **Dimensionality reduction**
  Variable selection acts as a filter that discards redundant or irrelevant variables and keeps a portion of the original variables in the database. Thus, we performed a ranking with the importance of the variables based on the results of the three selection algorithms: OneR, IG, and chi-squared. We explain it in detail in the section 3.4.1.

## 3.3 Performance measures

Among the various metrics available to evaluate the efficiency of the learning algorithm using the proposed method, we have chosen the ones listed below indicated in the [17] literature review by various authors in the field of financial series forecasting. Consider $y_i$ and $f(x_i)$ as the target real value and the predictive value of the forecast period, respectively; the length or lag of the forecast period $n$, the error or residuals $\mid u_i \mid = y_i - f(x_i)$, and the sample standard deviation $s_k$ over a series $k$. We consider the following metrics:

- **Coefficient of determination**: $R^2 = 1 - \sum_{i=1}^{n} u_i^2 / \sum_{i=1}^{n} (y_i - \overline{y})^2$, which measures the proportion of the total variation in the target variable explained by the regression model [32], where $\sum_{i=1}^{n} u_i^2$ is the sum of squares of the model residuals and $\sum_{i=1}^{n} (y_i - \overline{y})^2$, the sum of total squares. A value of 0 indicates that there is no linear relationship between the forecast model and the series, while a value of 1 indicates that the series is perfectly predictable by the model. The closer it is to 1, the better the model performs.
- **Mean absolute error**: $MAE = \frac{1}{n} \sum_{i}^{n} \mid y_i - f(x_i) \mid$, which measures the distance between predicted and actual values, specified above. The smaller the value, the better the model fit and the better the model fit.
- **Willmott index**: $d = 1 - \sum_{i=1}^{n} (D_{f(x_i)} - D_{y_i})^2 / \sum_{i=1}^{n} (\mid D_{f(x_i)} \mid + \mid D_{y_i} \mid)^2$, where $D_x = x - \overline{y}$ and $\overline{y}$ is the observed average of the target actual values, which is a measure that reflects the accuracy between predicted and observed values (for details see [63]).
- **kurtosis**: $1/n \sum_{i}^{n} (f(x_i) - f(\overline{x}) / s_{f(x_i)^4})$ measures the dispersion of errors between predictions and observed values. In other words, it is a way to measure how well the predicted value fits the actual value by evaluating the error dispersion of the residuals and the increase in the variance of the predictors (for details see [64]).

Finally, given the application of the two models (ensemble machine learning and ARIMA), we performed two hypothesis tests. The Wilcoxon test compares

the models' median of the absolute deviations of the predictions, and the F Test compares the differences in the variances of the forecasts in the two models.

## 3.4 Proposed model

Figure 1 illustrates a diagram of the proposed ML model and the main techniques used, detailed in the following sections.

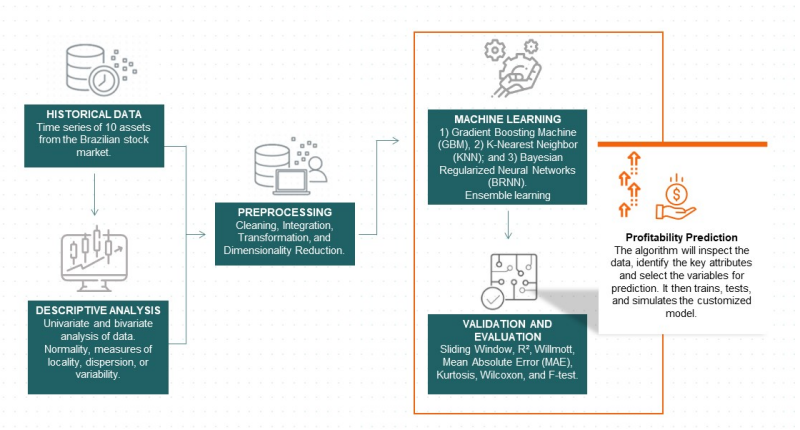


**Fig. 1** Proposed machine learning pipeline

### 3.4.1 Input variable selection

At this stage, we perform importance analysis of each variable to remove any irrelevant or redundant variables in the learning process for predicting future stock returns. We use embedded and filter-based approaches to evaluate each attribute for its characteristics and inherent information gain. The combination of the magnitude of the importance brings a less biased view of the relevance of the predictors. Because of the long lead time and the number of ten actions we selected, the time required to select the variables becomes critical. We present below the three techniques used to measure the importance of variables.

We first use the OneR algorithm that efficiently provides the results of classification tasks when applied to variable selection and is competitive with much more sophisticated methods [65]. Its operation starts from constructing a decision tree with rules that test a single variable. Each attribute defines a set of rules for each variable's value. The next step is to calculate the error rate for each feature, and the attribute with the lowest error rate is chosen [65].

The next algorithm was information gain which has fast execution and allows choosing the predictors from entropy calculation, unlike the OneR methodology, which induces a decision tree [66]. Entropy is used for its setup to measure the target attribute's randomness (difficulty in predicting). Entropy is a measure of random variable uncertainty that specifies the minimum number

of bits of information acquisition to encode the classification of an arbitrary member of $H(A)$ [67]. The authors [51] and [68] define the entropy of a random variable $A$ whose domain is $\{a_1, a_2, a_3, ..., a_v\}$ where the probability of observing each value is $p_1, p_2, p_3, ..., p_v$, as $H(A) = -\sum_i p_i log_2(p_i)$, given by:

1. $H(A) \in [O, log_2(v)]$;
2. $H(A)$ has a maximum equal to $log_2(v)$ if $p_i = p_j, \forall i \neq j$;
3. $H(A)$ has a minimum equal to 0 if $\exists i : p_i = 1$.

We close with a widely used test in the choice of attributes, the non-parametric chi-squared test [56]. It tests the adherence, homogeneity and independence between variables and checks whether the statistical model fits the data adequately.

In this approach to variable importance analysis, we calculate measures from the one rule, information gain, and chi-squared techniques to quantify the importance of each variable in the dataset concerning a given class or concept description. For example, with the IG technique, whose entropy formula measures the randomness of a random variable, the largest IG is obtained for each of the variables that define a set $A$ consisting of $s$ data samples. The variable with the highest IG is considered the most discriminating variable in the data set.

As a result aggregation calculation, we used the weighted sum of the number of occurrences of the frequencies of the variables in the nine positions. It was assigned weight 9 for the occurrence in position 1, 8 for position two to 1 for position 9. For example, if the variable $X_1$ appeared 40% of the times in position 3 (third most important variable) and 10% of the times in position 6, not occurring in other positions, its total importance is: $7 \times 0.4 + 4 \times 0.1 = 3.2$.

With this, we obtain a ranking of the variables computed with the measures of the three techniques described above. Finally, the relevant threshold is determined to select only the strong relevant variables used in the forecast models.

### 3.4.2 Predictive model

As with variable selection, we perform a new ensemble time series method to predict the direction of the return values via the GBM, KNN, and BRNN algorithms presented concisely below.

We use the GBM that combines weak models to generate a model with higher accuracy, improving the performance and computational efficiency [57], [58]. The BRNN had its choice influenced by the algorithm presenting fast convergence accuracy and contributing to the reduction of the effects of overfitting [69]. And finally the KNN algorithm, a methodology widely used in pattern-based predictions of financial series. The choice was strongly motivated due to its satisfactory results, ability to be applied in complex problems, and as a result of its incremental nature, since the training step consists of storing the history that makes up the predictors in memory [51].

### 3.4.3 Model simulation

We predict the ROC which is direction of the return values of the closing asset prices at $t + s$ from the value of the predictors at $t + 0$, where $s$ equals 180 calendar days (1 lag).

The simulation of the machine learning model consists of three steps. First, we consider a time window with $lag = 180$ days; for the target, we use forward lag and for the predictor variables, backward lag. Second, we partition our dataset into training, testing, and validation. We perform the division into $k$ groups (folds), with the same number of data in each, and group these divisions so that the training data is composed of $k - 1$ groups because the group that is not part of the training data we use for the model testing, while validation uses the out-of-sample data. Thus, we ensure in the construction of the forecast that the data is out-of-space and out-of-time. Third, with the earlier ensemble model training, we use these models to perform out-of-sample predictions on window size equal to $t+30$ and validate the model. The simulation logic follows in fig. 2 with details in the sequel.

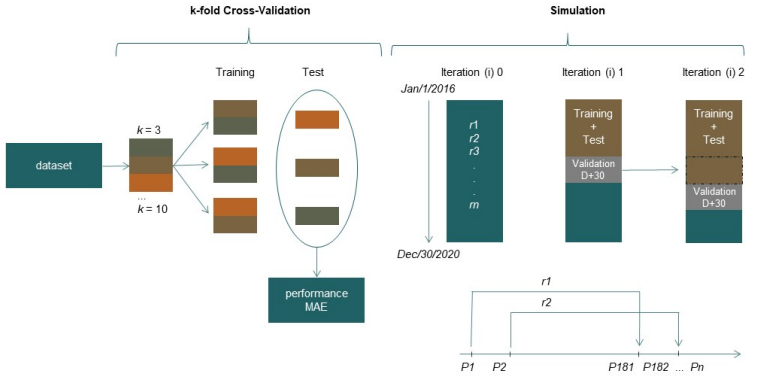


**Fig. 2** Ensemble model simulation [1] [2]

[1]$r[i]$ is the target value (ROC) at test start instance $(i)$
[2]$P[x]$ is the closing price of the asset on day $x$

We use 10-fold repeated cross-validation (3 repetitions x 10 folds) and create 30 data partitions with the resampling method to increase the reliability of the error estimates. In each cycle, a new partition is built with different datasets and tested on datasets independent of the training ones each time. The final predictor performance is measured by the $MAE$ of the observed performances on each test subset. This process is illustrated in fig. 2, using k-fold, e.g., equal to 3. We divide the data set into $k$ subsets of approximately equal size. We use the objects from $k - 1$ partitions in training the predictor, which we test on the remaining partition. In this way, the corresponding training and

testing set consists only of observations before the observation that forms the validation set.

Given that, algorithm 1 builds a training dataset by moving the input and output windows over the entire time series until the test instances are exhausted over a five-year horizon. The input pattern of the data is the past values of the predictor variables, and the output is the time series value at the 30-day forecast horizon. The prediction steps completed, the window size increases by $t+30$, with the range of data from the previous window remaining, plus the new range at the end.

**Algorithm 1** Sliding Window

1: $n :=$ number of bidding days related to a lag of 180 days
*Define each i-th prediction instance*
2: **for each** instance $i \in [1, i[$ **do**
3: $r_1 := \{P[j], j \in [n+1, i-1]\}$ ▷ Set of target variables
*Define the i-th window* $[1, i-1-n]$
*Each bid of the window, has predictor variables* $var_k$
4: $r_0 := \{\{\text{var}_k[j], \text{for all predictor variable } k\}, j \in [1, i-1-n]\}$
5: Calculate the remaining predictor variables concerning the instances between (1) and $(i-1-n)$;
6: Train the model on the generated dataset → target with reference to instances between $(n+1)$ and $(i-1)$ and predictors with reference to instances between (1) and $(i-1-n)$;
7: Estimate the value of the target on instance $(i)$ using the attributes of instance $(i-n)$;
8: Increase the value of $\{i\}$ by window size equal to $t+30$;
9: Repeat steps 1 to 8;
10: **Stop criterion** → the test instances are exhausted.

# 4 Results and Discussions

In this section, we first present the parameters used in each algorithm and the computational time. Then we highlight the results of the variable selection step for the prediction model. Immediately below are the outcomes of the prediction step of the direction of stock returns 180 days ahead. To complete, we perform some analyses to evaluate the comparative performance of the ensemble and ARIMA in-sample and ou-of-sample models. Finally, we present the results of the Wilcoxon non-parametric hypothesis test and the F-test to prove the difference in performance between the models.

## 4.1 Parameter setting

In table 2, we present the main parameters used in the algorithms. In the third and fourth columns are the final configurations of the experiments that we

justify by the good performance (within the metrics used in this work, detailed in section 3.3) and economy of computational resources, as shown in fig. 3. We performed 100 iterations per asset, totaling 1,000 iterations in 29 hours of execution. The model selected nine predictors presented in the next section.

**Table 2** Parameters of the machine learning models.

| Function | Parameter | Parameter value | Description |
|---|---|---|---|
| **Fselector** | | | |
| oneR | | | Gives the predictive accuracy of the attributes in descending order. |
| information.gain | | | Provides the weight of the discrete attributes based on their correlation with the continuous class attribute. |
| chi.squared | standard | standard | Determines whether two variables have a significant correlation. |
| **caret** | | | |
| preProcess | | | Estimates the parameters for each operation and predicts them. |
| predict | standard | standard | Estimates values based on the input data. |
| **gbm** | | | |
| gbm | shrinkage | 0.1 | Estimates values based on the input data learning rate. |
| | interaction.depth | 4 | Divisions must be executed on a tree starting from a single node. |
| | n.minobsinnode | 10 | Minimum observations at the leaf nodes of the trees. |
| | n.trees | 200 | Number of iterations. |
| **knn** | | | |
| knn | kmax | 5 | Maximum number of $k$. |
| | distance | 1 | Euclidean distance. |
| | kernel | optimal | Determines the type of core. |
| **brnn** | | | |
| brnn | neurons | 9 | The number of neurons per variable. |
| **cubist** | | | |
| cubist | committees | automatic | The number of interactions. |
| | neighbors | automatic | The number of nearest neighbors. |

Figure 3 presents the distribution of the execution time per iteration for the ML model, which resulted in an average of 104 seconds per iteration.

## 4.2 Variable Selection

As mentioned in section 3.4.1, the final goal of this stage is to prepare the data for the three machine learning models. Here, we used the OneR, information gain, and chi-squared algorithms, resulting in the importance of the variables that offer the greatest predictive power for each predictor. We provide this set of variables and the definition for each feature listed in table A1 in appendix A.

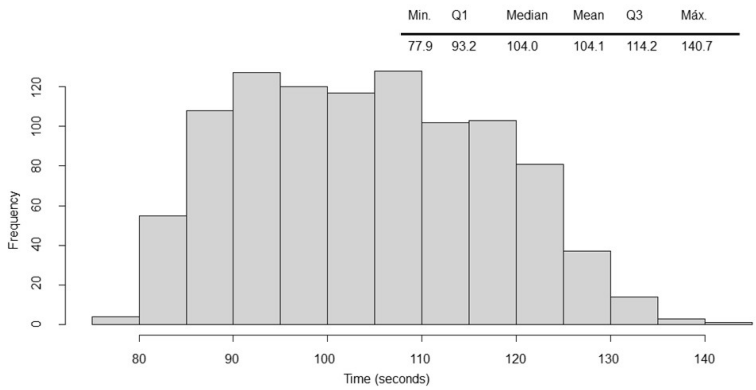


**Fig. 3** Runtime of the ensemble machine learning model

In fig. 4, we present the occurrence of the most frequent variables in each of the nine positions among the 18 predictors designed in the preprocessing stage. For example, the Importance 1 graph describes the variable *mml* (Slow Moving Average of the stock price) as the most frequently selected variable.

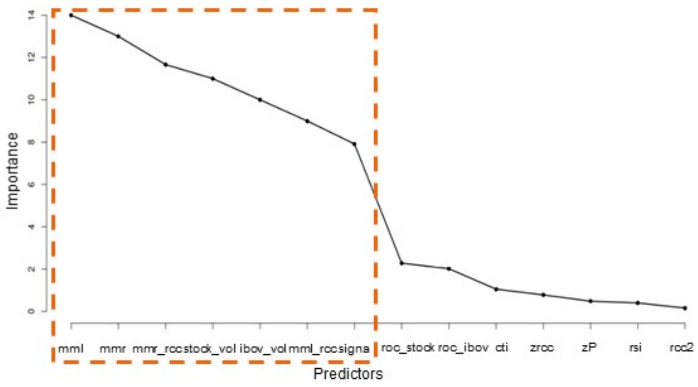


**Fig. 4** Most frequent variables in each of the nine predictor positions

Figure 5, shows the 14 variables selected in all iterations of the algorithm, with the average importance of the most frequent variables in descending order on the abscissa axis. We highlight the seven most important and frequent variables: *mml*, *mmr*, *mmr_rcc*, *stock_vol*, *ibov_vol*, *mml_rcc*, and *signal*, represented in table A1 for the remaining variables, we notice a drop in relative importance starting with the variable *stock_roc*. The results suggest that there

may be significant potential for prediction strategies in stock returns based on these variables.

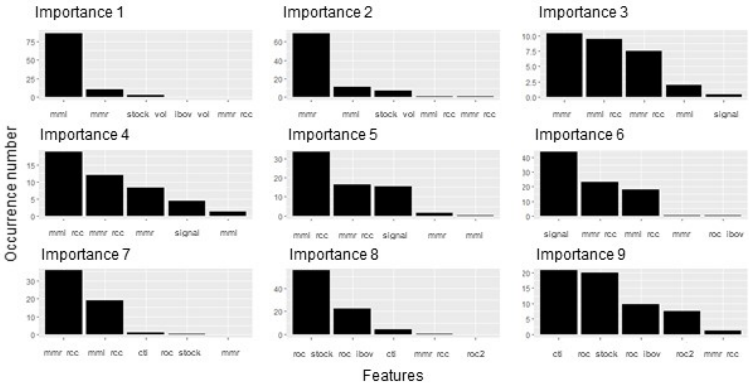


**Fig. 5** The importance of the predictors

## 4.3 Predictive Models

We use the evaluation metrics ($R^2$, Willmott, MAE, and kurtosis) to measure the stability of the models during the testing phase, developed for predicting the return values. Table 3 presents the performance metrics results and the average values obtained for each model. The calculation considered the data outside the training sample, using instances not seen by the trained model, to make the simulation closer to real life.

As we can see, the ML model outperforms the traditional ARIMA model in all metrics. The prediction errors measured by MAE indicate an average of 3.2% for the ML model, while the ARIMA model presented 32%; Willmott's agreement or fit index showed a better fit between the predicted and realized values in the ML model, with an average value close to 1.

In fig. 6 we compare the predicted return values with the actual values considering the six stocks that performed best in the ML model. Notably, the deviations from the prediction of the ML model present less dispersed values, measured by the more concentrated kurtosis. Besides higher $R^2$ values than the ARIMA, the ML model's prediction is adequate and explains the real data well.

Figure 7 compares the performance metrics of the models used. In all measures, the ARIMA was inferior to the ML model. To prove the model's difference in performance, we performed statistical significance analysis using the Wilcoxon non-parametric and F-test. The results in table 3 point to the acceptance of the alternative hypothesis that the deviations of the prediction of the

**Table 3** Performance of machine learning and ARIMA models.

| **Ticker** | **Model** | $R^2$ | **Willmott** | **MAE** | **kurtosis** | **Wilcoxon P-value** | **F P-valor** |
|---|---|---|---|---|---|---|---|
| Reference | | Bigger the better (max = 1) | | Smaller the better | Reference normal distribution = 3 | - | - |
| BRKM5 | | 0.970034 | 0.991709 | 0.032958 | 18,216822 | 0.240320 | 0.918488 |
| BRML3 | | 0.964421 | 0.991294 | 0.029572 | 9,109334 | 0.595266 | 0.758876 |
| CSAN3 | | 0.927291 | 0.986712 | 0.032980 | 15,010215 | 0.179380 | 0.663911 |
| EMBR3 | **Machine** | 0.972139 | 0.991109 | 0.031039 | 25,079928 | 0.033453 | 0.667514 |
| ENBR3 | **Learning** | 0.933676 | 0.990176 | 0.022744 | 10,569901 | 0.406338 | 0.601283 |
| GGBR4 | | 0.952141 | 0.986788 | 0.048842 | 6,332533 | 0.952020 | 0.792186 |
| MRFG3 | | 0.949316 | 0.987818 | 0.037041 | 8,241648 | 0.631479 | 0.883291 |
| VALE3 | | 0.957942 | 0.989490 | 0.038638 | 5,237920 | 0.087798 | 0.838927 |
| VIVT3 | | 0.888672 | 0.987060 | 0.025419 | 4,403632 | 0.607231 | 0.744828 |
| WEGE3 | | 0.959745 | 0.990208 | 0.029251 | 18,026506 | 0.033740 | 0.679524 |
| **Averages** | | **0,947538** | **0,989236** | **0,032848** | **12,022844** | **0,376703** | **0,754883** |
| **Ticker** | **Model** | $R^2$ | **Willmott** | **MAE** | **Kurtosis** | **Wilcoxon P-valor** | **F P-valor** |
| Reference | | Bigger the better (max = 1) | | Smaller the better | Reference normal distribution = 3 | - | - |
| BRKM5 | | 0.096021 | 0.449834 | 0.275175 | 2,752926 | 0.000796 | 0.000000 |
| BRML3 | | 0.018008 | 0.357350 | 0.331181 | 3,318790 | 0.109479 | 0.091276 |
| CSAN3 | | 0.001351 | 0.399164 | 0.351276 | 2,824857 | 0.334820 | 0.191151 |
| EMBR3 | | 0.075595 | 0.084169 | 0.600571 | 2,187870 | 0.686210 | 0.987901 |
| ENBR3 | **ARIMA** | 0.037713 | 0.466410 | 0.238525 | 3,521933 | 0.000958 | 0.000000 |
| GGBR4 | | 0.198983 | 0.557421 | 0.205132 | 2,458412 | 0.444255 | 0.825316 |
| MRFG3 | | 0.330618 | 0.246894 | 0.179208 | 2,395699 | 0.829849 | 0.000000 |
| VALE3 | | 0.036629 | 0.028636 | 0.640999 | 3,614053 | 0.784587 | 0.898107 |
| VIVT3 | | 0.003609 | 0.534696 | 0.277023 | 3,631085 | 0.322899 | 0.509939 |
| WEGE3 | | 0.110699 | 0.702129 | 0.088847 | 2,497368 | 0.005048 | 0.000000 |
| **Averages** | | **0,090923** | **0,382670** | **0,318794** | **2,920299** | **0,351890** | **0,350369** |

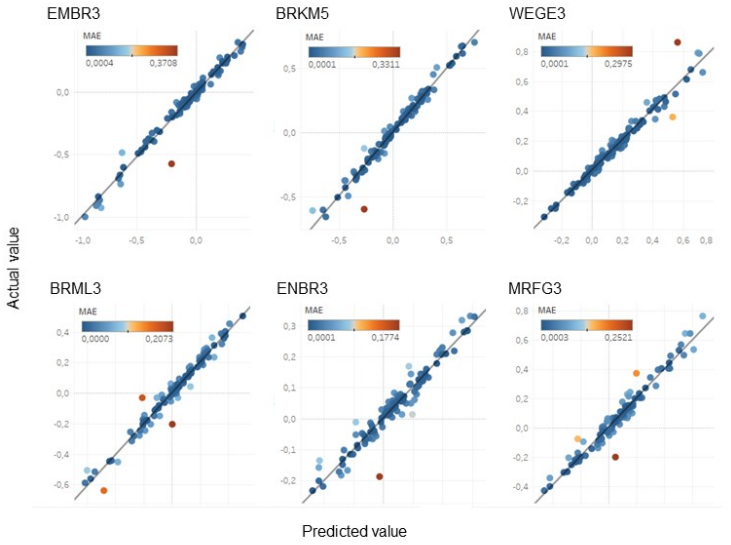


**Fig. 6** Comparison of predicted and actual return values for the top six models best in the ranking

ML model have a lower median and variance than the deviations generated by the ARIMA.

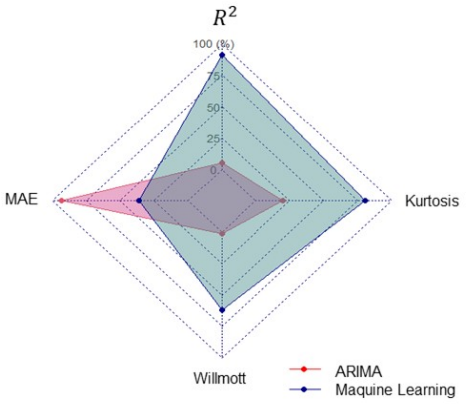


**Fig. 7** Model performance comparison

Similarly, table 4 ranks the selected assets according to each valuation metric and their averages. We highlight the six best-performing stocks: EMBR3, followed by BRKM5, WEGE3, BRML3, ENBR3, and MRFG3. These stocks, after submission of the statistical tests, whose values in table 3, point out that for the Wilcoxon test, only the medians of EMBR3 and WEGE3 in the ML

model, and BRKM5, ENBR3, and WEGE3 in the ARIMA model were below the significance value of 5%, while the predicted and actual values were very far from the median. On the other hand, the results of the F-test showed similar variance to the real values in the set of predicted stocks in the machine learning model, while in the ARIMA model, the stocks BRKM5, ENBR3, MRFG3, and WEGE3 were outside the significance level.

**Table 4** Stock performance ranking for the Machine Learning model.

| Ticker | $R^2$ | Willmott | MAE | Kurtosis | Average Ranking |
|---|---|---|---|---|---|
| EMBR3 | 1 | 3 | 5 | 1 | 2.50 |
| BRKM5 | 2 | 1 | 6 | 2 | 2.75 |
| WEGE3 | 4 | 4 | 3 | 3 | 3.50 |
| BRML3 | 3 | 2 | 4 | 6 | 3.75 |
| ENBR3 | 8 | 5 | 1 | 5 | 4.75 |
| MRFG3 | 7 | 7 | 8 | 7 | 7.25 |
| VALE3 | 5 | 6 | 9 | 9 | 7.25 |
| CSAN3 | 9 | 10 | 7 | 4 | 7.50 |
| VIVT3 | 10 | 8 | 2 | 10 | 7.50 |
| GGBR4 | 6 | 9 | 10 | 8 | 8.25 |

# 5 Conclusion

The proposed work addressed the stock prediction of the Brazilian stock market with machine learning techniques. We applied regression analysis of the time series of the stocks that use the combination of different machine learning algorithms according to the type of characteristic they represent, compared with the ARIMA econometric method. The results obtained support the use of the machine learning model for predicting stock returns based on the 7 most important variables that performed better than the traditional ARIMA model. The performance metrics of the ML model – $R^2$, Willmott, $MAE$ and Kurtosis – indicate that the results obtained in our research are encouraging for further investigations into the use of machine learning algorithms for prediction of stocks in developing economies, opposed to the efficient market assumption.

As suggestions for future works are the use of heuristic methods to select the set of assets with the lowest average correlation to build a probabilistic sampling by grouping; the incorporation of new independent variables, such as those suggested by the tsfresh package, which uses methods to evaluate the explanatory power and importance of the variables; the application of other non-parametric hypothesis tests, such as Friedman and Nemenyi; and, to reduce the processing time, introduce parallelization techniques in the algorithm.

# Appendix A Description of Variables for the Machine Learning model

**Table A1** Description of Variables for the Machine Learning model.

| Indicator | Variable name | Description |
|---|---|---|
| Technical | Return D+0 (rcc) | Log-return of the chosen assets in $t+0$ relative to $t-s$. |
| | MACD (signal) | Moving Average Convergence-Divergence: a trend indicator constructed by the relationship between two moving averages. |
| | RSI (rsi) | Relative Strength Index: a momentum indicator that measures the speed and magnitude of recent price changes. |
| | Moving Averages (mmr, mml) | 1 lag and 1/2 lag moving averages (e.g., for lag = 180 days (mmr), 1/2 lag = 90 days (mml)) that aim to identify trends from different time frames. |
| | CTI (cti) | Correlation Trend Indicator: trend indicator based on measuring Spearman's correlation between the historical price of each stock and the ideal trend. |
| Market | ROC Ibov (roc_ibov) | Ibovespa return on $D+0$. |
| | Ibov volatility (ibov_volat) | Standard deviation of the last 2 lag Ibovespa prices (current included). |
| Stock | ROC stock (roc_stock) | Stocks return on $t+0$. |
| | Stock volatility (stock_vol) | Standard deviation of the last 2 lag stock prices (current included). |
| Artificial Price | 1st price derivative (dP) | Discrete first-order derivative $EMA_i - EMA_{i-1}$ applied to the 10-period Exponential Moving Average of the stock price $EMA_i$. |
| | 2nd price derivative (d2P) | Discrete second-order derivative applied to the 10-period Exponential Moving Average of the stock price. |
| | Standard price (zP) | The z-score of the price, i.e., the price minus the average of the last 1-lag prices, divided by the standard deviation of the last 1-lag prices. |
| Artificial RCC | 1st RCC derivative (drcc) | Discrete first-order derivative applied to asset returns. |
| | 2nd RCC derivative (d2rcc) | Discrete second-order derivative applied to asset returns. |
| | $RCC^2$ (rcc2) | Asset returns squared. |
| | RCC Moving Averages (mmr_rcc, mml_rcc) | 1 lag (mmr_rcc) and 1/2 lag (mml_rcc) moving averages applied to asset returns. |
| | Normalized RCC (zrcc) | 1 Return minus the average of the last lag returns, and the result divided by the standard deviation of the last lag returns. |

## References

[1] Brandl, B., Keber, C., Schuster, M.G.: An automated econometric decision support system: forecasts for foreign exchange trades. Central European Journal of Operations Research **14**, 401–415 (2006). https://doi.org/10.1007/s10100-006-0013-8

[2] Pincheira, P.M., West, K.D.: A comparison of some out-of-sample tests of predictability in iterated multi-step-ahead forecasts. Research in Economics **70**(2), 304–319 (2016). https://doi.org/10.1016/j.rie.2016.03.002

[3] Meese, R.A., Rogoff, K.: Empirical exchange rate models of the seventies: Do they fit out of sample? Journal of International Economics **14**(1), 3–24 (1983). https://doi.org/10.1016/0022-1996(83)90017-X

[4] Engel, C., Mark, N.C., West, K.D.: Exchange rate models are not as bad as you think. NBER Macroeconomics Annual **22**(1), 381–442 (2007). https://doi.org/10.1086/ma.22.25554969

[5] Del Negro, M., Schorfheide, F.: DSGE model-based forecasting. In: Elliott, G., Timmermann, A. (eds.) Handbook of Economic Forecasting vol. 2, pp. 57–140. Elsevier, Amsterdam (2013). Chap. Second. https://doi.org/10.1016/B978-0-444-53683-9.00002-5

[6] Kumbure, M.M., Lohrmann, C., Luukka, P., Porras, J.: Machine learning techniques and data for stock market forecasting: A literature review. Expert Systems with Applications **197**, 116659 (2022). https://doi.org/10.1016/j.eswa.2022.116659

[7] Enke, D., Thawornwong, S.: The use of data mining and neural networks for forecasting stock market returns. Expert Systems with Applications **29**, 927–940 (2005). https://doi.org/10.1016/j.eswa.2005.06.024

[8] Edwards, R.D., Magee, J., Bassetti, W.H.C.: Technical Analysis of Stock Trends, Eleventh edn., p. 686. Boca Raton, CRC Press (2018). https://doi.org/10.4324/9781315115719

[9] Kumar, M., Thenmozhi, M.: Forecasting stock index movement: A comparation of support vector machines e random forest. In: Indian Institute of Capital Markets 9th Capital Markets, p. 16 (2006). https://doi.org/10.2139/ssrn.876544

[10] Box, G.M. G. E. P. Jenkins: Time Series Analysis, Forecasting and Control, pp. 1–575. Holden-Day, Oaklan, California. University of Wisconsin, U.S.A. and University of Lancaster, U.K. (1976). https://doi.org/10.4324/9781315115719

[11] Akaike, H.: In: Parzen, E., Tanabe, K., Kitagawa, G. (eds.) Information Theory and an Extension of the Maximum Likelihood Principle, pp. 199–213. Springer, New York, NY (1998)

[12] Hyndman, R.J., Khandakar, Y.: Automatic time series forecasting: the forecast package for R. Journal of Statistical Software **27**, 1–22 (2008). https://doi.org/10.18637/jss.v027.i03

[13] Martínez, F., Frías, M.P., Pérez, M.D., Rivera, A.J.: A methodology for applying k-nearest neighbor to time series forecasting. Artificial Intelligence Review **52**, 2019–2037 (2017). https://doi.org/10.1007/s10462-017-9593-z

[14] Wiranata, R.B., Djunaidy, A.: The stock exchange prediction using machine learning techniques: A comprehensive and systematic literature review. Journal of Computer Science and Information **14**, 91–112 (2021). https://doi.org/10.21609/jiki.v14i2.935

[15] de Oliveira, D.S.P., Montes, G.C.: Forecasting sovereign risk perception of brazilian bonds: an evaluation of machine learning prediction accuracy. International Journal of Emerging Markets **ahead-of-prin**, (2021). https://doi.org/10.1108/IJOEM-01-2021-0106

[16] Jiang, M., Liu, J., Zhang, L., Liu, C.: An improved stacking framework for stock index prediction by leveraging tree-based ensemble models and deep learning algorithms. Physica A: Statistical Mechanics and its Application **541**, 122272 (2020). https://doi.org/10.1016/j.physa.2019.122272

[17] Henrique, B.M., Sobreiro, V.A., Kimura, H.: Literature review: Machine learning techniques applied to financial market prediction. Expert Systems with Applications **124**, 226–251 (2019). https://doi.org/10.1016/j.eswa.2019.01.012

[18] Huck, N.: Large data sets and machine learning: Applications to statistical arbitrage. European Journal of Operational Research **278**(1), 330–342 (2019). https://doi.org/10.1016/j.ejor.2019.04.013

[19] Nobre, J., Neves, R.F.: Combining principal component analysis, discrete wavelet transform and xgboost to trade in the financial markets. Expert Systems with Applications **125**, 181–194 (2019). https://doi.org/10.1016/j.eswa.2019.01.083

[20] Khaidem, L., Saha, S., Dey, S.R.: Predicting the direction of stock market prices using random forest. Applied Mathematical Finance (2016). https://doi.org/https://arxiv.org/abs/1605.00003

[21] Ballings, M., den Poel, D.V., Hespeels, N., Gryp, R.: Evaluating multiple classifiers for stock price direction prediction. Expert Systems with Applications **42**, 7046–7056 (2015). https://doi.org/10.1016/j.eswa.2015.05.01

[22] B3: B3 publishes the third preview of ibovespa and other indices. PhD thesis, B3 Hypothetical Portfolios (2022). https://www.b3.com.br/pt_br/noticias/carteiras-teoricas-8AE490C97DB66D56017E07710611523C.htm

[23] Chen, R., Xiao, H., Yang, D.: Autoregressive models for matrix-valued time series. Journal of Econometrics **222**, 539–560 (2021). https://doi.org/10.1016/j.jeconom.2020.07.015

[24] Linton, O., Todorov, V., Zhang, Z.: Editorial for the special issue on financial econometrics in the age of the digital economy. Journal of Econometrics **222**, 265–268 (2020). https://doi.org/10.1016/J.JECONOM.2020.07.001

[25] Tsay, R.S.: Multivariate Time Series Analysis: With R and Financial Applications. John Wiley & Sons, Booth School of Business, University of Chicago, Chicago, IL (2014)

[26] Box, G.E.P., Jenkins, G.M., Reinsel, G.C., Ljung, G.M.: Time Series Analysis: Forecasting and Control, 5th edn. John Wiley & Sons, Inc., Hoboken, New Jersey. Book Series:Wiley Series in Probability and Statistics (2016)

[27] Chatfield, C., Xing, H.: The Analysis of Time Series: An Introduction with R, 7th edn. Chapman & Hall/CRC Texts in Statistical Science. CRC Press, Taylor & Francis Group, Boca Raton, Florida (2019)

[28] Diebold, F.X.: The past, present, and future of macroeconomic forecasting. Journal of Economic Perspectives **12**, 175–192 (1998). https://doi.org/10.1257/jep.12.2.175

[29] Wooldridge, J.M.: Introductory Econometrics: A Modern Approach, 7th edn. Cengage Learning, Boston, Massachusetts, EUA (2019)

[30] Bueno, R.L.S.: Econometria de Séries Temporais, Segunda edn. Cengage Learning, São Paulo, Brasil (2018)

[31] Morettin, P.A.: Econometria Financeira: Um Curso em Séries Temporais Financeiras, Segunda edn. Editora Blucher, São Paulo, SP, Brasil (2017)

[32] Gujarati, D.N., Porter, D.C.: Econometria Básica, Quinta edn. MC Graw Hill, São Paulo, SP, Brasil (2011)

[33] Ang, A., Bekaert, G.: Stock return predictability: Is it there? The Review

of Financial Studies **20**, 651–707 (2007). https://doi.org/10.1093/rfs/hhl021

[34] Chen, M.Y., Chen, B.T.: A hybrid fuzzy time series model based on granular computing for stock price forecastin. Information Sciences **294**, 227–241 (2015). https://doi.org/10.1016/j.ins.2014.09.038

[35] Efendi, R., Arbaiy, N., Deris, M.M.: A new procedure in stock market forecasting based on fuzzy random auto-regression time series model. Information Sciences **441**, 113–132 (2018). https://doi.org/10.1016/j.ins.2018.02.016

[36] Shukor, S.A., Sufahani, S.F., Khalid, K., Wahab, M.H.A., Idrus, S.Z.S., Ahmad, A., Subramaniam, T.S.: Forecasting stock market price of gold, silver, crude oil and platinum by using double exponential smoothing, holt's linear trend and random walk. Journal of Physics: Conference Series **1874**(1), 012087 (2021)

[37] Uras, N., Marchesi, L., Marchesi, M., Tonelli, R.: Forecasting bitcoin closing price series using linear regression and neural networks models. PeerJ Computer Science **6:e279** (2020). https://doi.org/10.7717/peerj-cs.279

[38] Engle, R.F., Lilien, D.M., Robins, R.P.: Estimating time varying risk premia in the term structure: The arch-m model. The Econometric Society **55**, 391–407 (1987). https://doi.org/10.2307/1913242

[39] Zivot, E.: Practical issues in the analysis of univariate garch models. In: Handbook of Financial Time Series, pp. 113–155. Springer, ??? (2009)

[40] Babu, N., Reddy, B.E.: A moving-average filter based hybrid ARIMA-ANN model for forecasting time series data. Applied Soft Computing **23**, 27–38 (2014). https://doi.org/10.1016/j.asoc.2014.05.028

[41] Ariyo, A.A., Adewumi, A.O., Ayo, C.K.: Stock price prediction using the ARIMA model. In: 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, pp. 106–112 (2014). IEEE

[42] Hsu, Y.T., Liu, M.C., Yeh, J., Hung, H.F.: Forecasting the turning time of stock market based on Markov-Fourier grey model. Expert Systems with Applications **36**, 8597–8603 (2009). https://doi.org/10.1016/j.eswa.2008.10.075

[43] M.L.P, M.: Selection of best ARIMA modeling approach for forecasting time series patterns: A case study on colombo stock exchange. International Journal of Business,Economics and Management Works **4**(11), 1–5 (2017). https://doi.org/10.5281/zenodo.1050921

[44] Bahrammirzaee, A.: A comparative survey of artificial intelligence applications in finance: Artificial neural networks, expert system and hybrid intelligent systems. Neural Computing and Applications **19**, 1165–1195 (2010). https://doi.org/10.1007/s00521-010-0362-z

[45] Huang, S.C., Wu, T.K.: Integrating GA-based time-scale feature extractions with SVMs for stock index forecasting. Expert Systems with Applications **35**, 2080–2088 (2008). https://doi.org/10.1016/j.eswa.2007.09.027

[46] Lin, C.S., Chiu, S.H., Lin, T.Y.: Empirical mode decomposition–based least squares support vector regression for foreign exchange rate forecasting. Economic Modelling **29**(6), 2583–2590 (2012). https://doi.org/10.1016/j.econmod.2012.07.018

[47] Yu, L., Chen, H., Wang, S., Lai, K.K.: Evolving least squares support vector machines for stock market trend mining. IEEE transactions on evolutionary computation **13**(1), 87–102 (2008). https://doi.org/10.1109/TEVC.2008.928176

[48] Zhang, N., Lin, A., Shang, P.: Multidimensional k-nearest neighbor model based on EEMD for financial time series forecasting. Physica A: Statistical Mechanics and its Applications **477**, 161–173 (2017). https://doi.org/10.1016/j.physa.2017.02.072

[49] Nguyen, H.T., Tran, T.B., D., B.P.H.: An effective way for taiwanese stock price prediction: Boosting the performance with machine learning techniques. Concurrency and Computation Practice and Experience (2021). https://doi.org/10.1002/cpe.6437

[50] Huang, C.-L., Tsai, C.-Y.: A hybrid sofm-svr with a filter-based feature selection for stock market forecasting. Expert Systems with applications **36**, 1529–1539 (2009). https://doi.org/10.1016/j.eswa.2007.11.062

[51] Lorena, A.C., Facelli, K., Gama, J., Almeida, T.A., Carvalho, A.C.P.L.F.: Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina, Segunda edn. LTC, Rio de Janeiro, Brasil (2021)

[52] Yan, D., Zhou, Q., Wang, J., Zhang, N.: Bayesian regularisation neural network based on artificial intelligence optimisation. International Journal of Production Research **55**(8), 2266–2287 (2016). https://doi.org/10.1080/00207543.2016.1237785

[53] Zhou, Z.-H.: Ensemble Methods Foundations and Algorithms, p. 234. Chapman & Hall/CRC, Boca Raton, London, New York (2012)

[54] Dietterich, T.G.: Ensemble Methods in Machine Learning, pp.

1–15. Springer, Berlin, Heidelberg (2000). https://doi.org/10.1007/3-540-45014-9_1

[55] Friedman, J.H.: Greedy function approximation: A gradient boosting machine. The Annals of Statistics **29**, 1189–1232 (2001). https://doi.org/10.1214/aos/1013203451

[56] Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference and Prediction, 2nd edn. Springer, New York (2009)

[57] Flach, P.: Machine Learning: The Art and Science of Algorithms that Make Sense of Data. Cambridge University Press, New York (2012)

[58] Krauss, C., Do, X.A., Huck, N.: Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P500. European Journal of Operational Research **259**(2), 689–702 (2017). https://doi.org/10.1016/j.ejor.2016.10.031

[59] Rodriguez, P.N., Rodriguez, A.: Predicting stock market indices movements. COMPUTATIONAL FINANCE AND ITS APPLICATIONS In Marco Constantino, Carlos Brebia, eds., Wessex Institute of Technology, Southampton (2004)

[60] Lunga, D., Marwala, T.: Online forecasting of stock market movement direction using the improved incremental algorithm. Lecture Notes in Computer Science **4234**, 440–449 (2006). https://doi.org/10.1007/11893295_49

[61] Patel, J., Shah, S., Thakkar, P., Kotecha, K.: Predicting stock and stock price index movement using trend deterministic data preparation and machine learning technique. Expert Systems with Applications **42**, 259–268 (2015). https://doi.org/10.1016/j.eswa.2014.07.040

[62] Kuhn, M.: Building predictive models in r using the caret package. Journal of Statistical Software **28**, 1–26 (2008). https://doi.org/10.18637/jss.v028.i05

[63] Willmott, C.J., Robeson, S.M., Matsuura, K.: Short communication: A refined index of model performance. International Journal of Climatology **32**, 2088–2094 (2012). https://doi.org/10.1002/joc.2419

[64] McAlevey, L.G., Stent, A.F.: Kurtosis: a forgotten moment. International Journal of Mathematical Education in Science and Technology, 1464–5211 (2017). https://doi.org/10.1080/0020739X.2017.1357848

[65] Holte, R.C.: Very simple classification rules perform well on most commonly used datasets. Machine Learning **11**, 63–90 (1993). https://doi.org/10.1023/A:1022631118932

[66] Azhagusundari, B., Thanamani, A.S.: Feature selection based on information gain. International Journal of Innovative Technology and Exploring Engineering (IJITEE) **2**, 18–21 (2013)

[67] Mitchell, T.M.: Machine Learning. McGraw-Hill, New York (1997)

[68] Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach, 3rd edn. Pearson Education, New Jersey (2010)

[69] Lahmiri, S., Bekiros, S.: Intelligent forecasting with machine learning trading systems in chaotic intraday Bitcoin market. Chaos, Solitons & Fractals **133**, 109641 (2020). https://doi.org/10.1016/j.chaos.2020.109641