Springer Open

Search 🔍

Get published

Explore Journals

Books

About

Login

Menu ▾

Journal of Big Data

| About | **Articles** | Submission Guidelines |

Research │ Open Access │ Published: 11 March 2020

A comprehensive evaluation of ensemble learning for stock-market prediction

Isaac Kofi Nti ✉, Adebayo Felix Adekoya & Benjamin Asubam Weyori

Abstract

Stock-market prediction using machine-learning technique aims at developing effective and efficient models that can provide a better and higher rate of prediction accuracy. Numerous ensemble regressors and classifiers have been applied in stock market predictions, using different combination techniques. However, three precarious issues come in mind when constructing ensemble classifiers and regressors. The first concerns with the choice of base regressor or classifier technique adopted. The second concerns the combination techniques used to assemble multiple regressors or classifiers and the third concerns with the quantum of regressors or

classifiers to be ensembled. Subsequently, the number of relevant studies scrutinising these previously mentioned concerns are limited. In this study, we performed an extensive comparative analysis of ensemble techniques such as boosting, bagging, blending and super learners (stacking). Using Decision Trees (DT), Support Vector Machine (SVM) and Neural Network (NN), we constructed twenty-five (25) different ensembled regressors and classifiers. We compared their execution times, accuracy, and error metrics over stock-data from Ghana Stock Exchange (GSE), Johannesburg Stock Exchange (JSE), Bombay Stock Exchange (BSE-SENSEX) and New York Stock Exchange (NYSE), from January 2012 to December 2018. The study outcome shows that stacking and blending ensemble techniques offer higher prediction accuracies (90–100%) and (85.7–100%) respectively, compared with that of bagging (53–97.78%) and boosting (52.7–96.32%). Furthermore, the root means square error (RMSE) recorded by stacking (0.0001–0.001) and blending (0.002–0.01) shows a better fit of ensemble classifiers and regressors based on these two techniques in market analyses compared with bagging (0.01–0.11) and boosting (0.01–0.443). Finally, the results undoubtedly suggest that an innovative study in the domain of stock market direction prediction ought to include ensemble techniques in their sets of algorithms.

Introduction

The stock market is considered to be a stochastic and challenging real-world environment, where the stock-price movements are affected by a considerable number of factors [1, 2]. Billions of structured and unstructured data are generated daily from the stock market around the globe, increasing the "volume", "velocity", "variety" and "veracity" of stock market data, and making it complex to analyse [1, 3]. In analysing this "Big Data" from the stock market, two methods have generally been accepted, namely: fundamental analysis and technical analysis. The fundamental analysis focuses on the economic trends of local and international milieus, public sentiments, financial-statement and assets reported by companies, political conditions and companies associations worldwide [1, 4]. The technical analysis is based on statistical analysis, using the historical movement of the stock-prices. Technical indicators such as moving-average, dead cross and golden-cross are employed for effective stock trading decisions. Despite the existence of these techniques, market analysis is still challenging and open [1].

To overcome the challenges in the stock market analysis, several computational models based on soft-computing and machine learning paradigms have been used in the stock-market analysis, prediction, and trading. Techniques like Support Vector Machine (SVM) [2, 5], DTs [6], neural networks [7], Naïve Bayes [8, 9] and artificial neural networks (ANN) [10, 11] were reported to have performed better in stock-market prediction than conventional arithmetic methods like Logistic regression (LR), in respect of error prediction and accuracy.

Nevertheless, ensemble learning (EL) based on a learning-paradigm that combines multiple learning algorithms, forming committees to improve-predictions (stacking and blending) or decrease variance (bagging), and bias (boosting) is believed to perform better than single classifiers and regressors [12, 13].

Succinctly, EL techniques have been applied in serval sectors such as health [14], agriculture [15], energy [16], oil and gas [17], and finance [12, 18]. In all these applications, their reported accuracies support the argument that ensemble classifiers or regressors are often far more precise than the discrete classifiers or regressors. For this reason, the need for building a better-ensemble classification and regression models has become a critical and active research area in supervised learning, with boosting and bagging being the most common amalgamation methods used in the literature [16].

Despite numerous works revealing the dominance of ensemble classifier over single classifier, most of these studies only ensemble a specific type of classifier or regressor for stock-market prediction, such as NN [18,19,20], DT [21, 22] and SVM [12, 23]. Also, most previous studies [12, 19, 21, 22, 24,25,26,27,28,29,30], on ensemble methods for stock-market predictions adopted the decrease variance approach (boosting or bagging) and experimented with data from one country. Furthermore, a comparison between bagging (BAG) and boosting (BOT) combination techniques by [12, 21] revealed that the BAG technique outperformed

the BOT technique. However, the conclusion of these studies pointed out that the performance of ensemble classifiers using boosting or bagging in stock-market prediction is territory dependent. Thus, the authors foresee that some ensemble methods may perform better on data from some parts of the globe than other parts. This assumption calls for the application of different ensemble techniques to be benchmarked with stock-data from different continents, to ascertain their performance.

Besides, little is known on comparing ensemble classifiers and regressors using different combination techniques with same or diverse base learners in predicting the stock market. Hence, in stock-market prediction, to the best of our knowledge, there is no comprehensive comparative study to evaluate the performances of a good pool of diverse ensembles regressors and classifiers based on stock-data from three or more continents.

Therefore, this study seeks to perform a comprehensive comparative study of ensemble learning methods for classification and regression machine learning tasks in stock market prediction. The following specific objectives aiding this study are as follows:

1. i.
   To bring together the theory of EL and appreciate the algorithms, which use this technique.

2. ii.

To review some of the recently published articles on ensemble techniques for classification and regression machine learning tasks in stock market prediction.

3. iii.

   To set up ensemble classifiers and regressors with DTs, SVM and NN using stacking, blending, bagging, and boosting combination techniques.

4. iv.

   To examine and compare execution times, accuracy, and error metric of techniques in (iii) over stock data from GSE, JSE, NYSE and BSE-SENSEX.

Hopefully, this paper brings more clarity on which ensembles techniques is best suitable for machine learning tasks in stock market prediction. Again, offer help to beginners in the machine-learning field, to make an informed choice concerning ensemble methods that quickly offer best and accurate results in stock-market prediction. Furthermore, we probe the arguments made in [12, 21] about the consistency of ensemble learning superiority over stock data from different countries. Finally, this paper contributes to the literature in that it is, to the best of our knowledge, the first in stock market prediction to make such an extensive comparative analysis of ensemble techniques.

The remaining sections of the paper are organised as follows. "Related works evaluation" section presents a review of related works. In "Procedure of proposed

method" section, we present a quick dive-into basic and advanced ensemble methods and the study procedure. "Predictive models" section discusses the results of empirical studies. "Ensemble methods (EMs)" section concludes this study and describes avenues for future research.

Related works evaluation

Literature has shown that the applications of some powerful ML algorithms have significantly improved the accuracy of stock prices classification and prediction [31, 32]. As such, ML has drawn the attention in stock market prediction, and several ensemble ML techniques have recorded high prediction accuracy in current studies.

Sohangir et al. [33] examined the ability of deep learning techniques such as LSTM and CNN to improve the prediction accuracy of the stock using public sentiments. The out of the study showed that deep learning technique (CNN) outperformed ML algorithms like Logistic regression and Doc2vec. Their Simulation outcome demonstrated the attractiveness of their proposed ensemble method compared with auto-regressive integrated moving average, generalised autoregressive conditional heteroscedasticity. Likewise, Abe et al. [34] applied a deep neural network technique to predict stock price and reported that deep technique is more accurate than shallow neural networks.

An ensemble of state-of-the-art ML techniques, including deep neural networks, RF and gradient-boosted trees were proposed in [35], to predict the next day stock price return on the S&P 500. Their experimental findings were hopeful, signifying that a sustainable profit prospect in the short-run is exploitable through ML, even in the case of a developed-market. Qiu et al. [36] presented a stock prediction model based on ensemble ν-Support Vector Regression Model.

Similarly, an ensemble of Bayesian model averaging (BMA), weighted-average least squares (WALS), least absolute shrinkage and selection operator (LASSO) using AdaBagging was proposed in [24] to predict stock price. Pasupulety et al. [37] proposed an ensemble of extra tree regressor and support vector regressor using stacking to predict the stock price based on public sentiment. Pulido et al. [38] ensembled NN with fuzzy incorporation (type-1 and type-2) for predicting the stock market [38], they achieved a high prediction accuracy by the proposed model compared with single NN classifier. An ensemble of trees in an RF using LSboost was carried out [25]; the study achieved reduced prediction error.

A Comparison of single, ensemble and integrated ensemble ML techniques to predict the stock market was carried out in [39]. The study showed that boosting ensemble classifiers outperformed bagged classifiers. Sun et al. [26] proposed an ensemble LSTM using AdaBoost for stock market prediction. Their results show that the proposed AdaBoost-LSTM ensemble outperformed some other single

forecasting models. A homogenous ensemble of time-series models including SVM, logistic regression, Lasso regression, polynomial regression, Naive forecast and more was proposed in [40] for predicting stock price movement. Likewise, Yang et al. [41] ensembled SVM, RF and AdaBoost using voting techniques to predict a buy or sell of stocks for intraday, weekly and monthly. The study shows that the ensemble technique outperformed single classifier in terms of accuracy. Gan et al. [42] proposed an ensemble of feedforward neural networks for predicting the stock closing price and reported a higher accuracy in prediction as compared with single feedforward neural networks.

In another study, a 2-phase ensemble framework, including several non-classical disintegration models, namely, ensemble empirical mode decomposition, empirical mode decomposition, and complete ensemble empirical mode decomposition with adaptive noise, and ML models, namely, SVM and NN, was proposed for predicting stock-prices [43]. Implementation and evaluation of RF robustness in stocks selection strategy was carried out [31]. Using the fundamental and technical dataset, they concluded that in sound stocks investment, fundamental features, and long-term technical features are of importance to long-term profit. Mehta et al. [44] proposed a weighted ensemble model using weighted SVM, LSTM and multiple regression for predicting the stock market. Their results show that the ensemble learning technique attained maximum accuracy with lesser variance in stock prediction.

Similarly, Assis et al. [45] proposed an NN ensemble for predicting stock price movement. A deep NN ensemble using bagging for stock market prediction was proposed in [29]. The study revealed that assembling several neural networks to predict stock price movement is highly accurate than a single deep neural network. Jiang et al. [27] implemented different state-of-the-art ML techniques, including a tree-based and LSTM ensemble using stacking combination technique to predict stock price movement based on both information from the macroeconomic conditions and historical transaction data. The authors recorded an accuracy of 60–70% on average. Kohli et al. [46] examined different ML algorithms (SVM, RF, Gradient Boosting and AdaBoost) performance in stock market price prediction. The study showed that AdaBoost outperformed Gradient Boosting in terms of predicting accuracy.

The work in [19] presents an ensemble classifier of NN using bagging. Their results revealed that the ensemble of NN performs much better than a single NN classifier. Equally, Wang et al. [4] proposed an RNN ensemble framework that combines trade-based features deduced from historical trading records and characteristic features of the list companies to perceive stock-price manipulation activities effectively. Their experimental results reveal that the proposed RNN ensemble outperforms state-of-the-art methods in distinguishing stock price manipulation by an average of 29.8% in terms of AUC value. Existing studies have shown that ensemble classifiers and regressors are of higher

predicting accuracy than a single classifier and regressor.

In the same way, Ballings et al. [12] compared LR, NN, K-Nearest Neighbour (K-NN), and SVM ensembles using bagging and boosting. The study results revealed that bagging algorithm (random forest) outperformed boosting algorithm (AdaBoost). Nevertheless, the study concluded that the performance of ensemble methods is dependent on the domain of the dataset used for the study. Therefore, to obtain a generalisation of EL methods, a comprehensive comparison among ensemble methods using datasets from different continents are required.

Table 1 (Appendix A), present a summary of pertinent studies on stock market prediction using EL based on different combination techniques. We categorised the relevant literature based on (i) the base (weak) learner and the total number used. (ii) The type of machine learning task (classification or regression). (iii) The origin of the data used for the experimental analysis. (iv) The combination technique used and (v) evaluation metric used to contrast and compare the relative metamorphoses.

Table 1 Comparison of related studies

As observed in Table 1 (Appendix A), creating of ensemble classifiers and regressors in the domain of stock-market predictions has become an area of interest in recent studies. However, most of these studies [12, 19, 21, 22, 24,25,26,27,28,29,30] were

based on boosting (BOT) or bagging (BAG) combination method. Only a few [4, 18, 20, 37] examined ensemble classifiers or regressors based on stacking or blending combinational technique.

Once more, as shown in Table 1 (Appendix A) most of the studies compared between ensemble classifiers [12, 18,19,20, 22, 23, 28] or regressors [21, 30] machine learning algorithms, but not both. On the other hand, literature shows that most machine learning algorithms can be used for classification and regression tasks. However, some are better for classification than regression, while others are vice versa [47, 48]. Hence a good comparison among ensemble methods should cover both regression and classification tasks with same weaker learners.

Concerning combination techniques, Table 1 (Appendix A) affirms that a high percentage of existing literature used either BAG or BOT for classifier ensembles. Thus, only a few minorities examine the performance of different classifier using BAG and BOT and Stacking (STK).

Furthermore, the quantity of assembled classifiers in previous studies is diverse, whiles some used different numbers, other used fixed of say 10 for comparisons, and to the best of our knowledge, previous studies did not compare ensembles classifiers and regressor with same single classifiers using same combinational methods.

Considering the above discussions presented in Table 1 (Appendix) carefully it leaves a gap for

conducting a comprehensive comparison study of ensemble classifiers and regressors of the same or a different number of base learners using different combination methods for stock-market prediction.

Procedure of proposed method

This section presents the details of Machine Learning (ML) algorithms adopted in this study and their implementation for predicting the stock market.

Predictive models

Like many other studies [18, 19, 21, 23, 49], this study adopts three bases line ML algorithms, namely DT, SVM and NN, based on their superiority for ensemble learning in financial analysis.

Decision tree (DT)

DT is a flow-chart-like tree structure that uses a branching technique to clarify every single likely result of a decision. The interpretability and simplicity of DT, its low-slung computational cost and the ability to represent them graphically have contributed to the increase in its use for classification task [50]. An information gain approach was used to decide the appropriate property for each node of a generated tree. The test attributes of each current node are selected based on the attribute that has the maximum information. The operation of a DT on a dataset (DS) is expressed in [51] as follows:

1. 1.

Estimate the entropy *E (S)* value of the *DS* as expressed in Eq. (1).

$$E(S) = \sum_{i=1}^{m} -p_i \log_2 p_i$$

(1)

where E(S) = entropy of a collection of *DS*, *m* = represents the number of classes in the system and pi = represents the number of instances proportion that belongs to class *i*.

2. 2.

    Calculate the information gain for an attribute *K*, in a collection *S*, as expressed in Eq. (2). where *E(S)* represents the entropy of the entire collection and S_u = the set of instances that have value u for attribute *K*.

$$G(S, K) = E(S) - \sum_{u \in values(K)} \frac{S_u}{S} E(S_u).$$

(2)

Support vector machine (SVM)

SVM is a supervised machine learning tool used for regression and classification tasks [52]. SVM serves as the linear separator sandwiched between two data nodes to detect two different classes in the multidimensional environs. The following steps show the implementation of SVM.

Let DS be the training dataset,

$$DS = \{(x_i, y_i, \ldots, (x_n, y_n))\} \in X.\,R \quad \text{where } i = (1, 2, 3, \ldots, n).$$

The SVM represents *DS* as points in an N-dimensional space and then tries to develop a

hyperplane that will split the space into specific class labels with a right margin of error [51]. Equations (3) and (4) shows the formula used in the algorithm for the SVM optimisation.

$$\min_{d,b\omega,} \frac{1}{2} W^T W + C \sum_{i=1}^{n} \omega i$$

(3)

$$\text{subject to } y_i \left(W^T \theta \left(x_i + b \right) \geq 1 - \omega i \right), \quad \omega i > 0$$

(4)

The function θ of vectors xi (DS) are mapped in space dimension of higher space. In this dimension, the SVM finds a linear separating hyperplane with the best margin. The kernel function can be formulated as $K(x_i, x_j) \equiv \theta(x_i)^T \theta(x_j)$. The Radial Basis Function (RBF) kernel expressed in Eq. (5) was adopted for this study.

$$RBF\colon K\left(x_i, x_j\right) = \exp\left(-y||x_i - x_j||^2\right), \quad y > 0$$

(5)

where $(x_i - x_j)$ is the Euclidian distance between two data point.

Neural networks (NN)

NN is a network of interrelated components that accepts input, actuates, and then forwards it to the next layer. The NN can be connected in several ways, but in this paper, the Multilayer Perceptron (MLP) for the neural network was adopted. The MLP is a supervised ML algorithm that studies a function $f(.) : R^D \rightarrow R^o$, by training on a dataset (DS), where (D) represents the dimension of the input DS, and o represents the number of dimensions of

expected output. Given *X* set of features, and a target *y*, where $X = \{x_1, x_2, x_3, \ldots, x_D\}$, the MLP can learn a non-linear function approximator for both regression and classification. MLP trains using Adam, Limited- Memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS) or Stochastic Gradient Descent. However, for this study, the Tikhonov regulariser [53], and Adam (Adaptive Moment Estimation) optimiser were adopted. The logistic sigmoid activation function (Eq. 7) was adopted as an activation function in each layer. The mapping-functions for individual layer *l*, are given as expressed in Eq. (6). The backpropagation algorithm was used in training the MLP in this study.

$$Z^1 = W^{[l]T} \times a^{[l-1]} + b^{[l]}$$

(6)

$$g(x) = \frac{1}{1 + e^{-x}}$$

(7)

where $W^{[l]}$ and $b^{[l]}$ represents the weight matrix and bias respectively x is the sum of the weighted inputs.

Ensemble methods (EMs)

Ensemble methods are prevalent in machine learning and statistics. EMs offers techniques to merge multiple single classifiers or predictors to form a committee, to achieve amassed decision for better and accurate results than any of the single or base predictors [54, 55]. Thus, EMs highlights the strong point and watered-down the feebleness of the single classifiers [54, 55]. Two types of ensemble methods are defined by Opitz and Maclin [55], namely:

cooperative and competitive ensemble classifiers. Ensemble involves training diverse single classifiers independently with the same or different dataset, but not with the same parameters. Then, the final prediction (expected output) is obtained by finding an average of all individual single or base classifier output (or other similarities). Whiles the cooperative ensemble is a divide and conquers based approach. The prediction task is subdivided into two or more tasks, where each subtask is sent to the appropriate single classifier based on the characteristics and nature of the subtasks, and the final prediction output is obtained by the sum of all distinct single or base classifiers. In the creation of ensemble classifier and regresses models, three factors need careful consideration. (1) The availability of numerous classification and regression methods makes it difficult to identify which one of them is suitable for the application domain. (2) The number of single classifiers or regressors to assembled for better and higher accuracy. (3) The amalgamation techniques are suitable for combining the outcomes (outputs) of the various single classifiers and regressor to obtain the final prediction or output. We present a brief discussion of some basic and advanced combination techniques for EL in the subsequent section.

Basic ensemble techniques

In this section, we discuss 3 basic but powerful ensemble methods, namely: (i) Weighted averaging (WA) (ii) Max voting (MV) (iii) Averaging.

Max voting (MV)

The primary application of MV is for a classification task. In the MV technique, several single classifier models are employed to decide on every data-point. The output of every individual or single classifier is taken as a 'vote', the final output (decision) is based on the majority's answer. Let M_1, M_1 and M_3 represent single different classifier models, and x_train and y_train be training datasets, independent and dependent variables respectively. While x_test and y_test be independent variables and target variables of the testing dataset, respectively. Let M_1, M_2 and M_3 be trained separately with the same training dataset, thus,
$M_1 . fit (x_{train}, y_{train}) , M_2 . fit (x_{train}, y_{train})$ and $M_3 . fit (x_{train}, y_{train})$, respectively. Let $\widehat{y_{m1}}, \widehat{y_{m2}}$ and $\widehat{y_{m3}}$, represent the predicted output of the respective models. Then, the final prediction (Fp) is a simple majority vote among the predicted output.

Averaging

The averaging technique is very similar to the MV technique; however, an average of the outputs of all individual or single classifiers represents the final output (decision). However, unlike the MV, the averaging technique can be used for both regression and classification machine learning task. With models {M_1, M_2 and M_3} separately trained and tested with the same dataset, final prediction (Fp) is the average of individual models, as expressed in Eq. ([8](#)). where $\widehat{y_1}, \widehat{y_2}, \ldots, \widehat{y_n}$ are the predicted output of individual models.

$$F_p = \sum_{i=1}^{n} \left(\frac{\widehat{y_1} + \widehat{y_2}, \ldots, \widehat{y_n}}{n} \right)$$

(8)

Weighted average (WA)

The WA is an extension of the averaging techniques. In WA technique, different weights are assigned to every model signifying the prominence of an individual model for prediction. However, with WA, M_1, M_1 and M_3 are assigned with different weights of say (0.5, 0.2 and 0.7) respectively, then, the final prediction (Fp) given as Eq. (9).

$$F_p = ((0.5 \times \widehat{y_1}) + (0.2 \times \widehat{y_2}) +, \ldots, + (0.7 \times \widehat{y_n}))$$

(9)

Advanced EL techniques

The following section discusses three advanced combination techniques in brief.

Stacking (STK)

Stacking is an EL technique that makes use of predictions from several models $(m_1, m_2, \ldots, m_n)$ to construct a new model, where the new model is employed for making predictions on the test dataset. STK seeks to increase the predictive power of a classifier [16]. The basic idea of STK is to "stack" the predictions of $(m_1, m_2, \ldots, m_n)$ by a linear combination of weights $a_j, \ldots, (i = 1, \ldots, n)$ as expressed in Eq. (10) [16]. The mlens library [56] was used to implement the stacked EL technique in this study.

$$f_{STK}(x) = \sum_{i=1}^{n} a_i f_i(x)$$

(10)

where the weight vector "a" is learned by a meta-learner.

Blending (BLD)

The blending ensemble approach is like stacking technique. The only difference is that, while stacking uses test dataset for prediction blending uses a holdout (validation) dataset from the training dataset to make predictions. That is predictions take place on only the validation dataset from the training dataset. The outcome of the predicted dataset and validation dataset is used for building the final model for predictions on the test dataset.

Bagging (BAG)

Bagging also called bootstrap aggregating involves combining the outcome of several models (for instance, N number of K-NNs) to acquire a generalised outcome. Bagging employs bootstrapping-sampling techniques to create numerous subsets (bags) of the original train dataset with replacement. The bags created by the bagging techniques severs as an avenue for the bagging technique to obtain a non-discriminatory idea of the sharing (complete set) [48]. The bags' sizes are lesser than the original dataset. Some machine learning algorithms that use the bagging techniques are bagging meta-estimator and random forest. BAG seeks to decrease the variance of models.

Boosting (BOT)

Boosting also called "meta-algorithm" is a chronological or sequential process, where each successive model tries to remedy or correct the errors of the preceding model. Here, every successive model depends on the preceding model [57]. A BOT algorithm seeks to decrease the model's bias. Hence, the boosting techniques lump together several weak-learners to form a strong leaner. However, the single models might not achieve better accuracy of the entire dataset; they perform well for some fragment of the dataset. Therefore, each of the single models substantially improves (boosts) the performance of the ensemble. Some commonly boosting algorithms are AdaBoost, GBM, XGBM, Light GBM and CatBoost.

Study framework

Figure 1 shows the study framework. We adopted STK, BLD, BAG, and BOT combination methods and used DTs, SVM and NNs algorithms as discussed above. To build 'homogeneous' and 'heterogeneous' ensemble classifiers and regressor for predicting stock price and compare their accuracy and error metrics. The study process, as shown in Fig. 1, is grouped into three-phase, namely: (1) Data preprocessing phase. (2) The building of homogenous and heterogeneous ensemble classifiers and regressor models. (3) Comparing the accuracy and error metrics of models. We discuss in detail each phase in the following section.

Fig. 1
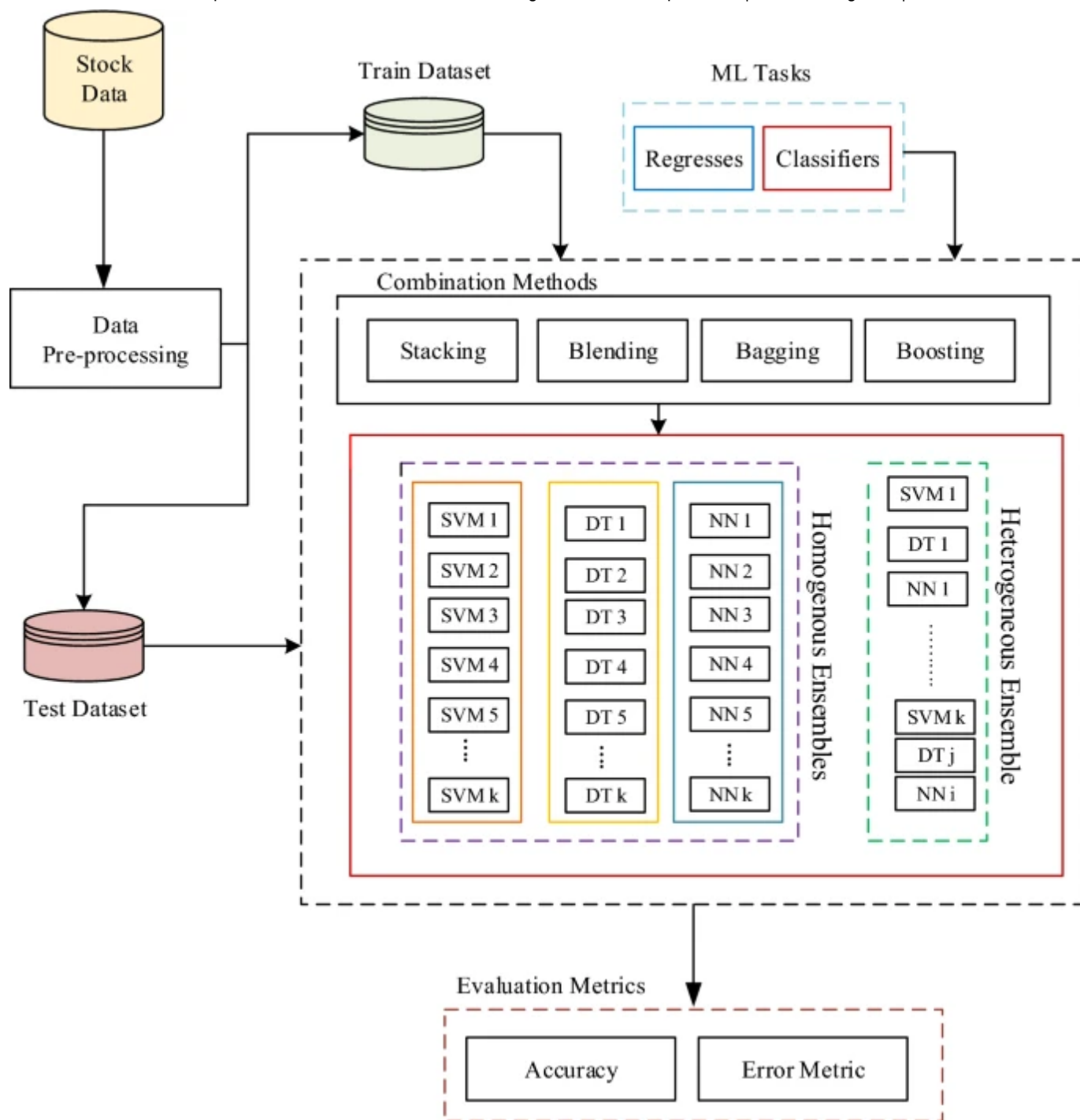
Study Framework

Research data

Market indices were downloaded from the Ghana stock exchange (GSE), the Johannesburg stock exchange (JSE), the New York Stock Exchange (NYSE) and Bombay Stock Exchange (BSE-SENSEX) from

January 2012 to December 2018, to test ensemble methods with datasets from different continents. By doing so, we can verify works that pointed out that some ensemble methods might underperform on datasets from some continents [12, 47]. The datasets consist of daily stock information (year high, year low, previous closing price, opening price, closing price, price change, closing bid price, closing offer). To produce a generalisation of this study, five (5) well-known technical indicators, namely: simple-moving average (SMA), exponential moving average (EMA), Moving average convergence/divergence rules (MACD), relative-strength index (RSI), On-balance-volume (OBV), discussed in [1, 27, 58] were selected and added to some feature from the various dataset. All indicators were calculated from 5 fundamental quantities (opening-price, the highest-price, the lowest-price, closing price, and trading volume). We aimed at predicting a 30-day-ahead closing price and price movement for regression and classification, respectively. The downloaded datasets were preprocessed by going through 2 primary stages, namely: (i) data cleaning, (ii) data transformation.

Data cleaning

The complexity and stochastic nature of stock data make it always prone to noise, which might disturb the ML algorithm from studying the structure and trends in data. The wavelet transform (WT) expressed in Eq. (11) was applied to free the dataset from noise and data inconsistency. We transformed the data X_ω, using WT as follows, remove coefficients (a, b)

with values more than standard deviation (STD). Now we inverse transformed the new coefficients to get our new data free from noise. The WT was used based on its reported ability to adopt and developed the localisation-principle of the short-time Fourier-transform technique, as-well-as features of good-time frequency characteristics and multi-resolution [59].

$$
X_{\omega}(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t)\varphi\left(\frac{t-b}{a}\right) dt
$$

(11)

Data transformation

Machine learning algorithms offer higher accuracy and better error metrics when the input data is scaled within the same range [60]. The min–max normalisation techniques expressed in Eq. (12) guarantees all features will have the same scale [0, 1] as compared with other techniques [61], hence adopted for this study.

$$
b' = \frac{b - b_{\min}}{b_{\max} - b_{\min}}
$$

(12)

where b is the original data value, b' is the value of b after normalisation, $b_{(\max)} and b_{(min)}$ are the maximum and minimum values of the input data.

Empirical analysis and discussion

In our quest to achieve a comprehensive comparative study among ensemble techniques, four (4) different stock-datasets were downloaded from Ghana, South

Africa, United States, and India. Each data had a different number of selected independent variables (features), as shown in Table 2 (Appendix 1). 10-fold cross-validation (10-CV) was adopted and applied in this study to attain an enhanced valuation of training accuracy. With the (10-CV) method, the training set was subdivided into ten subsets of training data, and nine out the ten were used in training each model. Whiles the remaining one (1) was used as test data. This process was repeated ten times, representing the number of folds (10-CV). 80% of each dataset was used for training, whiles the remaining 20% was for testing.

Table 2 Details of dataset

Figure 2 shows the variation between the open and close stock price of the Bombay stock exchange dataset. The graph shows a close range between the opening and closing stock price. We observed that the price of the stock went up in January 2018 as compared with all other years during the period of this study. Figure 3 shows a graph of the open and close stock price of the NYSE dataset. The graph shows little marginal changes between open and closing stock price.
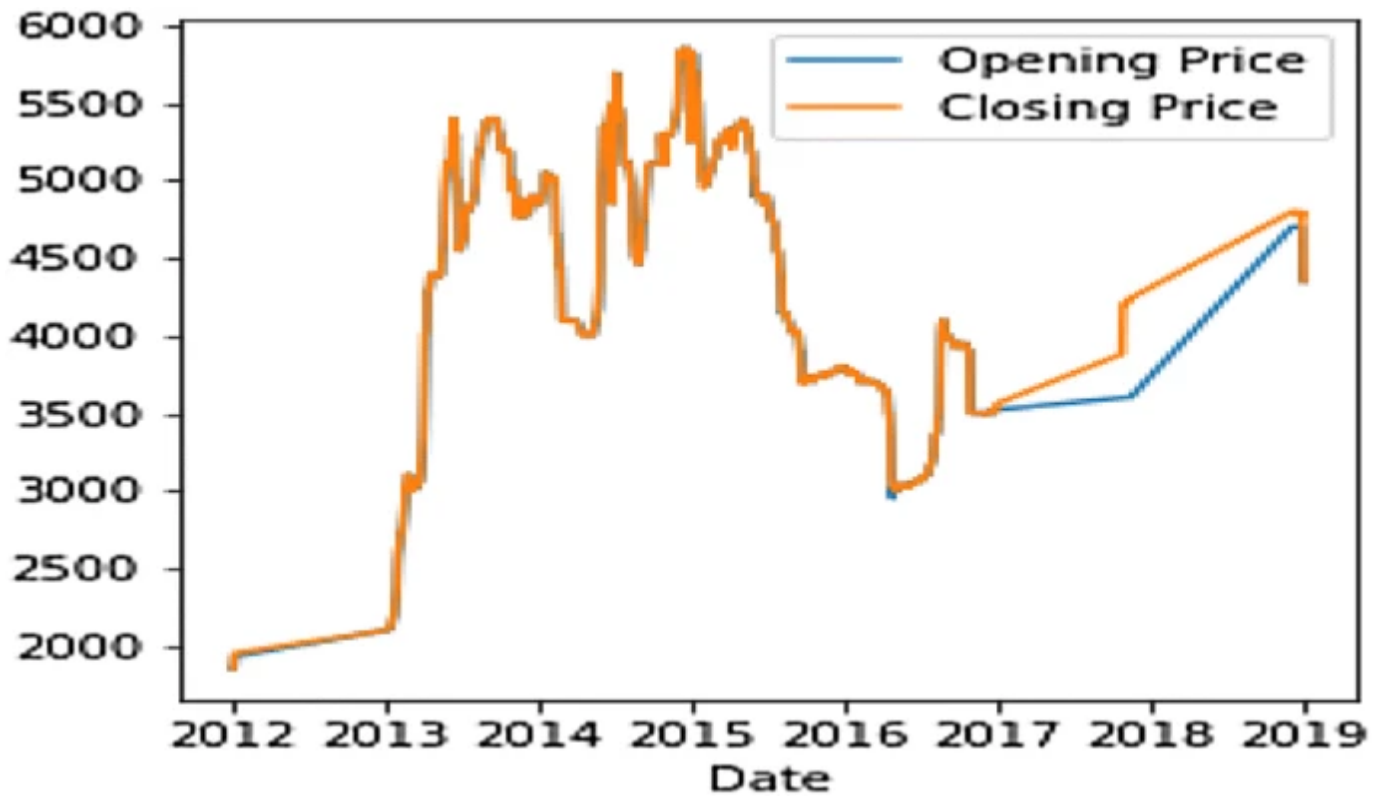
Fig. 2

Bombay stock exchange dataset overview

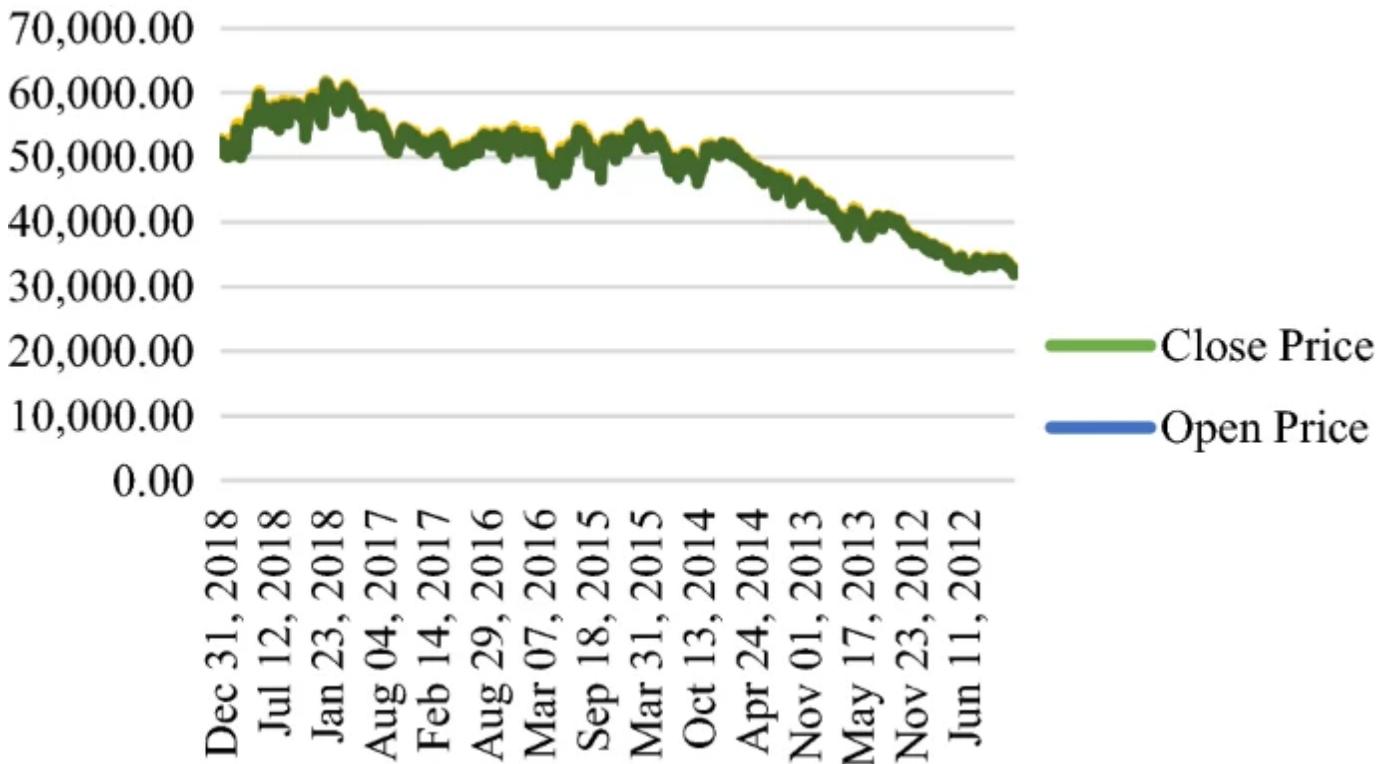**Fig. 3**
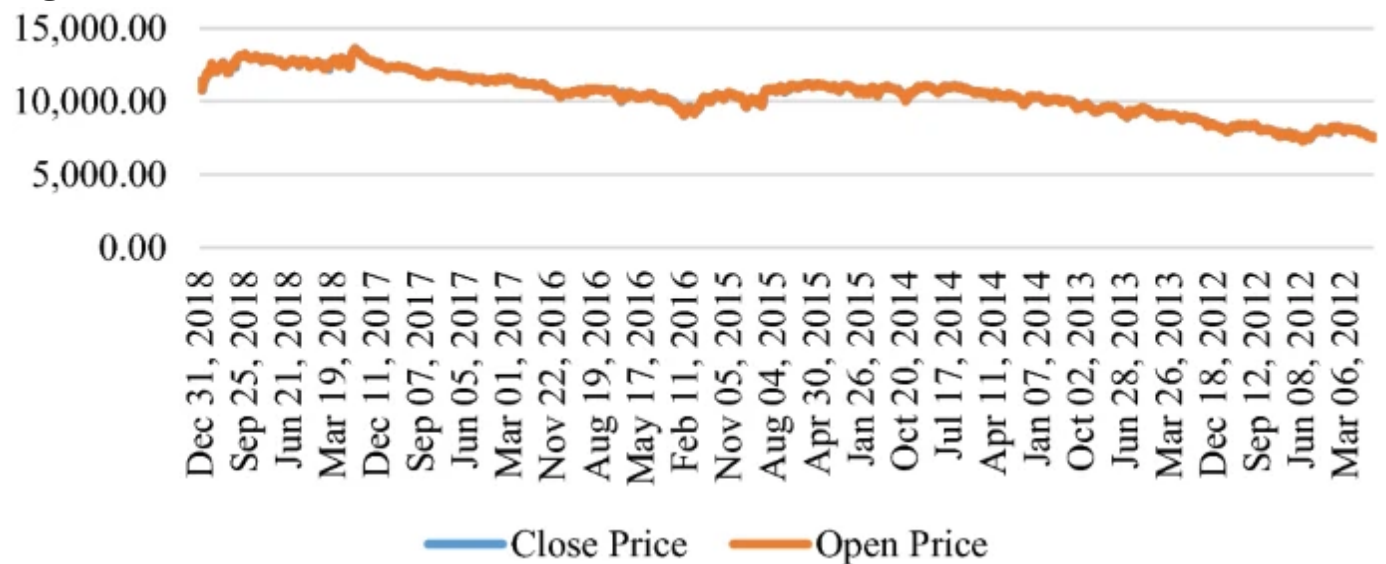
NYSE dataset overview

A graph of the open and close price of the GSE data is as shown in Fig. [4]. A very close variation between open and close is observed in the dataset. Figure [5] shows a plot of the JSE dataset. A graph shows some variation in open and close price.

Fig. 4

GSE dataset overview

Fig. 5

JSE data

Empirical setup

We constructed twelve homogenous ensemble classifiers and regressors based on bagging and boosting combination techniques and thirteen different classifiers, as seen in (Appendix A Table 4) and regressors using stacking, blending and maximum voting combination techniques, as seen in (Appendix A Table 5). Our base leaners parameters were set as follows: MLP three hidden layer (HL), HL1 and HL2 (with five (5) nodes), and HL3 (with ten (10) nodes), the maximum iteration was set to 5000, optimiser = Adam, activation = logistic. For SVM, the Radial Basis Function (RBF) kernel was used, and the regularisation (C) = 100. The DT setting were, criterion = entropy, max_depth = 4. In all, 25 models were built in this study using the Scikit-learn Library, the mlens library [56] and Python. The number of base-leaners was set in a range of [1–200] for "homogeneous" ensemble experiments based on findings in [1] The parameter setting of the SVM and MLP were based on the findings of [12]. An Intel Core i5 64bit with 8 GB memory laptop was used for the implementation of all experiments.

Model evaluation

There are several evaluation metrics available for measuring the performance of classifiers and regressors [1]. However, twelve (12) accuracy and closeness evaluation metrics were selected for evaluating the performance among adopted techniques in this study (see Table 3, Appendix 1). These metrics were selected due to their appropriateness and effectiveness for classification

and regression ML tasks in stock market prediction
[1, 27, 62].

Table 3 Used evaluation metrics

Results and discussion

This section presents the results and discussions of our experimental.

## Homogenous ensembled classifiers by BAG and BOT

The prediction accuracies of the ensemble classifiers over the GSE, BSE, NYSE, and JSE datasets are shown in Figs. 6, 7, 8, 9, respectively, where the x-axis represents the number of base-learners, and the y-axis represents the accuracy of prediction.
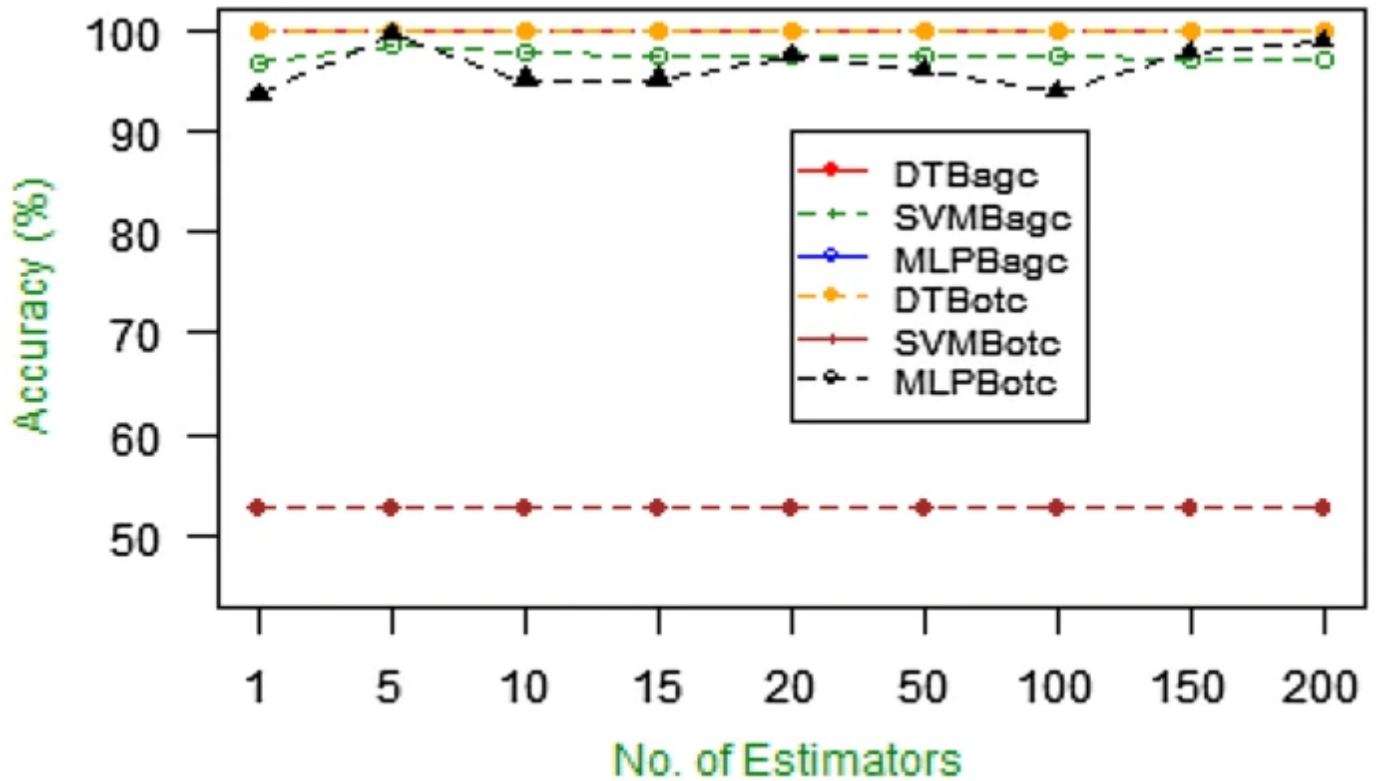
Fig. 6
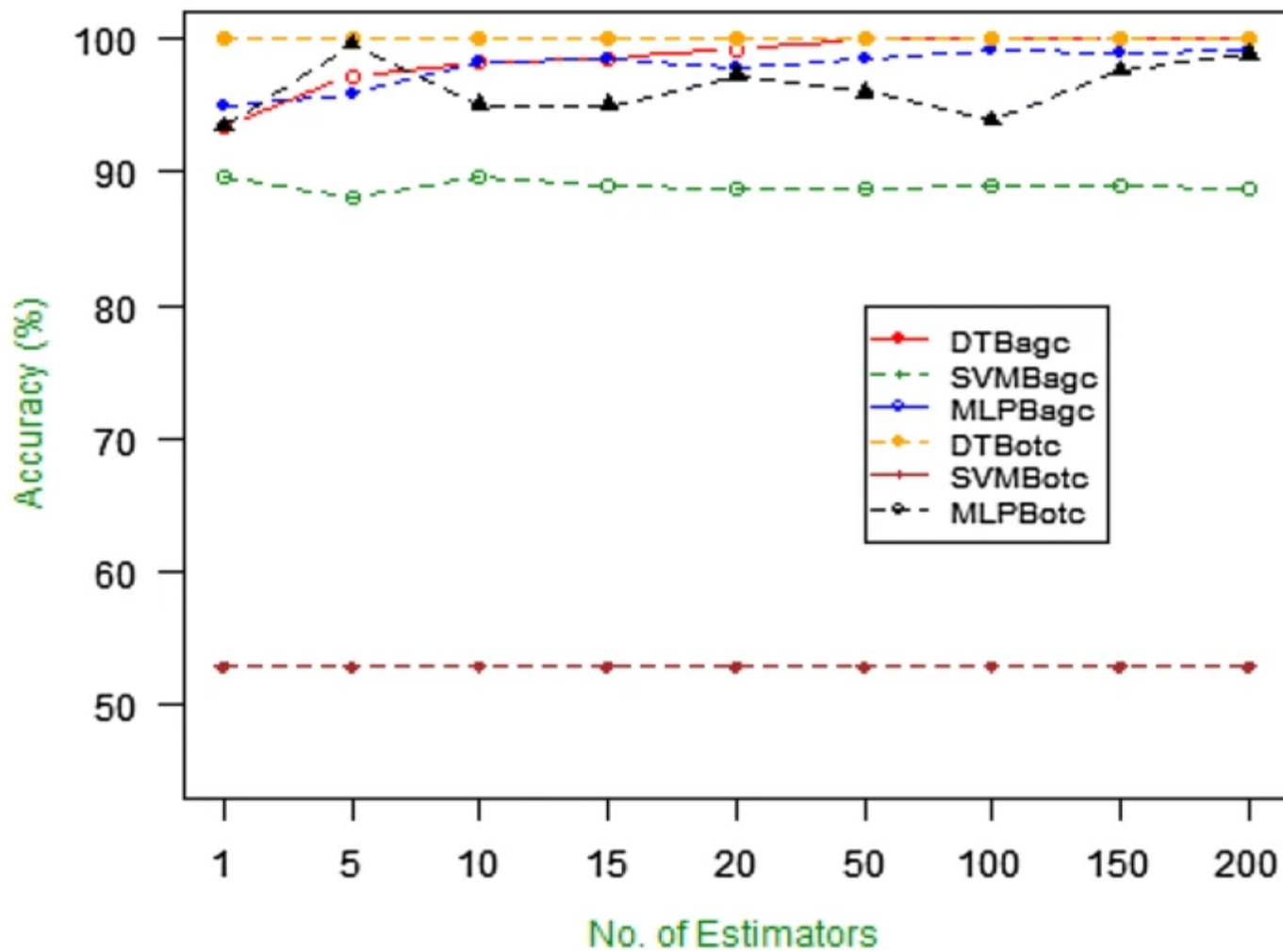
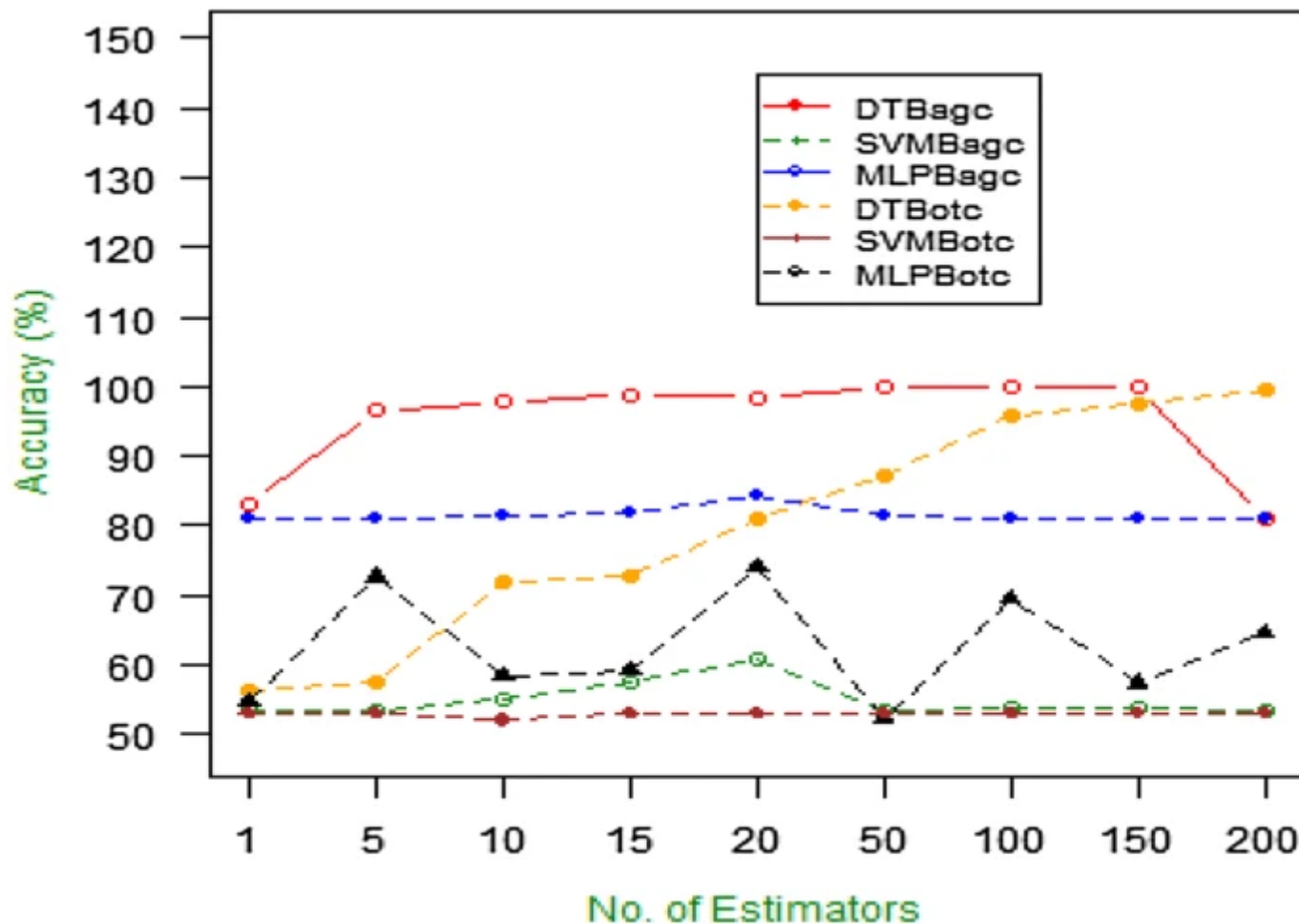Bagging and boosting classifiers accuracy over the GSE dataset

Fig. 7
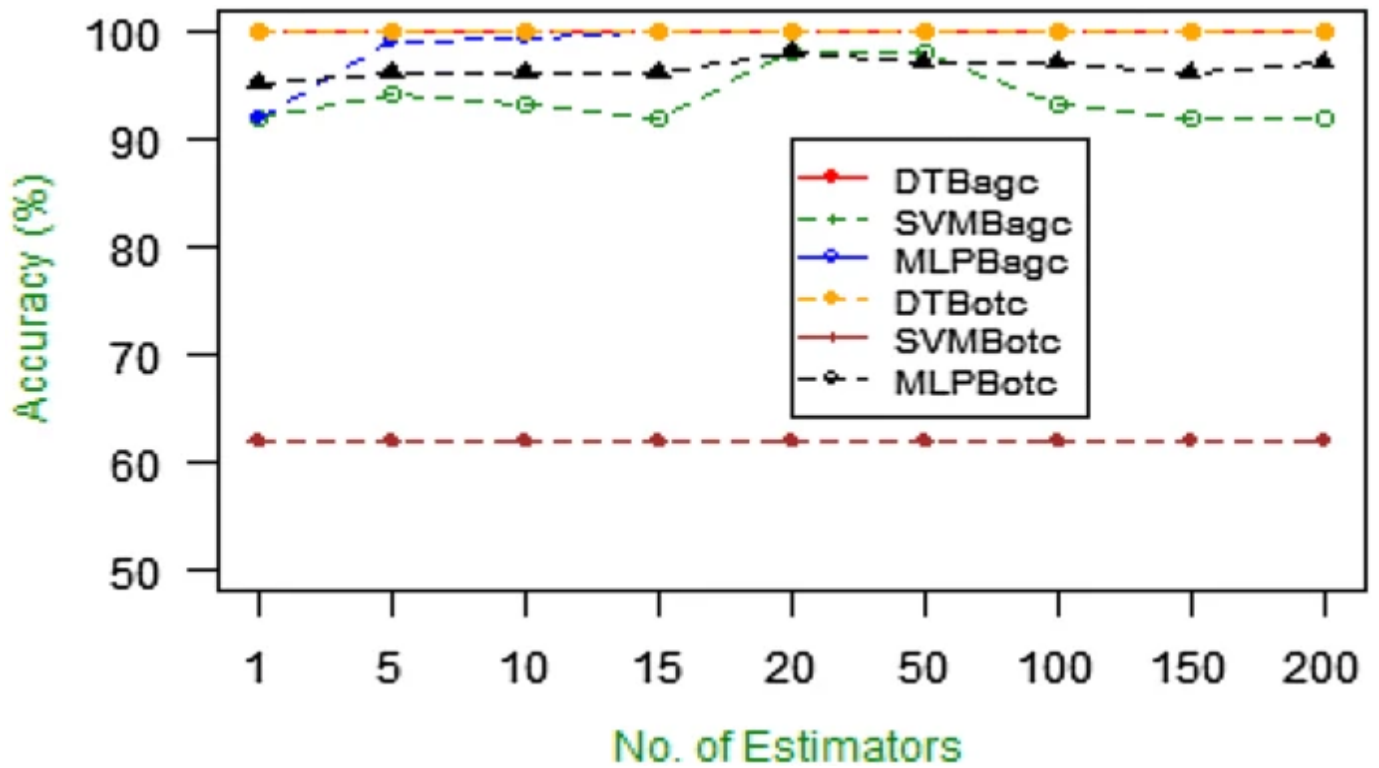
Bagging and boosting classifiers accuracy
over the BSE dataset

Fig. 8

Bagging and boosting classifiers accuracy
over the NYSE dataset

Fig. 9

Bagging and boosting classifiers accuracy over the JSE dataset

We observed that the DT ensemble classifiers by boosting (DTBotc) and bagging (DTBagc) obtain an accuracy of 99.98% with (10–200) estimators over the GSE, BSE, and NYSE dataset (Figs. 6, 7, 8, 9). The accuracy of the MLP ensemble by bagging (MLPBagc) performed better with an accuracy of 100% over the NYSE dataset for estimators from (1 to 200), 94–98% over GSE and 92–100% over BSE, while 80–84% over JSE dataset. The SVM (SVMBagc) recorded 96–97% over NYSE, 53–60% over JSE dataset, 88–89% over GSE dataset.

On an average, the DT ensemble classifier by boosting (DTBotc) recorded an accuracy measure of 100% over NYSE, 95.09% over JSE, and 98.52% over GSE and 99.98 over BSE as shown in Figs. [6], [7], [8] and [9] respectively. While DT ensemble classifier via bagging (DTBagc) obtained an accuracy measure of 100% over the NYSE, 79.95% over JSE, 98.78% over GSE, and 99.93 over BSE. The MLP ensemble classifier by bagging (MLPBagc) obtained an accuracy of 100% over NYSE, 81.53% over JSE, 97.98% over GSE and 98.93 over BSE, while MLP ensemble classifier by boosting (MLPBotc) recorded 96.32% over NYSE, 62.45% over JSE dataset, 88.99% over GSE dataset and 96.45% over BSE dataset. The SVM ensemble classifier by bagging (SVMBagc) recorded an accuracy of 97.43% over NYSE, 54.98% over JSE, 88.87% over GSE and 93.78 over BSE, while SVM ensemble classifier using boosting (SVMBotc) 52.7% over NYSE, 53.02% over JSE dataset, 62.74% over GSE dataset and 62% over BSE dataset.

Likewise, it was observed that the DT ensemble classifiers DTBotc and DTBagc performed very-well over NYSE at 100% accuracy with (1–20) estimators as compared with the JSE, GSE, and BSE. The high accuracy level of the DT ensemble on NYSE might be the fact that the NYSE dataset is the largest (1760) with the highest feature (15) when compared with the rest of all the dataset. This outcome might imply that ensemble classifier performs best with the larger dataset. The MLP ensemble classifier through boosting and bagging did perform well over the NYSE and BSE datasets, as compared with JSE and GSE datasets. On the other hand, the SVM ensemble

employing bagging performed very well over NYSE (97.43%), BSE (93.78%) and GSE (88.87%) but low accuracy on the JSE (54.98%). Overall, the SVM ensemble by boosting recorded low accuracy on all datasets.

Furthermore, we observed that the performance of DT ensemble classifiers (DTBotc and DTBagc) increase with the increase in the number of estimators used. This outcome shows that for a higher and better accuracy measure, the number of estimators for DT ensemble should be high. On the other hand, the accuracy SVM ensemble via boosting and bagging was not directly proportional to the number of estimators. Thus, irrespective of the number of estimators, the accuracy of the SVM was stable. Although the performance of SVM ensemble as compared with DT ensemble and MLP ensemble was low, the outcome obtained implies that when building an ensemble of SVM, the accuracy is independent of the number of estimators. The variation in accuracy by classifier ensembles over different dataset shows that the accuracy of ensemble methods in stock-market prediction is dependent on the origin of the data being analysed, which affirms literature [12, 21].

Error metrics analysis of homogenous ensembled classifiers

Measuring the performance of classifiers and regressors models based on only the accuracy score is not enough for a truthfully [63]. Hence, we further calculated some known error metric. Tables 6, 7, 8, 9, 10, 11, 12 and 13 (Appendix A) shows the error

metrics of DT, SVM, and MLP ensemble classifiers based on boosting and bagging over GSE, BSE, NYSE, and the JSE, respectively. The area-under-curve (AUC) of a DT ensemble classifier by boosting and bagging (DTBotc and DTBagc) falls within (0.920–1) for one estimator to 200 estimators over GSE, BSE and NYSE as shown in Tables 6, 7, 8, 10, 11 and 12 (Appendix A). Hence, confirms the accuracy score obtained by the DT ensembles by boosting and bagging over these datasets shown in Figs. 6, 7, 8 and 9. This finding suggests some skill in the prediction by DT ensembles. On the other hand, AUC measure on JSE dataset falls within 0.5 for one estimator to 0.996 for 200 estimators. The F1-score values of the DT ensemble classifiers (DTBotc and DTBagc) shown in Tables 6, 7, 8, 10, 11 and 12 (Appendix A), shows a balance between recall and precision of the DT ensemble classifiers. Again, the values of RMSE and MAE of DT ensembles by bagging and boosting (DTBagc and DTBotc respectively) are approximately 0.00 from 10 estimators to 200 estimators, which again confirms that the accuracy of DT ensembles is highly dependent on the number of estimators used.

The MLP ensemble classifier by boosting (MLPBotc) recorded AUC values of (0.845–0.847), (0.938–0.965), (0.934–0.938) and (0.523–0.626) over GSE, BSE, NYSE and JSE datasets respectively (Appendix A, Tables 6, 7, 8 and 9). Whiles, MLP ensemble classifier by bagging (MLPBagc) recorded AUC values of (0.943–0.99) (1–1), (1–1), (0.810–0.811) over GSE, BSE, NYSE and JSE datasets respectively for estimators within 1–200 (Appendix A, Tables 10, 11, 12 and 13). The results show that MLP ensemble (MLPBagc)

outperformed (MLPBotc) over GSE, BSE, NYSE and JSE datasets. This implies that an MLP ensemble classifier with bagging outperforms MLP ensemble classifier with boosting for stock-market prediction.

Though the overall performance of SVM ensemble by boosting and bagging is low as compared to DT and MLP ensembles by same combination methods, the AUC, RMSE MAE and recall values of SVM ensemble classifier were more moderate over the BSE dataset with smaller size of 984 than the GSE, BSE and NYSE datasets, as shown in Tables 6, 7, 8, 9, 10, 11, 12 and 13 (Appendix A). This result shows that the classical SVM classifier in its natural form is not suitable for the larger dataset. Except it is enhanced with techniques such as dimensionality reduction.

Homogenous ensembled regressors by BAG and BOT

In other to ascertain the superiority of same machine learning algorithm as an ensemble classifier and regressor, the selected machine learning algorithm DT, MLP and SVM were homogeneously ensemble as regressors using bagging and boosting. Table 14, 15, 16 and 17 (Appendix A) shows the error metrics obtained by the DT, MLP and SVM ensemble regressors over GSE, BSE, NYSE and JSE datasets.

We observed that MLP ensemble regressor by boosting (MLPBotr) and bagging (MLPBagr) offered better accuracy of prediction done DT ensemble regressor by bagging (DTBagr) and bossing (DTBotr) over all datasets. This finding shows that MLP as ensemble regressor is suitable than DT ensemble

regressor by bagging and boosting. Again, no significant difference was seen between MLP ensemble by boosting (MLPBotr), and that of bagging (MLPBagr) as far as the results of this study is a concern as shown in Tables 14, 15, 16 and 17.

The SVM bagged ensemble regressor (SVMBagr) recorded RMSE values of (0.0685–0.0431), (0.0463–0.0465), (0.11–0.071) and (0.010–0.010) over JSE, NYSE, BSE and GSE datasets as shown in Tables 14, 15, 16 and 17 (Appendix A) respectively. While the boosted SVM ensemble regressor (SVMBotr) recorded RMSE values of (0.0681–0.443), (0.0457–0.0455), (0.081–0.056) and (0.010–0.010) over JSE, NYSE, BSE and GSE datasets as shown in (Appendix A, Tables 14, 15, 16 and 17) respectively.

Despite the below-average performance of the SVM ensemble classifier on all datasets, the results of the SVM ensemble regressor by bagging (SVMBagr) and boosting (SVMBotr) obtained better error metrics, which signifies better accuracy levels. The outcome suggests that the SVM is suitable for regression than classification when the dataset is small. Furthermore, the RMSLE and R^2 values of (SVMBotr) compared with (SVMBagr) values as in Tables 14, 15, 16 and 17 (Appendix A), reveals that boosting is more suitable and accurate for SVM ensemble regressors. Subsequently, we observed that the SVM ensemble regressor over GSE outperforms NYSE, BSE and JSE datasets. Once more, this confirms that ensemble techniques, performance is dependents on the origin of the dataset. However, in some cases, the (R^2) of

the SVM was negative, indicating that SVM at this case is worse than predicting the mean.

Furthermore, the training and testing time of bagged and boosted regressors are higher compared with their counterparts (classifiers). On average, the MLP ensemble (regressor and classifier) requires more time for training and testing as the number of estimators and dataset size increases.
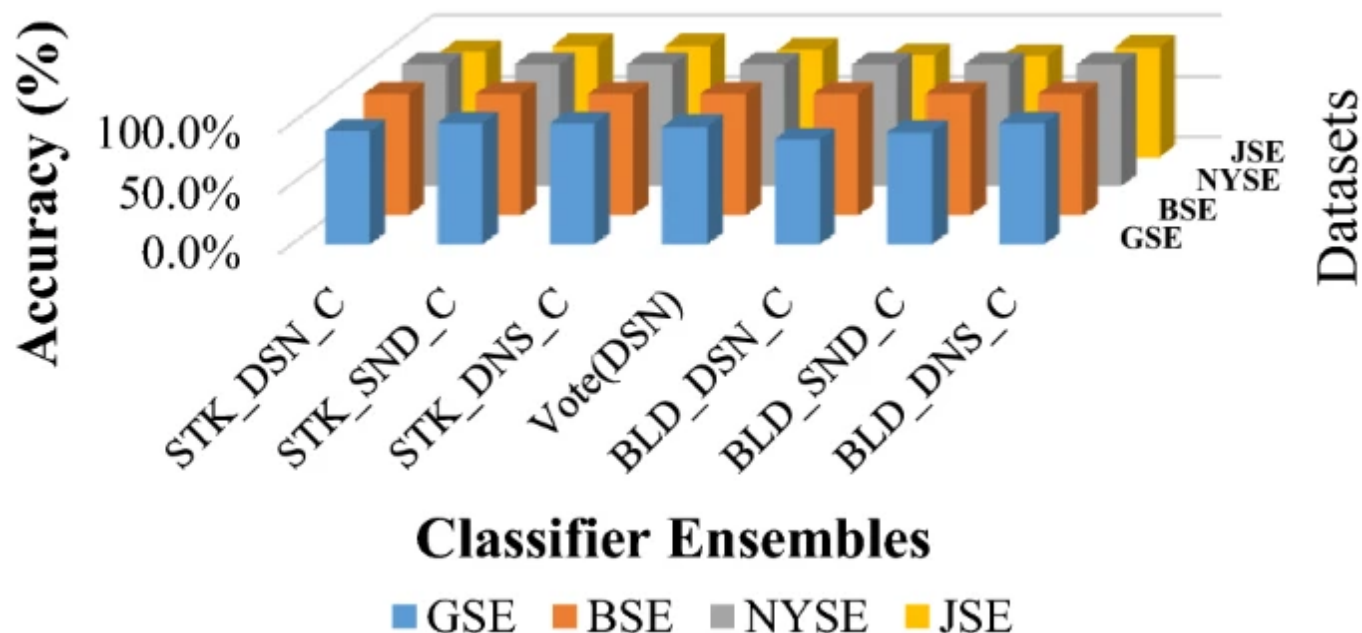
Heterogeneous ensembled classifier by STK and BLD

The section discusses the empirical results of the heterogeneous selected machine learning algorithms (DTs, SVM and NN (MLP)) using stacking, blending and maximum voting combination techniques.

Accuracy measure of heterogeneous ensembled classifier by STK and BLD

Figure 10 shows the accuracy measure of stacked and blended classifier ensemble with DT, NN for (MLP) and SVM classifiers. The stacking ensemble was clustered in three models STK_DSN_C (where DT and SVM were the base learners respectively, and MLP the meta-learner), STK_SND_C (where SVM and MLP were the base learners respectively, and DT the meta-learner), and STK_DNS_C (where DT and MLP were the base learners respectively and SVM the meta-learner). On the same way, the blending ensembles were three, namely: BLD_DSN_C (where DT and SVM were the base learners respectively, and MLP the meta-learner), BLD_SND_C (where SVM and MLP were the base learners respectively, and DT the meta-learner) and BLD_DNS_C (where DT and MLP

were the base learners respectively and SVM the meta-learner). The maximum voting technique was also used to ensemble DT, SVM and MLP with the name vote (DSN).

Fig. 10

Heterogeneous ensembles by stacking and blending

The results (Fig. 10) Shows an average accuracy of 100% over BSE and NYSE datasets, 90% and 98% over JSE and GSE dataset by all stacking ensemble classifiers. However, all blending ensemble classifiers recorded an average of 100% accuracy over BSE and NYSE datasets, but 85.7% and 93.14% over JSE and GSE datasets.

The finding reveals that stacking ensemble classifiers outperforms bagging and boosting ensemble classifier over all datasets and blending over GSE dataset. Despite only two base classifiers and one meta-classifier as compared to 200 base learners for

bagging and boosting, stacking, and blending offered higher accuracy. However, the training time and testing time are far lesser than boosting and bagging of 100–200 estimators that achieved 100% accuracy.

On the other hand, the accuracy obtained by STK_SND_C (100% and 91.5%), STK_DNS_C (100% and 91.5%) and STK_DSN_C (93.4% and 86.3%) over GSE and JSE respectively, has a massive implication on building stacked and blended ensemble. That is, in building stacking and blending ensemble, the choice of base-learners and meta-learner, and how the base learners are position is a significant determinant of the accuracy level of the classifier. This outcome also implies for blending ensemble classifier, as it is evident in Table [18], [19], [20] and [21] (Appendix A) for (BLD_SND_C).

The higher accuracy obtained by stacking and blending ensemble over BSE and NYSE as compared to the JSE and GS shows that ensemble techniques might not perform well on all datasets. Though the maximum voting is a simple ensemble technique *Vote (DSN)*, it showed its ability with better accuracy measure of 97.1%, 100%, 100% and 87.9% over GSE, BSE, NYSE and JSE respectively.

Error metrics analysis of heterogeneous ensembled classifier by STK and BLD

Table [18], [19], [20] and [21] (Appendix A) shows the error metrics of stacking and blending ensemble classifier over BSE, GSE, NYSE and JSE, respectively. The average values 0.9936 (mean), 0.0071 (STD), 0

(RMSE), 0 (MAE) 1 (R^2), 1 (Precision), 1 (Recall) and 1 (AUC) over BSE and NYSE. These values of R^2 reveal that blending and stacking ensemble classifier is good as compared with the naive mean model and are well optimised.

We also observed that the training and testing times of blending classifier ensembles as compared with stacking ensemble classifier overall all datasets were high. However, the accuracy of stacking was higher than blending. The study reveals that the accuracy of ensemble classifiers is not dependent on the time used by the classifier to learn or predict. Again, the cost-efficient of building blended ensemble is high due to higher training and predicting time.

Furthermore, the NYSE dataset was of higher dimension (1760) than the JSE dataset (1749). Nonetheless, the training and predicting the time of blending ensemble classifier over JSE was higher than the NYSE. This result might be due to the noise in the JSE dataset, as shown in Fig. 5.

Tables 22, 23, 24 and 25 (Appendix A), shows the error metrics of ensemble regressors by stacking and blending over BSE, GSE, NYSE and JSE, respectively. The blending and stacking jointly perform well over the NYSE dataset, as shown in Table 24. Oddly, stacking ensemble regressor (STK_DSN_R) outperformed all regressors by stacking and blending over all datasets. Again, this indicates that the selection and position of base learners and the meta-learner is a necessity when building predictive

ensemble model by stacking or blending. The training and prediction time of ensemble classifiers and regressors by stacking and blending were quite higher, compared with ensemble classifiers and regressors through other combination techniques.

To sum-up, stacking combination technique outperformed all other combination techniques for ensemble classifier and regressor. However, the DT ensemble with (10–200) estimators by boosting and bagging did offer good accuracy measure. Though DT ensemble by boosting and bagging offered higher accuracy for stock market prediction, the selection of estimators requires careful assessment. The selection of the base-learner and meta-learner for stacking and boosting ensemble needs careful consideration since the wrong choice can profoundly affect model performance in stock-market prediction.

Furthermore, despite the higher accuracy by DT ensembles by boosting and bagging as compared with MLP and SVM ensembles same combination techniques, the MLP and SVM ensembles were more stable than DT ensembles. Thus, the number of estimators less affected MLP and SVM ensembles. Notwithstanding the number of estimators required by the DT ensemble to offer better accuracy as compared to MLP and SVM ensemble, for stacking and blending ensemble, the computational cost of the DT ensemble is lower. The reason is that the design of MLP, SVM, stacking, and blending ensemble is sophisticated, requiring much time in design.

Conclusion

This paper sought to perform an extensive comparative analysis of ensemble methods such as bagging, boosting, stacking, and blending for stock-market prediction, using stock market indices (dataset) from four countries. Since the performance of ensemble regressors and classifiers based on these techniques for stock market prediction have not wholly been scrutinised in literature. This study attempts to provide answers to the following questions:

1. 1.

   Which of these amalgamation techniques (as bagging, boosting, stacking, and blending) is best suitable for regression and classification tasks in stock market prediction?

2. 2.

   Is the performance of ensemble techniques in stock market prediction associated with the origin of stock data?

3. 3.

   Again, in building ensemble classifiers and regressors, what is the appropriate number of estimators required in building a homogenous ensemble?

To obtain answers to these questions, three well-used machine-learning algorithms, namely; decision trees (DTs), support vector machine (SVM) and a multilayer perceptron (MLP) neural networks, were

employed. Using boosting, bagging, stacking, blending and simple maximum voting combination techniques, we, constructed twenty-five (25) different ensemble regressors and classifiers using DT, MLP and SVM for stock market prediction. We experimented our models on four available public stock-data from GSE, BSE, NYSE and JSE, and compared their accuracy and error metrics. The obtained result revealed that the combination technique (stacking) for building an ensemble classifier or regressor outperformed all other combination techniques like boosting, bagging, blending and simple maximum in stock market prediction. They are followed by blending classifier and regressor ensembles and DT ensembles by boosting and bagging. Again, it was found that stacking and blending though offered high accuracy; they are computationally expensive as compared with DT by boosting and bagging, due to their high training and testing time. For that reason, DT ensemble of 50–100 estimators by boosting can be taken as a classifier baseline for low-cost computation. However, where higher and better accuracy is of vital interest, stacking should be preferred, followed by blending. To the best of our knowledge, this study is the first to carry out a comprehensive evaluation of ensemble techniques (bagging, boosting, stacking, and blending) in a stock market prediction.

Though the SVM ensemble by boosting and bagging was stable, it suffered some deficiencies concerning input variables (input features) and dataset sizes. This defect was overcome when DT and MLP were used as

base-learner respectively, and SVM as meta-learner for stacking and blending ensemble. Thus, the classical SVM algorithm assumes that all the features of samples give the same contribution to the target value, which is not always accurate in several real problems as pointed out by Chen and Hao [58]. Therefore, the practicality of SVM is impacted, due to the problems of choosing suitable parameters of SVM $(C, \sigma \text{ and } \varepsilon)$..

Hence, in future work, some feature selection and SVM parameter optimisation methods such as genetic algorithm (GA), principal component analysis (PCA) can be adapted to assess the effect of carrying-out feature-selection and SVM parameter setting of the classical SVM. Furthermore, we focus on predicting stock market indices, hence we used market indices dataset, where underperforming stocks usually are pulled out from top-line indices and replaced by outperforming stocks to offer market stability. Another focus can be predicting the exact stock prices using ensemble techniques.

Availability of data and materials

The datasets used and/or analysed during the current study are publicly available.

Abbreviations

DT:
 Decision trees

SVM:

Support vector machine

NN:

Neural network

GSE:

Ghana stock exchange

JSE:

Johannesburg stock exchange

NYSE:

York stock exchange

BSE-SENSEX:

Bombay stock exchange

ML:

Machine learning

EL:

Ensemble learning

BAG:

Bagging

BOT:

Boosting

K-NN:

K-nearest neighbour

STK:

Stacking

EMs:

Ensemble methods

WA:

Weighted averaging

MV:

Max voting

BLD:

BLD

SMA:

Simple-moving average

EMA:

Exponential moving average

MACD:

Moving average convergence/divergence rules

RSI:

Relative-strength index

OBV:

On-balance-volume

WT:

Wavelet transform

STD:

standard deviation

MLP:

Multi-layer perceptron

LBFGS:

Broyden–Fletcher–Goldfarb–Shanno

AUC:

Area-under-curve

LR:

Logistic regression

RMSE:

Root mean squared error

MAE:

Mean absolute error

RMSLE:

Root mean squared logarithmic error

MedAE:

Median absolute error

EVS:

Explained variance score

DTBotc:

DT ensembles classifier by boosting

SVMBotc:

SVM ensembles classifier by boosting

MLPBotc:

MLP ensembles classifier by boosting

DTBotc:

DT ensembles classifier by bagging

SVMBagc:

SVM ensembles classifier by bagging

MLPBagc:

MLP ensembles classifier by bagging

DTBotr:

DT ensembles regressor by boosting

SVMBotr:

SVM ensembles regressor by boosting

MLPBotr:

MLP ensembles regressor classifier by boosting

DTBotr:

DT ensembles regressor by bagging

SVMBagr:

SVM ensembles regressor by bagging

MLPBagr:

MLP ensembles regressor by bagging

PCA:

principal component analysis

GA:

Genetic algorithm

BMA:

Bayesian model averaging

WALS:

Weighted-average least squares

LASSO:

Least absolute shrinkage and selection operator

BSE:

Bombay stock exchange

References

1. 1.

   Nti IK, Adekoya AF, Weyori BA. A systematic review of fundamental and technical analysis of stock market predictions. Artif Intell Rev. 2019. https://doi.org/10.1007/s10462-019-09754-z.

2. 2.

   Bousono-Calzon C, Bustarviejo-Munoz J, Aceituno-Aceituno P, Escudero-Garzas JJ. On the economic significance of stock market prediction and the no free lunch theorem. IEEE Access. 2019;7:75177–88. https://doi.org/10.1109/ACCESS.2019.2921092.

3. 3.

   Nti IK, Adekoya AF, Weyori BA. Random forest based feature selection of macroeconomic variables for stock market prediction. Am J Appl Sci. 2019;16:200–12. https://doi.org/10.3844/ajassp.2019.200.212.

4. 4.

Wang Q, Xu W, Huang X, Yang K. Enhancing intraday stock price manipulation detection by leveraging recurrent neural networks with ensemble learning. Neurocomputing. 2019;347:46–58. https://doi.org/10.1016/j.neucom.2019.03.006.

5. 5.

Liu L, Wu J, Li P, Li Q. A social-media-based approach to predicting stock comovement. Expert Syst Appl. 2015;42:3893–901. https://doi.org/10.1016/j.eswa.2014.12.049.

6. 6.

Gupta K. Oil price shocks, competition, and oil and gas stock returns—global evidence. Energy Econ. 2016;57:140–53. https://doi.org/10.1016/j.eneco.2016.04.019.

7. 7.

Billah M, Waheed S, Hanifa A. Stock market prediction using an improved training algorithm of neural network. In: 2016 2nd international conference on electrical, computer and telecommunication engineering. IEEE; 2016. pp. 1–4. http://doi.org/10.1109/ICECTE.2016.7879611.

8. 8.

Kraus M, Feuerriegel S. Decision support from financial disclosures with deep neural networks and transfer learning. Decis Supp Syst.

2017;104:38–48.
https://doi.org/10.1016/j.dss.2017.10.001.

9. 9.

   Pimprikar R, Ramachadran S, Senthilkumar K.
   Use of machine learning algorithms and twitter
   sentiment analysis for stock market prediction.
   Int J Pure Appl Math. 2017;115:521–6.

10. 10.

   Göçken M, Özçalici M, Boru A, Dosdoıru AT.
   Integrating metaheuristics and artificial neural
   networks for improved stock price prediction.
   Expert Syst Appl. 2016;44:320–31.
   https://doi.org/10.1016/j.eswa.2015.09.029.

11. 11.

   Dosdoğru AT, Boru A, Göçken M, Özçalici M,
   Göçken T. Assessment of hybrid artificial neural
   networks and metaheuristics for stock market
   forecasting Ç.Ü. Sos Bilim Enstitüsü Derg.
   2018;24:63–78.

12. 12.

   Ballings M, Van den Poel D, Hespeels N, Gryp R.
   Evaluating multiple classifiers for stock price
   direction prediction. Expert Syst Appl.
   2015;42:7046–56.
   https://doi.org/10.1016/j.eswa.2015.05.013.

13. 13.

   Akyuz AO, Uysal M, Bulbul BA, Uysal MO.
   Ensemble approach for time series analysis in
   demand forecasting: Ensemble learning. In:

2017 IEEE international conference on
innovations in intelligent systems and
applications. IEEE; 2017. pp. 7–12.
https://doi.org/10.1109/inista.2017.8001123.

14. 14.
Bergquist SL, Brooks GA, Keating NL, Landrum
MB, Rose S. Classifying lung cancer severity with
ensemble machine learning in health care claims
data. In: 2nd machine learning for healthcare
conference. 2017. pp. 25–38.

15. 15.
Priya P, Muthaiah U, Balamurugan M. Predicting
yield of the crop using machine learning
algorithm. Int J Eng Sci Res Technol. 2018;7:1–7.

16. 16.
Khairalla MA, Ning X, AL-Jallad NT, El-Faroug
MO. Short-term forecasting for energy
consumption through stacking heterogeneous
ensemble learning model. Energies. 2018;11:1–
21. https://doi.org/10.3390/en11061605.

17. 17.
Zhao Y, Li J, Yu L. A deep learning ensemble
approach for crude oil price forecasting. Energy
Econ. 2017;66:9–16.
https://doi.org/10.1016/j.eneco.2017.05.023.

18. 18.
Macchiarulo A. Predicting and beating the stock
market with machine learning and technical
analysis. J Intern Bank Commer. 2018;23:1–22.

19. 19.

Mabu S, Obayashi M, Kuremoto T. Ensemble learning of rule-based evolutionary algorithm using multi-layer perceptron for supporting decisions in stock trading problems. Appl Soft Comput. 2015;36:357–67. https://doi.org/10.1016/j.asoc.2015.07.020.

20. 20.

Maknickiene N, Lapinskaite I, Maknickas A. Application of ensemble of recurrent neural networks for forecasting of stock market sentiments. Equilib Q J Econ Econ Policy. 2018;13:7–27. https://doi.org/10.24136/eq.2018.001.

21. 21.

Weng B. Application of machine learning techniques for stock market prediction. Auburn: Auburn University; 2017.

22. 22.

Khaidem L, Saha S, Dey SR. Predicting the direction of stock market prices using random forest. Appl Math Financ. 2016;2016:1–20.

23. 23.

Gonzalez TR, Padilha AC, Couto AD. Ensemble system based on genetic algorithm for stock market forecasting. In: 2015 IEEE congress on evolutionary computation. 2015. pp. 3102–8.

24. 24.

Jacobsen B, Jiang F, Zhang H. Ensemble machine learning and stock return predictability. SSRN Electron J. 2018. https://doi.org/10.2139/ssrn.3310289.

25. 25.

Sharma N, Juneja A. Combining of random forest estimates using LSboost for stock market index prediction. In: 2017 2nd international conference for convergence in technology I2CT 2017. 2017. pp. 1199–202. https://doi.org/10.1109/i2ct.2017.8226316.

26. 26.

Sun S, Wei Y, Wang S. AdaBoost-LSTM ensemble learning for financial time series forecasting, lecturer notes computer science (including subseries lecturer notes in artificial intelligence and lecture notes in bioinformatics). 10862 LNCS; 2018. pp. 590–7. https://doi.org/10.1007/978-3-319-93713-7_55.

27. 27.

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. Phys Stat Mech Appl. 2019. https://doi.org/10.1016/j.physa.2019.122272.

28. 28.

Pulido M, Melin P, Castillo O. Particle swarm optimization of ensemble neural networks with fuzzy aggregation for time series prediction of

the Mexican Stock Exchange. Inf Sci (Ny).
2014;342:317–29. https://doi.org/10.1007/978-
3-319-32229-2_23.

29. 29.

Yang B, Gong ZJ, Yang W. Stock market index
prediction using deep neural network ensemble,
in: 2017 36th Chinese control conferrence. IEEE;
2017. pp. 3882–7.
https://doi.org/10.23919/chicc.2017.8027964.

30. 30.

Booth A, Gerding E, Mcgroarty F. Automated
trading with performance weighted random
forests and seasonality. Expert Syst Appl.
2014;41:3651–61.
https://doi.org/10.1016/j.eswa.2013.12.009.

31. 31.

Tan Z, Yan Z, Zhu G. Stock selection with random
forest: an exploitation of excess return in the
Chinese stock market. Heliyon. 2019;5:e02310.
https://doi.org/10.1016/j.heliyon.2019.e02310.

32. 32.

Mathur R, Pathak V, Bandil D. Stock market price
prediction using LSTM RNN. Singapore:
Springer; 2019. https://doi.org/10.1007/978-
981-13-2285-3.

33. 33.

Sohangir S, Wang D, Pomeranets A,
Khoshgoftaar TM. Big Data: deep learning for

financial sentiment analysis. J Big Data. 2018;5:3.
https://doi.org/10.1186/s40537-017-0111-6.

34. 34.

Abe M, Nakayama H. Deep learning for
forecasting stock returns in the cross-section. In:
Phung D, Tseng V, Webb G, Ho B, Ganji M,
Rashidi L, editors. Advanced techniques in
knowledge discovery and data mining. PAKDD
2018 lecture notes in computer science. Cham:
Springer; 2018. p. 273–84.
https://doi.org/10.1007/978-3-319-93034-3_22.

35. 35.

Krauss C, Do XA, Huck N. Deep neural networks,
gradient-boosted trees, random forests:
statistical arbitrage on the S&P 500. Eur J Oper
Res. 2017;259:689–702.
https://doi.org/10.1016/j.ejor.2016.10.031.

36. 36.

Qiu X, Zhu H, Suganthan PN, Amaratunga GAJ.
Stock price forecasting with empirical mode
decomposition based ensemble v-support
vector regression model. In: Mandal J, Dutta P,
Mukhopadhyay S, editors. Computational
intelligence, communications, and business
analytics CICBA 2017. Communications in
computer and information science. Singapore:
Springer; 2017. p. 22–34.
https://doi.org/10.1007/978-981-10-6427-2_2.

37. 37.

Pasupulety U, Abdullah Anees A, Anmol S, Mohan BR. Predicting stock prices using ensemble learning and sentiment analysis. In: Proceedings of IEEE 2nd international conference on artificial intelligence and knowledge engineering. AIKE; 2019. pp. 215–22. https://doi.org/10.1109/aike.2019.00045.

38. 38.
Pulido M, Melin P. Optimization of ensemble neural networks with type-1 and type-2 fuzzy integration for prediction of the Taiwan stock exchange. Stud Fuzziness Soft Comput. 2018;361:151–64. https://doi.org/10.1007/978-3-319-75408-6_13.

39. 39.
Zhu Y, Xie C, Wang GJ, Yan XG. Comparison of individual, ensemble and integrated ensemble machine learning methods to predict China's SME credit risk in supply chain finance. Neural Comput Appl. 2017;28:41–50. https://doi.org/10.1007/s00521-016-2304-x.

40. 40.
Yadav S, Sharma N. Homogenous ensemble of time-series models for indian stock market. Springer. 2018. https://doi.org/10.1007/978-3-030-04780-1_7.

41. 41.
Yang J, Rao R, Hong P, Ding P. Ensemble model for stock price movement trend prediction on different investing periods. In: Proceedings of

12th international conference on computational intelligence in security. CIS 2016. 2017. pp. 358–61. https://doi.org/10.1109/cis.2016.86.

42. 42.

K.S. Gan, K.O. Chin, P. Anthony, S.V. Chang, Homogeneous ensemble feedforward neural network in CIMB stock price forecasting In: Proceedings of international conference on artificial intelligence in engineering and technology IICAIET 2018. 2019, pp. 111–6. https://doi.org/10.1109/iicaiet.2018.8638452.

43. 43.

Jothimani D, Yadav SS. Stock trading decisions using ensemble-based forecasting models: a study of the Indian stock market. J Bank Financ Technol. 2019. https://doi.org/10.1007/s42786-019-00009-7.

44. 44.

Mehta S, Rana P, Singh S, Sharma A, Agarwal P. Ensemble learning approach for enhanced stock prediction. In: 2019 12th international conference on contemporary computing IC3 2019. 2019, pp. 1–5. https://doi.org/10.1109/ic3.2019.8844891.

45. 45.

Assis JDM, Pereira ACM, Silva RCE. Designing financial strategies based on artificial neural networks ensembles for stock markets. In: Proceedings of international joint conference

neural networks. 2018, pp. 1–8.
https://doi.org/10.1109/ijcnn.2018.8489688.

46. 46.

Kohli PPS, Zargar S, Arora S, Gupta P. Stock
prediction using machine learning algorithms.
In: Malik H, Srivastava S, Sood YR, Ahmad A,
editors. Applications of artificial intelligence
technology and engineering advances in
intelligent systems and computing. Singapore:
Springer; 2019. p. 405–14.
https://doi.org/10.1007/978-981-13-1819-1_38.

47. 47.

Kumar M, Thenmozhi M. Forecasting stock
index movement: a comparison of support
vector machines and random forest. In: 9th
Capital Mark Conference Paper, Indian Institute
of Capital Mark. 2006, pp. 1–16.

48. 48.

Tsai CF, Hsu YF, Yen DC. A comparative study of
classifier ensembles for bankruptcy prediction.
Appl Soft Comput J. 2014;24:977–84.
https://doi.org/10.1016/j.asoc.2014.08.047.

49. 49.

Usmani M, Ebrahim M, Adil SH, Raza K.
Predicting market performance with hybrid
model. in: 2018 3rd international conference
emergency of trends engineering science and
technology. IEEE; 2018. pp. 1–4.
https://doi.org/10.1109/iceest.2018.8643327.

50. 50.

Ghosh S, Sadhu S, Biswas S, Sarkar D, Sarkar PP. A comparison between different classifiers for tennis match result. Malays J Comput Sci. 2019;32:97–111.

51. 51.

Akanbi OA, Amiri IS, Fazeldehkordi E. A machine-learning approach to phishing detection and defense. Syngress. 2014. https://doi.org/10.1016/c2014-0-03762-8.

52. 52.

Agarwal P, Bajpai S, Pathak A, Angira R. Stock market price trend forecasting using. Int J Res Appl Sci Eng Technol. 2017;5:1673–6.

53. 53.

Golub GH, Christian PER, Leary DPO. Tikhonov regularization and total least squares. SIAM J Matrix Anal Appl. 1999;21:185–94.

54. 54.

Guzman E, El-halaby M, Bruegge B. Ensemble methods for app review classification: an approach for software evolution. In: 30th IEEE/ACM international conference on software engineering. 2015, pp. 771–6. https://doi.org/10.1109/ase.2015.88.

55. 55.

Ren Y, Suganthan PN, Srikanth N. Ensemble methods for wind and solar power forecasting—a state-of-the-art review. Renew Sustain Energy

Rev. 2015;50:82–91.
https://doi.org/10.1016/j.rser.2015.04.081.

56. 56.

    Flennerhag S. ML-Ensemble. 2017.

57. 57.

    Mayr A, Binder H, Gefeller O, Schmid M. The
    evolution of boosting algorithms from machine
    learning to statistical modelling. Methods Inf
    Med. 2014;53:419–27.

58. 58.

    Chen Y, Hao Y. A feature weighted support
    vector machine and K-nearest neighbor
    algorithm for stock market indices prediction.
    Expert Syst Appl. 2017;80:340–55.
    https://doi.org/10.1016/j.eswa.2017.02.044.

59. 59.

    Shobana T, Umamakeswari A. A review on
    prediction of stock market using various
    methods in the field of data mining. Indian J Sci
    Technol. 2016;9:9–14.
    https://doi.org/10.17485/ijst/2016/v9i48/107985.

60. 60.

    Chong E, Han C, Park FC. Deep learning
    networks for stock market analysis and
    prediction: methodology, data representations,
    and case studies. Expert Syst Appl. 2017;83:187–
    205. https://doi.org/10.1016/j.eswa.2017.04.030.

61. 61.

Academy C. Normalization. 2019.
https://www.codecademy.com/articles/normalization.
Accessed 1 Dec 2019.

62. 62.

Kamel SR, Yaghoubzadeh R, Kheirabadi M.
Improving the performance of support-vector
machine by selecting the best features by Gray
Wolf algorithm to increase the accuracy of
diagnosis of breast cancer. J Big Data. 2019.
https://doi.org/10.1186/s40537-019-0247-7.

63. 63.

Mishra A. Metrics to evaluate your machine
learning algorithm. 2018.
https://towardsdatascience.com/metrics-to-
evaluate-your-machine-learning-algorithm-
f10ba6e38234.

Acknowledgements

Funding

Author information

Affiliations

1. Department of Computer Science and
   Informatics, University of Energy and Natural

Resources, Sunyani, Ghana

Isaac Kofi Nti, Adebayo Felix
Adekoya & Benjamin Asubam Weyori

2. Department of Computer Science, Sunyani
   Technical University, Sunyani, Ghana

   Isaac Kofi Nti

Contributions

IKN obtained the datasets for the research and explored different methods discussed. AFA and BAW contributed to the modification of study objectives and framework. Their rich experience was instrumental in improving our work. All authors contributed to the editing and proofreading. All authors read and approved the final manuscript.

Corresponding author

Correspondence to <u>Isaac Kofi Nti</u>.

Ethics declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Additional information

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Appendix A

Table 4 Homogeneous ensemble classifiers and regressors

Table 5 Heterogeneous ensembles classifiers and regressors

Table 6 Boosting Ensemble Classifiers training time, prediction time and error metrics result on GSE Dataset

Table 7 Boosting ensemble classifiers training time, prediction time and error metrics result on bse dataset

Table 8 Boosting ensemble classifiers accuracy and error metrics result on NYSE dataset

Table 9 Boosting ensemble classifiers training time, prediction time and error metrics result on JSE dataset

Table 10 Bagging ensemble classifiers training time, predicting time and error metrics result on GSE dataset

Table 11 Bagging ensemble classifiers training time, predicting time and error metrics result on BSE dataset

Table 12 Bagging ensemble classifiers accuracy and error metrics result on NYSE dataset

Table 13 Bagging ensemble classifiers training time, predicting time and error metrics result on JSE dataset

Table 14 Bagging and boosting ensemble regressors error metrics result over JSE dataset

Table 15 Bagging and boosting ensemble regressors error metrics result over NYSE dataset

Table 16 Bagging and boosting ensemble regressors error metrics result over BSE dataset

Table 17 Bagging and boosting ensemble regressors error metrics result over GSE dataset

Table 18 Stacking and blending ensemble classifiers error metrics result over BSE dataset

Table 19 Stacking and blending ensemble classifiers error metrics result over GSE dataset

Table 20 Stacking and blending ensemble classifiers error metrics result on NYSE dataset

Table 21 Stacking and blending ensemble classifiers error metrics result on JSE dataset

Table 22 Stacking and blending ensemble regressors error metrics result on BSE dataset

Table 23 Stacking and blending ensemble regressors error metrics result on GSE dataset

Table 24 Stacking and blending ensemble regressors error metrics result on NYSE dataset

Table 25 Stacking and blending ensemble regressors error metrics result on JSE dataset

Rights and permissions

Reprints and Permissions

About this article

Cite this article

Nti, I.K., Adekoya, A.F. & Weyori, B.A. A comprehensive evaluation of ensemble learning for stock-market prediction. *J Big Data* **7,** 20 (2020). https://doi.org/10.1186/s40537-020-00299-5

Share this article

Anyone you share the following link with will be able
to read this content:

Get shareable link

Provided by the Springer Nature SharedIt content-
sharing initiative

Keywords

- Machine-learning
- Ensemble-classifiers
- Artificial intelligence
- Predictions
- Ensemble-regressors
- Stacking
- Blending

- Support and Contact
- Jobs
- Language editing for authors
- Scientific editing for authors
- Leave feedback

- Terms and conditions
- Privacy statement
- Accessibility
- Cookies

Follow
SpringerOpen

SPRINGER NATURE