Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
A novel hybrid approach to forecast crude oil futures using intraday data

Jeevananthan Manickavasagam\textsuperscript{a}, S. Visalakshmi\textsuperscript{b,⁎}, Nicholas Apergis\textsuperscript{c}

\textsuperscript{a}International Management Institute, Kolkata, India
\textsuperscript{b}Department of Management, Central University of Tamil Nadu, Tamil Nadu, India
\textsuperscript{c}School of Business, Law and Social Sciences, University of Derby, Derby, UK

\textbf{ARTICLE INFO}

\textbf{Keywords:}
Crude oil prices
Forecasting
Flower Pollination model
Machine learning model
Particle Swarm Optimization model
Intraday data

\textbf{JEL classification:}
Q31
Q47
C53

\textbf{ABSTRACT}

Prediction of oil prices is an implausible task due to the multifaceted nature of oil markets. This study presents two novel hybrid models to forecast WTI and Brent crude oil prices using combinations of machine learning and nature inspired algorithms. The first approach, MARSplines-IPSO-BPNN, Multivariate Adaptive Regression Splines (MARSplines) find the important variables that affect crude oil prices. Then, the selected variables are fed into an Improved Particle Swarm Optimization (IPSO) method to obtain the best estimates of the parameters of the Backpropagation Neural Network (BPNN). Once these parameters are obtained, the variables are fed into the BPNN model to generate the required forecasts. The second approach, MARSplines-FPA-BPNN, generates the parameters of BPNN through the Flower Pollination Algorithm (FPA). The forecasting ability of these new models is compared to certain benchmark models. The findings document that the MARSplines-FPA-BPNN model performs better than the other competitive models.

1. Introduction

Crude oil is considered as the important source of energy in the world. It is considered as the barometer of global economic trends, while its price fluctuations have a significant impact on economic growth (Zhang et al., 2015). The sale and the profit of the oil business are influenced by crude oil’s price fluctuations and have an effect on capital budgeting decisions (Moshiri and Foroutan, 2006). Crude oil prices clearly indicate whether an economy is in recession or in a booming phase, with the central banks changing their monetary policy stance accordingly (Singh et al., 2019). The prices of crude oil are highly uncertain in the long run (EIA, 2019) as they are inclined by natural disasters, government actions, and demand and supply conditions. There are also other various drivers that determine crude oil prices, such as new oil reserves, weather conditions, and wars. EIA (2019) has estimated the crude oil prices that could rise to $212 per barrel by 2050. In a shorter period of time, it is highly influenced by its lags, non-linearity and price movements in oil markets, such as in Canada, Dubai and Oman (Chen et al., 2016). The short-term crude oil prices fluctuate considerably over the long run (Polanco-Martínez and Abadie, 2016) due to sudden occurrences, such as an explosion in the plant of the Saudi Aramco Company, Coronavirus pandemic and war conflicts in OPEC economies.

Most of the crude oil trades take place in a futures market where prices fluctuate along with the above-mentioned factors. Policymakers, firms and traders/investors are skeptical about a new factor in the market called algorithmic trading. In the short run, this makes the market to fluctuate more than before. The big giants are taking the help of machine learning to forecast future contract prices. Such trades become a problem to the economy, traders/investors, and organizations involved in crude oil markets. Moreover, it becomes difficult to predict prices in the short run (Zhao et al., 2017) due to the factors mentioned above, as well as to the presence of nonlinear patterns and complexity of nature. A precise forecasting can help the stakeholders to minimize losses and make profits in their transactions. Firms that use crude oil as a raw material and traders who are interested in making profits from the commodity markets have to get a good forecasting model to lock the price beforehand. Recent brunt of coronavirus raises doubts about the potential for crude oil demand. The epidemic has already consumed a substantial portion of global oil demand. Hence, it is important for policymakers/companies, firms and traders/investors to forecast crude oil price (Yu et al., 2016). Therefore, the crude oil price forecasting is a serious research study due to its significant role in the process of economic growth.

As there are different crude oil types and grades, benchmarks are used as a reference price. The major benchmarks used in the literature are West Texas Intermediate (WTI), Brent Blend and Dubai crude. Another recognized blend includes the OPEC reference basket. From the

\textsuperscript{⁎} Corresponding author.
\textit{E-mail addresses:} reachjeevananthan@gmail.com (J. Manickavasagam), visa.sr@gmail.com (S. Visalakshmi), n.apergis@derby.ac.uk (N. Apergis).

https://doi.org/10.1016/j.techfore.2020.120126
Received 19 December 2019; Received in revised form 2 April 2020; Accepted 11 May 2020
0040-1625/ © 2020 Elsevier Inc. All rights reserved.
literature, it is known that there are no single models that can predict crude oil prices accurately. Every model has its own pros and cons. Over the last decades, many forecasting models have been developed, tested and recommended (Fan et al., 2016; Chai et al., 2018a). Traditional economic and statistical models, such as the Autoregressive Moving Average (ARMA), Autoregressive Integrated Moving Average (ARIMA), the Generalized Autoregressive Conditional Heteroskedasticity (GARCH) and other GARCH-family models have been employed to predict crude oil price. To use these traditional models, data should be transformed from nonlinear into a linear form or near to linear, from non-stationary to stationary and pretesting for co-integration. These traditional models forecast the values of the converted data, but not the actual data, while the output is of no use in forecasting the actual values of the series explored. Therefore, to overcome the drawback of these models, the literature has recommended machine learning methods that are capable of dealing with noise and nonlinearity problems by itself (Herrera et al., 2019; Chen and Hao (2017) argue that machine learning techniques have ability to handle chaotic and nonlinear data better than the traditional methods. Commonly used machine learning models are those of Backpropagation Neural Network (BPNN), Support Vector Machine (SVM), Random Forest (RF), Fuzzy logic and others (Ahmed and Shabri, 2014; Chen and Hao, 2017; Hamid and Shabri, 2017).

Currently, researchers are building hybrid models by combining two or more models and finding it as a successful process in minimizing the deviation between actual and predicted prices (Azadeh et al., 2015; or more models and finding it as a successful process in minimizing the Shabri, 2017). Henderson and Konstan (2014) argue that machine learning techniques have the ability and predictive efficiency for short-term forecasting. Hou and Siardi (2012) employs GARCH models to characterise the fluctuations of crude oil prices. The authors’ document that the nonparametric GARCH had better results than other GARCH models. In 2012, Wang and Wu, forecasted the volatility of four different energy commodity using GARCH family models. Their results highlight that multivariate models forecasted better than the univariate GARCH models. Nademi and Nademi, (2018) forecast three different crude oil prices using Markov Switching AR-ARCH model. It is found that the proposed model outperformed the compared models namely, ARIMA and GARCH. A study by Gupta and Wohar (2017) forecasts WTI oil prices and S&P500 stock returns using Qualitative Vector Autoregressive (Qual VAR) modelling. The results show that the Qual VAR model outperforms the AR and VAR models. There are a few studies that have used the methods of Self-Exciting Threshold Auto-regressive (SETAR) models (de Albuquerque et al., 2018), and Dynamic Model Averaging (DMA) (Naser, 2016; Drachal, 2016; Wang et al., 2016) in crude oil price forecasting. Such econometric models are capable of forecasting the prices or their volatility under the assumption of linearity and stationarity characteristics in the original data but they are unable to capture the nonlinearity and the complexity present in crude oil prices. Studies (Jianwei et al., 2017; Li et al., 2019) show that the forecasting accuracy of such econometric models is less accurate when compared to machine learning models or any combination of machine learning and standard econometric models.

2. Literature review

2.1. Econometric approaches

The first strand of the literature associated with this paper is the application of traditional/standard econometric models to predict crude oil futures prices. Xiang and Zhuang (2013) analyse the Brent crude oil price using ARIMA modelling. The results demonstrate good prediction effects. A similar study by Zhao and Wang (2014) uses ARIMA modelling to predict the average annual price of world crude oil. The findings reveal that the model has the perfect approach capability and predictive efficiency for short-term forecasting. Hou and Siardi (2012) employs GARCH models to characterise the fluctuations of crude oil prices. The authors’ document that the nonparametric GARCH had better results than other GARCH models. In 2012, Wang and Wu, forecasted the volatility of four different energy commodity using GARCH family models. Their results highlight that multivariate models forecasted better than the univariate GARCH models. Nademi and Nademi, (2018) forecast three different crude oil prices using Markov Switching AR-ARCH model. It is found that the proposed model outperformed the compared models namely, ARIMA and GARCH. A study by Gupta and Wohar (2017) forecasts WTI oil prices and S&P500 stock returns using Qualitative Vector Autoregressive (Qual VAR) modelling. The results show that the Qual VAR model outperforms the AR and VAR models. There are a few studies that have used the methods of Self-Exciting Threshold Auto-regressive (SETAR) models (de Albuquerque et al., 2018), and Dynamic Model Averaging (DMA) (Naser, 2016; Drachal, 2016; Wang et al., 2016) in crude oil price forecasting. Such econometric models are capable of forecasting the prices or their volatility under the assumption of linearity and stationarity characteristics in the original data but they are unable to capture the nonlinearity and the complexity present in crude oil prices. Studies (Jianwei et al., 2017; Li et al., 2019) show that the forecasting accuracy of such econometric models is less accurate when compared to machine learning models or any combination of machine learning and standard econometric models.

2.2. Machine learning models

When compared to traditional prediction tools, machine learning techniques, such as Artificial Neural Networks (ANN) (Movagharnejad et al., 2011; Debnath and Moshoush, 2018), SVM (Xie et al., 2006; Papadimitriou et al., 2014; Fan et al., 2016), and Deep Learning (Zhao et al., 2017) are able to deliver better forecasting accuracy due to their ability of handling nonlinear, non-stationary and complex data structures. However, the application of a single method in forecasting oil prices is affected by the parameter sensitiveness and the overfitting of the data used (Tang et al., 2012). Nowadays, researchers have developed hybrid methods to predict crude oil prices so as to increase the accuracy level of forecasts. Jammazi and Aloui (2012) proposed an integrated technique with Harr A Troux wavelet decomposition (HTW) and BPNN to forecast WTI crude oil spot prices. They apply wavelet analysis to decompose the data series and feed those data series into an ANN process. They find that this hybrid method seems to out best weights for BPNN. The performances of the proposed models are compared with benchmark models. The superiority of the models is statistically checked, while the robustness of the model performance is also checked to explore the efficiency of these models across different time frames.

The rest of the study is structured as follows. Section 2 discusses the current strands of the literature on forecasting crude oil prices. In Section 3, we describe and explain the novel models proposed, while methodological issues are discussed in Section 4. The empirical results and the associated discussions are provided in Section 5, while Section 6 concludes.
outperform benchmark models. Guo et al. (2012) develop a Genetic Algorithm (GA) based on SVM to predict oil prices; their findings provide supportive evidence for this process against the normal SVM approach. Similarly, Zhao et al. (2015) developed VAR-SVM model to predict the crude oil prices. Here, the authors used VAR to find out the crude oil determinants. Then they applied GA to optimize SVM parameters for forecasting the crude oil prices. It was found that the VAR-SVM model was superior in accuracy when compared with other forecasting models. Jianwei et al. (2017) predict crude oil prices by applying the combination of a Variational Mode Decomposition (VMD) and an ARIMA model. Their approach highlights that the VMD-based model outperforms Ensemble Empirical Mode Decomposition (EEMD)-based models. Lahmiri (2015) reaches similar results when applying VMD-based generalized regression neural network (GRNN) ensemble model over Empirical Mode Decomposition (EMD) models to estimate California electricity and Brent crude oil price. Ding (2018) proposes a new hybrid model using Akaike Information Criteria (AIC), EEMD and ANN models. He suggested selection of lag using AIC, decomposing the variable using EEMD and forecast through ANN and ADD (Additional Ensemble Method). His findings demonstrate that the combinations of these methods have a superior performance compared to other models.

To ensure the highest accuracy in forecasting, it is always better to extract the variables which affect oil prices patterns. The normal ANN model can give a better accuracy in forecasting financial time series, if the input variables are carefully selected (Atsalakis and Valavanis, 2009). Futures contracts and spot prices of crude oil are influenced by various determinants, many of which are used as input variable while developing a forecasting model. Thus, it is essential to carefully select the most prominent input, if the machine learning models expect to give an efficient and accurate prediction. There are studies that have used the Principal Component Analysis (PCA) (Ding et al., 2017; Sad et al., 2019) and the Independent Component Analysis (ICA) (Fan et al., 2016; Jianwei et al., 2017; Xian et al., 2020) as a dimension reduction method in the stock market, as well as in crude oil markets.

However, there are a few studies which have used the dimension reduction methods in the course of crude oil price forecasting. Wu et al. (2019) develop a hybrid model using EEMD and Long Short-Term Memory (LSTM) methods. EEMD methods are used to de-noise the data and divide them into different intrinsic mode functions (IMFs). LSTM is applied to extract the necessary IMFs and to forecast crude oil prices. The authors find that selecting relevant IMFs helps to accurately predict crude oil prices against other competitive models. Zou et al. (2020) use VMD to extract the risk factors and model the same with ARMA-GARCH. Furthermore, Convolutional Neural Network (CNN) is employed to assemble the different risk forecast to find out the ensemble forecasted prices. The authors document that their proposed model which reduces the dimensions performs better than other competitive models. Tang et al. (2020) extract information from search engines (search engines data) to predict crude oil prices. They use an advanced version of EMD modelling, called Multivariate EMD (MEMD), to extract the relevant modes, and then BPNN is applied to forecast crude oil prices. Xian et al. (2020) integrate EEMD and ICA to extract the features from the dataset for forecasting WTI crude oil prices.

In particular, certain studies have applied decomposing methods, such as Wavelet (Jammazi and Aloui, 2012; Hamid and Shabri, 2017; Uddin et al., 2019), EMD (Hu et al., 2018; Ding, 2018), and VMD (Jianwei et al., 2017; Lahmiri, 2015; He et al., 2018) to decompose the input data series into subsets of data and then they use optimization methods, namely GA, Artificial Bee Colony (ABC), and Particle Swarm Optimization (PSO) to obtain the optimal parameters of the forecasting methods of ANN and SVM. The recent techniques, namely, EMD is facing mode mixing problems, while the VMD approach has its own problems in setting parameters, leading either to over decomposing the series or to under decomposing the series; this, in turn, implies lower accuracy in the forecasting process (Isham et al., 2019). Though the machine learning methods are efficient in handling nonlinear and nonstationary data, MARSplines methods are capable of finding important input variables for models, such as BPNN, SVM and RF. It can act as a dimension reduction approach, as well as a nonlinear regression method. Therefore, in this paper we propose MARSplines that help in finding the important variable(s) to predict crude oil prices accurately using its nonlinear, irregular and chaotic data handling features. It has been applied as a variable selection tool in stock market studies (Chen et al., 2019; Kao et al., 2013), while no crude oil studies have used this approach to obtain the important variables.

With the machine learning methods, computational algorithms are mixed to find the optimal parameters of the model. Among the computational algorithms, the Particle Swarm Optimization (PSO) approach is highly popular among forecasting studies. This is applied to get the optimal parameters of the ANN, SVR and other Machine Learning modelling approaches. There was perplexity in finding the optimal parameters using normal PSO, while it also ended in local minima (Xiao et al., 2014). Due to this drawback, Alfi and Modares (2011) develop an improved PSO (IPSO) method, called the Adaptive PSO approach, which helps in finding the optimal system and control parameters. Wang and Shoup (2011) develop Poly-hybrid PSO for intelligent parameter adjustment. In this study, we use standard PSO methods by implying crossover and mutation functions of the GA (Xiao et al., 2014) to increase the candidate particle performance. This improved PSO (IPSO) helps to find the best parameters of BPNN. Another computational algorithm called the Flower Pollination Algorithm (FPA), introduced by Yang (2012), has the ability to optimize the parameters globally and thus to increase the performance of forecasting. This FPA approach is improved compared to GA and PSO in resolving issues related to optimization (Yang, 2012). Chiroma et al (2016) develop a hybrid model by combining FPA and BPNN to forecast the OPEC countries petroleum consumption. This model outperforms other hybrid models by producing lower forecast errors. FPA has been also used in forecasting wind speeds (Zhang et al., 2017) and electricity loads (Fan et al., 2019). This optimization algorithm’s performance is yet to be explored across different fields. In this study, we use FPA to optimize neural network parameters so as to build a hybrid model.

Overall, the current literature documents that researchers are trying to forecast crude oil prices for a long time using traditional models, while recently they are using single machine learning methods or a combination of two or more of those methods. Due to changing market and trading scenarios, stakeholders in oil markets must be aware about the availability of new techniques to predict oil prices. At the same time, researchers are focusing on daily, weekly and monthly crude oil price forecasts, but not on intraday futures contract prices, where most of the crude oil trades occur. With high frequency data, incremental forecasting ability can be achieved (Wang et al., 2010; Degiannakis and Filis, 2018). Scant attention has hitherto been provided to study the future markets with very high frequency data. In this study, the analysis investigates the combination of machine learning methods with nature inspired algorithms in forecasting every sixtieth minute crude oil futures contract prices.

3. Methodology of forecasting

3.1. Multivariate adaptive regression spline

MARSplines was developed by Friedman (1991); it is a nonlinear and nonparametric method that does not need any assumption or link between the dependent and the independent variables. This method helps in finding the variables that are affecting the course and pattern of the dependent variable. It works on the basis of the Basis Function (BF) for each independent variable as given below:

$$
\max(0, \ X - v) \text{ or } \max(v - X, 0)
$$

(1)
where, \( v \) is a knot, and \( X \) is an independent variable. The general MARSplines model yields:

\[
f(x_i) = a_0 + \sum_{m=1}^{M} \beta_m B_m(X)
\]

where \( f(x_i) \) is the forecasted price, \( a_0 \) is a constant, \( \beta_m \) is the parameter coefficient of \( m \)th BF, \( B_m \) is \( m \)th BF, \( M \) is number of BF terms. The best model is selected using the Generalized Cross Validation (GCV) criterion. It is a goodness of fit test of the MARSplines model which helps to select the optimal independent variables. The best model is the result of two state processes in MARSplines: (i) State 1: while building the model, the MARSplines considers more numbers of BFs using all possible combinations of links among the independent variables so as to overfit the data, (ii) State 2: using the GCV, it removes the least contributing combinations of variables. The GCV is calculated as follows:

\[
GCV = \frac{MSE_{train}}{\left( 1 - \left( \frac{\epsilon_p}{n} \right) \right)^2}
\]

where \( MSE_{train} \) is the Mean Square Error of training data, \( \epsilon_p \) is the effective number of parameters, and \( n \) is the number of observations.

### 3.2. Backpropagation neural network

There are various types of ANN, mostly to predict the time series data. In our case, the BPNN approach is used. It has the ability to handle nonlinear and chaotic behaviors of the data through the employment of activation functions and repeatedly revising the weight of the nodes. It has input, hidden and the output layers with neurons. The predictor variables are fed towards the input layer (i.e., a neuron for every independent variable), it is processed using a tan sigmoid activation function and, then the weights are updated and sent to the next layer called the hidden layer (i.e., multiple neurons). This process repeats in the middle layer and the updated weights are sent to the output layer (i.e., single neuron).

In the final layer, the outcome value, as well as the actual value of crude oil prices are checked using the Mean Square Error (MSE) criterion. If too much deviations occur, then the last layer neuron returns the error to the hidden layer neurons and the hidden layer neurons, in turn, returns the errors to the initial layer so as to re-calculate the weights. This procedure continues till the output layer provides the least difference between the actual and the outcome value.

The basic BPNN structure can be seen in Fig. 1. The forecasting equation (Eq. (1)) of BPNN is:

\[
T = f(wd + u)
\]

where \( T \) is the Output, \( w \) represents the Weight of the neuron, and \( u \) represents the error.

![BPNN Architecture](image)

**Fig. 1.** BPNN architecture with 2-3-1 network topology.

### 3.3. Optimization methods

#### 3.3.1. Particle Swarm Optimization (PSO)

The PSO was developed by Eberhart and Kennedy in 1995. The idea was generated from flock and swarm behaviors by birds, fishes, and insects for food. The PSO helps to optimize the parameters of ANNs. It is considered better than the GA in terms of the optimization process.

In a normal PSO algorithm, let us assume that a search space has \( m \) dimensions, while the swarm particles can be denoted by a \( n \) vectors \( A_t = (a_{t1}, a_{t2}, ..., a_{tm})^T \). The objective function of the original optimization problem can evaluate particle fitness. Every particle velocity is denoted by \( n \) dimensional vectors \( B_t = (b_{t1}, b_{t2}, ..., b_{tm})^T \). The specific best position of \( t \)th particle is mentioned as \( P_t = (p_{t1}, p_{t2}, ..., p_{tm})^T \) and the global best position as \( G_t = (g_{t1}, g_{t2}, ..., g_{tm})^T \). The new particle’s velocity is expressed as:

\[
b_{j(t+1)} = \omega b_{j(t)} + c_1 r_1 [p_{j(t)} - a_{j(t)}] + c_2 r_2 [g_{j(t)} - a_{j(t)}]
\]

where \( c_1 \) and \( c_2 \) are considered as acceleration parameters, which are self-thinking of particle and group, respectively, \( \omega \) is the inertia weight, and \( r_1 \) and \( r_2 \) are random numbers (0 to 1). Every dimension of particle velocities are fixed to a maximum velocity: \( b_{max} \). The particles new spot is find by Eq. (5).

Improved PSO (IPSO): The general PSO can produce local minima and fails to attain optimal solutions. There are other PSOs which were built to overcome these drawbacks, but they were not able to do it efficiently. Hence, Xiao et al. (2014) develop an improved PSO by incorporating GA in it to solve the local minima and convergence problems.

In stage 1, to improve the balance of the acceleration parameters \( c_1 \) and \( c_2 \), dynamic acceleration parameter adjustment strategies are used and these parameters are controlled using increasing and decreasing arcosine functions. This strategy is followed to know the most suitable solution in the initial stages of the optimization process and to overcome the drawbacks of local minima and parameter convergence in the last stages of the optimization process. The right selection of \( w, c_1 \) and \( c_2 \) provides stability among global and local search (Engelbrecht, 2005).

This is represented as:

\[
c_1 = c_{start} + (c_{end} - c_{start}) \times \left[ 1 - \arccos \left( \frac{-\text{epoch}}{\text{epoch}_{\text{max}}} + 1 \right) / \pi \right]
\]

\[
c_2 = c_{start} c_1
\]

where \( c_{start} \) and \( c_{end} \) are the initial and final values of acceleration parameters’ initial values, respectively, \( \text{epoch} \) is the present epoch number, and \( \text{epoch}_{\text{max}} \) is the maximum epoch number. Appropriate inertia weights and acceleration parameters makes the balance between local and global search. Thus, based on the fitness value of the particles, adaptive nonlinear adjustment inertia weight strategies are used in this study. A particular strategy yields:

\[
w = \begin{cases} 
\frac{w_{min} + \left( \frac{w_{max} - w_{min}}{\text{fit}_{\text{avg}} - \text{fit}_{\text{min}}} \right) (\text{fit} - \text{fit}_{\text{min}})}{\text{fit}_{\text{max}} - \text{fit}_{\text{min}}}, & (\text{fit} \leq \text{fit}_{\text{avg}}) \\
\frac{w_{max}}{\text{fit}_{\text{avg}}} - \text{fit}_{\text{max}}, & (\text{fit} > \text{fit}_{\text{avg}}) 
\end{cases}
\]

where \( w_{min} \) and \( w_{max} \) are the minimum and maximum \( w \), \( \text{fit} \) is a current fitness value, \( \text{fit}_{\text{min}} \) and \( \text{fit}_{\text{avg}} \) are minimum and average fitness values of particles, respectively. From Eq. (10), we can infer that the \( w \) will rise when the particles’ fitness values are stable and it will come down when the values are scattered. Therefore, to preserve their properties, the inertia weights of the greater particles whose fitness values are higher than the mean fitness value that are lower. In comparison, the inertia weights of weak particles whose fitness values are lower than the mean...
fitness value that are greater in order to search for better space.

In stage 2, the Adaptive Genetic Operators (AGO) process is applied to increase the candidate’s particle performance. Based on certain probabilities, the particles will implement crossover and mutation operations. The important operator in GA is crossover; two-point crossover is used in this study which develops offspring by mixing the sections of the parent genome arbitrarily. The crossover rate is planned according to the epochs and not to the discrete fitness. The rate of crossover is calculated as follows:

\[ P_{c} = P_{c,\text{max}} \times q^{-(T/T')} \quad q \in (2, 10) \]  
\[ P_{c}(t) = \begin{cases} P_{c,\text{min}} \quad & \text{if } P_{c,\text{min}} \leq P_{c}(t) < P_{c,\text{min}} \times L_{c} \leq P_{c,\text{min}} \times \text{max crossover probability,} \\
\end{cases} \]  
\[ \left[ P_{c,\text{min}} < P_{c,\text{max}} \right] \in (0, 1) \]  

where \( p_{c} \) is a calculated variable, \( T \) is the highest epoch number, \( t \) is the present epoch number, \( q \) represents crossover probability’s decreasing coefficient, \( P_{c,\text{min}} \) represent minimum crossover probability, \( P_{c,\text{max}} \) is the maximum crossover probability, and \( p_{c}(t) \) is the crossover probability of \( t \)th iteration.

In order to initiate genetic mutations, fractions of off springs are arbitrarily chosen. Then, the mutation operator arbitrarily selects a domicile from a bit-string and turned over its contents. In order to come to early convergence, a more adaptive crossover rate is applied based on the individual fitness, which is based on Eqs. (14) and (15) below:

\[ P_{m} = \begin{cases} P_{m,\text{min}} + \frac{(P_{m,\text{max}} - P_{m,\text{min}})(f - f_{\text{avg}})}{f_{\text{avg}} - f_{\text{mean}}} \quad & f \leq f_{\text{avg}} \\
(P_{m,\text{max}}, f) > f_{\text{avg}} \end{cases} \]  
\[ \left[ P_{m,\text{min}} < P_{m,\text{max}} \right] \in (0, 1) \]  

where \( f_{\text{max}} \) is the highest fitness value of present population, \( f_{\text{avg}} \) is the mean fitness value of every epoch, \( f \) is the fitness value of present mutation individual, \( P_{m,\text{min}} \) is the lowest mutation probability, and \( P_{m,\text{max}} \) is the highest mutation probability. To increase the efficiency of search, the AGO picks the merits of PSO training speed and global search of GA's. The AGO functions as follows:

\[ CP_{t} = 1 - \frac{1}{1 + \ln n} \quad n = 1, 2, ..., \]  
\[ \text{rand}(0, 1) < CP_{t} \]  

where \( n \) represents the nth epoch number. In every epoch, a number is generated randomly between 0 and 1. When the number is below \( CP_{t} \), the present particle initiates the AGO. The AGO will be implemented at a smaller probability when the premature iterations \( CP_{t} < 1 \), according to Eq. (16) and vice versa. During the course of iterations, the AGO increases population search space shrinking to make the particles move out from an earlier optimal search to larger space search. By retaining the multifariousness of the population by particles leads to intensifying the prospects of finding good solutions.

PSO-BPNN: The disadvantage of neural network is the manual parameter feeding and confining to local minima. These demerits can be removed by incorporating PSO-GA into the BPNN to optimize the weights (\( w \)) and biases (\( b \)) (Xiao et al., 2014). The weights and biases are optimized by using Decimal PSO (DPSO), while the structure of the BPNN is optimized using Binary PSO (BPSO). Let the count of nodes in the neural network layers are denoted as \( L \) (Input), \( H \) (Hidden) and \( O \) (Output). The weight and the error terms are referred as:

\[ V = [w_{01}^{1}, ... , w_{LH}^{1}, w_{11}^{2}, ... , w_{21}^{2}, ... , w_{L1}^{2}, b_{1}^{1}, b_{1}^{2}, b_{H}^{1}, w_{1H}^{1}, w_{2H}^{1}, ... , w_{1O}^{2}, b_{2}^{1}, ... , b_{O}^{2}] \]  

where \( w_{ij}^{kl} \) (\( i = 1, 2, ... , L; j = 1, 2, ... , H \)) denote the initial layer to the hidden layer vector's weight, \( w_{ji}^{kl} \) (\( i = 1, 2, ... , H; j = 1, 2, ... , O \)) denote the hidden layer to the final layer vector's weight, \( b_{ij}^{l} \) (\( i = 1, 2, ... , H \)) denoting bias vector of hidden layer in the BPNN, \( b_{j}^{O} \) (\( i = 1, 2, ... , S \)) shows the bias vector of the output layer in BPNN and \( d \) is the dimension of vector \( V \).

The binary coded particle using BPSO is denoting the equivalent hidden layer node's presence (1) and absence (0). According to Eq. (5), the particle velocity is updated. Using the state transition probability, the spot of the particle is updated, which depends on the particle velocity. Generally, the nodes in the hidden layer are not more than the sum of twice of the input and output layer nodes. The BPSO formulation is described as follows:

\[ x_{p}(t + 1) = \begin{cases} 0, \rho_{p}(t + 1) > \text{sig} (v_{p}(t + 1)) \\
1, \rho_{p}(t + 1) \leq \text{sig} (v_{p}(t + 1)) \end{cases} \]  
\[ j = 1, 2, ..., h; h = 2 \times (L + H), \quad \rho_{p}(t + 1) \in (0, 1), j \text{ is a random number, sig(.) denotes the sigmoid function, and } h \text{ denotes the vector's dimension. When } x = 1, \text{ it represents that the corresponding nodes in the hidden layer are present, and the weights and bias vector of that node in DPSO is valid, and vice versa.}

3.3.2. Flower Pollination Algorithm

Like PSO, the Flower Pollination Algorithm (FPA) is a nature inspired algorithm, developed by Yang (2012) on the features of the pollination process of flowering plants using the below set of rules:

(a) Biotic pollination and cross-pollination are known to be mechanisms of global pollination with pollinators executing Lévy flights.

(b) Local pollination is seen as self-pollination and abiotic.

(c) Reproduction probability is seen as flower constancy that is resemblance of two flowers related.

(d) The switch probability \( p \in (0, 1) \) is used to regulate global and local pollinations, because the local pollination can have a fraction \( p \) in the entire process due to physically close and other factors like wind.

In the global pollination, the pollens can move to longer distances by the pollinators, like insects and birds. This progression confirms the pollination and reproduction of the fittest solution which is denoted as \( u^{*} \). The equation of flower constancy (first rule) is:

\[ q_{i}^{t+1} = q_{i}^{t} + L(q_{i}^{t} - u^{*}) \]  

where \( q_{i}^{t} \) is the pollen or solution vector \( q_{i} \) at epoch \( t \), and \( u^{*} \) is the present finest solution of all considered solutions at the existing generation/iteration. \( L \) is the strength of the pollination, which is a step size. There are insects which can travel longer distances; to simulate this nature, Lévy flight is used. Hence \( L > 0 \) from the Lévy distribution is valid for greater steps \( s > 0 \). The local pollination given in the second rule and the flower constancy of the flower are mentioned as:

\[ q_{i}^{t+1} = q_{i}^{t} + \text{sp} \left( q_{i}^{t} - q_{i}^{t} \right) \]  

where \( q_{i}^{t} \) and \( q_{i}^{t} \) are solution vectors that are drawn arbitrarily from the solution set. \( \text{sp} \) is a parameter that is derived from a uniform distribution which ranges between 0 and 1. The FPA is recognized for the following: (a) to elude the local landscape and to have a large search space (explore), the insect pollinators trips long distances, and (b) it...
ensures the convergence to the optimal solution (exploit).

3.3.1. FPA-BPNN
The BPNN’s weights and biases are optimized using FPA (Chiroma et al., 2016). The bias in BPNN is represented as the amount of pollen gametes (n). Before initiating the process, the BPNN’s and FPA’s initial parameters have to set due to the sensitivity of the parameter setting in the first step. From the second iteration onwards, the weights and biases are updated using FPA. The output of FPA-BPNN is compared with the given Mean Square Error value (MSE) in every

Fig. 2. Flow chart of FPA-BPNN.
iteration, which is a stopping criterion. If the criterion is not met, again the FPA process starts generating the parameters of BPNN, till it finds the optimal solution. The flow chart (Fig. 2) shows the steps to generate final output by FPA-BPNN process. The average sum of error squares are represented as:

\[
\text{MSE} = \{V^n_1(q), V^n_2(q), V^n_3(q), ..., V^n_n(q)\}
\]

(24)

where the \(V^n_i(q), V^n_2(q), V^n_3(q), ...., V^n_n(q)\) show the mean performance of every iteration. The MSE, which is imitated by pollen, is obtained once the inputs are processed for every pollen population. This results in the pollen gamete \(q_i\) and is calculated through the below Eq. (25):

\[
q_i = \text{Min}\{V^n_1(q), V^n_2(q), V^n_3(q), ..., V^n_n(q)\}
\]

(25)

The balance pollen gametes are considered by the lasting MSE's. Next, with the help of the Lévy flight, a new result \(q^{i+1}\) is obtained for global pollination \(i\) as follows:

\[
q^{i+1} = q^i + sP(q^i - u^i)
\]

(26)

where \(q^i\) is the pollen i or solution vector \(q\) at epoch \(t\), \(u^i\) denotes the present best solution of all solutions at the existing generation/iteration, and \(L\) represents the strength of the pollination. A new result \(q^{i+1}\) local pollination \(i\) is developed:

\[
q^{i+1} = q^i + sP(q^i - q^*_i)
\]

(27)

The remaining pollen gametes \(q_i\) movements towards \(q_i\) are developed through Eq. (28):

\[
Q = \begin{cases} 
q^*_i + sP(q^*_i - q^i) & \text{if } sP > [0, 1] \rho x \\
q^i & \text{otherwise}
\end{cases}
\]

(28)

The pollen gamete examines the travels from \(q_i\) towards \(q_i\) through the Lévy flight process, as given below:

\[
Q_m = \begin{cases} 
\text{Lévy} \lambda \sim \text{sp} \left( \frac{\sqrt{\lambda}}{v_{i-1}} \right) & \text{if } \text{sp} > [0, 1] \rho x \\
q^i & \text{otherwise}
\end{cases}
\]

(29)

where \(Q_m\) is the movement of pollen gamete travels from \(q_i\) towards \(q_i\). The weights \(W^{n+1}_x, W^n_x\) and the biases \(B^{n+1}_x, B^n_x\) of every layer are adjusted based on Eqs. (30) and (31), respectively:

\[
W^{n+1}_x = W^n_x - Q_m
\]

(30)

\[
B^{n+1}_x = B^n_x - Q_m
\]

(31)

4. Data and forecasting methodology

Among the different crude oil markets, most contract trades occur in terms of European Brent prices (Brent) and U.S. West Texas Intermediate prices (WTI). This study focuses on WTI and Brent oil futures contract prices (rollover to the nearby futures contract) from the New York Mercantile Exchange and the Intercontinental Exchange, respectively. Good forecasting models can be generated when good predictor variables are used. To forecast the intraday values of KOSPI200 index, Son et al. (2012) used a list of technical indicators (provided in Table 1) as predictor variables. In this study, to forecast the intraday futures contract prices of WTI and Brent, the list of technical indicators considered by Son et al (2012) is used. During April 2019-Sep 2019, several events took place around the globe, such as the US-China trade war, political tensions in the Middle East, Iranian forces seizing a British oil tanker, talks between India and China on the establishment of an “oil buyer’s club, and attacks on Saudi Arabia’s Aramco Company which led to high oil price fluctuations. Moreover, weekly and monthly prices always subsidized these effects due to time lapses. Hence, this study investigates the intraday forecasting of WTI and Brent Crude oil futures prices, respectively. Every sixthtieth minute, close prices for crude oil (US$ per Barrel) are obtained from Bloomberg for the period of 6 months (April, 1 to Sep, 30, 2019). Missing data are handled using the mean value of that day and if any given day has few data points, that day’s data are not considered. The raw data are converted into a normalized form using min-max normalization (Han et al., 2011) to reduce the forecasting error. Then, the dataset is bifurcated into 80% (training) and 20% (testing) to build and evaluate the models, respectively.

To forecast oil prices, the analysis introduces two hybrid models, namely MARSplines-IPSO-BPNN (PM-1) and MARSplines-FPA-BPNN (PM-2). The performance of these models is compared with benchmark models, such as IPSO-BPNN, FPA-BPNN, MARSplines-BPNN, MARSplines and BPNN. The optimal model is chosen on the basis of the lowest MSE value across all models used in this study, except for the case of MARSplines, where the lowest GCV is used by default. The forecasted values are converted into a raw form to assess the model performance with one another. The statistical measures, namely MSE, Mean Absolute Percentage Error (MAPE), and the Theil U-statistic are used to check the comparison. Moreover, the Diebold Mariano (DM) test (Diebold and Mariano, 1995) is also applied to measure the superiority of the proposed model against other competitive models.

5. Empirical analysis

5.1. Dimension reduction

The technical indicators are fed into MARSplines for dimension reduction. The list of technical indicators used in this study is listed in Table 1. Among these variables, EMA(5), RDP(1), RDP(3), and EDISP (5) are selected by MARSplines for both energy commodity futures price forecasting based on its GCV value (0.000319 and 0.000345 for WTI and Brent prices, respectively).

5.2. MARSplines-IPSO-BPNN

The independent variables selected by MARSplines are fed into BPNN where its parameters are optimized by IPSO. Initially, the parameters of IPSO-BPNN are set as: the size of swarm population = 40, the determined epochs of IPSO = 500, the epochs of BPNN = 2000. Next, in DPSO, the maiden particle positions are set as random numbers between -20 and 20, the maiden particle velocity randomly ranges between -1 and 1, the initial particle velocity ranges between -0.7 and 0.7, \(c_1 = 2.85, c_2 = 1.05, w_{\text{min}} = 0.25,\) and \(w_{\text{max}} = 0.9\). In IPSO, the initial particle positions are set randomly between -1 and 1, the initial particle velocity ranges between -0.7 and 0.7, \(c_1 = 2.15\) and \(w = 1\). In GA, \(P_{c_{\text{min}}} = 0.4, P_{c_{\text{max}}} = 0.95, P_{m_{\text{min}}} = 0.01,\) and \(P_{m_{\text{max}}} = 0.7\). The MSE is set at 2 \(\times 10^{-6}\) or less is reached. The results of the model show that the 9 neurons in hidden layers are used to achieve the least MSE of 0.000016 and 0.000015 for WTI and Brent, respectively (Table 2).

| Table 1 |
| List of predictor variables. |
| Technical Indicator | Variables |
| Moving Average (MA) | MA(5), MA(10), MA(20) |
| Exponential Moving Average (EMA) | EMA(5), EMA(10), EMA(20) |
| Disparsity (DISP) | DISP(5), DISP(10), DISP(20) |
| EMA Disparsity (EDISP) | EDISP(5), EDISP(10), EDISP(20) |
| Relative Strength Index (RSI) | RSI(5), RSI(10), RSI(20) |
| Momentum (MTM) | MTM(5), MTM(10), MTM(20) |
| Relative Difference in Price (RDP) | RDP(1), RDP(2), RDP(3), RDP(5), RDP(10), RDP(20) |
| Price oscillator (OSCP) | OSCP |
| Price oscillator (EOSCP) | EOSCP |
the hidden layer neurons are set to 13. Hence, the basic network topology used is 4-13-1. The weights and the biases of the BPNN are initialized by FPA. The BPNN's initial weights are fed using FPA's weight values at each epoch. The FPA fed's updated weights to BPNN structure in every epoch. The model convergence is set to stop when the MSE reaches $2 \times 10^{-6}$ or maximum epochs reach, otherwise, the BPNN continues to train the model till is satisfied with the stopping conditions. The best model occurs when the MSE reaches the minimum level of 0.000002 and 0.000004 for WTI and Brent, respectively (Table 2).

### 5.4. IPSO-BPNN, FPA-BPNN and MARSplines-BPNN

For IPSO-BPNN and FPA-BPNN, the same parameters are set as given in the sections MARSplines-IPSO-BPNN and MARSplines-FPA-BPNN, respectively. From Table 2, it is observed that the lowest MSEs of IPSO-BPNN for WTI and Brent are 0.000608 and 0.000061, respectively. For the case of FPA-BPNN, the MSEs are 0.000597 and 0.000031 for WTI and Brent, respectively. For MARSplines-BPNN, the variables selected by MARSplines are directly fed into BPNN and through a trial and error basis. The lowest MSEs achieved are 0.000715 and 0.000602 for WTI and Brent, respectively.

### 5.5. Comparison of results

The normalized output (Table 2) is denormalized into a raw form to estimate the models’ performances using statistical measures, namely MSE, MAPE and Theil U-statistics. The comparison results are reported in Table 3. We can see that the error values obtained by using PM-2 are lower than the other novel model PM-1 and the benchmark models under MSE, MAPE and Theil U-statistic for both WTI and Brent crude oil futures prices.

To statistically check the performances of the models generated by the seven modelling methods, the DM test is performed. Tables 4 and 5 report the results of the DM test for WTI and Brent forecasted futures oil prices, respectively. Through the DM test, it is inferred that when the PM-1 and PM-2 are treated as target models, the p-values are lesser than the significance level of 5%, implying that when MARSplines are used with FPA-BPNN and IPSO-BPNN, the forecasted values are superior against those provided by the other models at a minimum of 95% confidence level for WTI and at 90% confidence level for Brent futures contract oil prediction, respectively. For WTI (Table 4), all tested models are significantly better than the comparable models at a minimum of 90% confidence level, except the performances between IPSO-BPNN and FPA-BPNN, where there is no difference in the predicted values even at the 90% level. In contrast, for the case of Brent prices (Table 5), the forecasting performance of IPSO-BPNN versus

### Table 4
Comparison of normalized outputs of WTI and Brent crude oil futures prices.

| Crude oil | Models       | MSE   | MAPE  | Theil U |
|----------|--------------|-------|-------|---------|
| WTI      | PM-1         | 0.000016 | 0.2933 | 0.0050  |
|          | PM-2*        | 0.000002 | 0.0808 | 0.0019  |
|          | IPSO-BPNN    | 0.000608 | 1.7639 | 0.0306  |
|          | FPA-BPNN     | 0.000597 | 1.6917 | 0.0303  |
|          | MARSplines-BPNN | 0.000715 | 1.9082 | 0.0332  |
|          | MARSplines   | 0.002579 | 3.9518 | 0.0636  |
|          | BPNN         | 0.027066 | 1.8964 | 0.0222  |

### Table 3
Comparison of denormalised outputs of WTI and Brent crude oil futures prices.

| Crude oil | Models       | MSE   | MAPE  | Theil U |
|----------|--------------|-------|-------|---------|
| WTI      | PM-1         | 0.004111 | 0.0297 | 0.0005  |
|          | PM-2*        | 0.000621 | 0.0079 | 0.0002  |
|          | IPSO-BPNN    | 0.152395 | 0.1776 | 0.0034  |
|          | FPA-BPNN     | 0.149436 | 0.1699 | 0.0034  |
|          | MARSplines-BPNN | 0.179046 | 0.1924 | 0.0037  |
|          | MARSplines   | 0.645522 | 0.4096 | 0.0071  |
|          | BPNN         | 5.681008 | 1.3029 | 0.0208  |

### Table 2
Comparison of denormalised outputs of WTI and Brent crude oil futures prices.

| Tested model | Compared models | IPSO-BPNN | FPA-BPNN | MARSplines-BPNN | MARSplines | BPNN |
|--------------|-----------------|-----------|----------|-----------------|------------|------|
| PM-1         | 2.1452 (0.0319) | 2.4158 (0.0156) | 2.3281 (0.0199) | 2.3235 (0.0201) | 2.6293 (0.008) | 2.7199 (0.0065) |
| PM-2         | 2.4071 (0.016) | 2.3238 (0.0201) | 2.3198 (0.0203) | 2.6269 (0.0086) | 2.7211 (0.0065) |
| IPSO-BPNN    | 0.6277 (0.5301) | 1.8888 (0.0569) | 2.0357 (0.0417) | 2.6858 (0.0072) | 2.6729 (0.0075) |
| FPA-BPNN     | 0.001180        | 0.0449     | 0.0244    | 0.0916          | 0.3615     |
| MARSplines-BPNN | 0.000578 | 0.4051       | 0.0076    |

* Denotes Least errors.

### 5.3. MARSplines-FPA-BPNN

Based on the initial experiments, the pollen gametes numbers are finalized to 22, $p = 0.11$ (trial and error), dimension = 350 and the iteration = 1000. Similarly, the BPNN input layer's neurons are set equal to the variables selected by applying MARSplines and single node in the final layer. Based on Zhang et al. (2001), the analysis sets one hidden layer for the network and according to Berry and Linoff (1997),
From these Tables it can be inferred that the PM-2 model, i.e. MARSplines-FPA-BPNN, forecast WTI and Brent futures crude oil prices with the least errors compared to the other models. This highlights the importance of using dimension reduction methods in forecasting studies. The next model which is closely forecasting WTI and Brent futures contracts prices was the MARSplines-BPNN. FPA-BPNN performs better than IPSO-BPNN without the application of dimension reduction method, probably because of the presence of demerits of BPNN, its forecasting performance (Table 6 and 7) is lower than other models in this study.

To check the forecasting robustness of the models used, the data sample is split in the ratio 60:40, 70:30, 80:20 (original) and 90:10 to train and test respectively. The robustness evaluation of WTI and Brent's one-step ahead future contract's denormalized predicted values train and test respectively. The robustness evaluation of WTI and Brent futures crude oil prices with the least errors compared to the other models. This highlights the importance of using dimension reduction methods in forecasting studies. The next model which is closely forecasting WTI and Brent futures contracts prices was the MARSplines-BPNN. FPA-BPNN performs better than IPSO-BPNN without the application of dimension reduction method, probably because of the presence of demerits of BPNN, its forecasting performance (Table 6 and 7) is lower than other models in this study.

6. Conclusion

In the current state of globalization, crude oil prices play a vital role in economic growth. Therefore, it is important to forecast these prices to reach better economic and investment decisions. However, forecasting crude oil prices is challenging due to its volatility nature. Though researchers have used statistical methods to forecast oil prices, it still remains a difficult task. Previous literature focuses on nonlinear methods, such as ANN, SVM, and fuzzy logic for forecasting oil prices. Due to the complex dynamic behavior of the crude oil prices, this study
proposed two new hybrid approaches namely, MARSplines-IPSO-BPNN and MARSplines-FPA-BPNN to forecast every sixty sixth minute of WTI and Brent crude oil futures contract prices (rollover). The normalized WTI and Brent close prices were considered as dependent and the normalized technical indicators as the predictor variables. Statistical measures like MSE, MAPE and Theil U-statistics were used to check the forecasting performance of the models.

The proposed models (MARSplines-IPSO-BPNN and MARSplines-FPA-BPNN) are compared with the benchmark models (IPSO-BPNN, FPA-BPNN, MARSplines-BPNN, MARSplines and BPNN) and the results obtained indicate that the MARSplines-FPA-BPNN model of groups outperforms the other models. The forecasting performance of MARSplines-FPA-BPNN is shown to be quite promising than other competitive models, namely IPSO-BPNN and FPA-BPNN developed by Xiao et al. (2014) and Chiroma et al. (2016), respectively. Next to MARSplines-FPA-BPNN, MARSplines-IPSO-BPNN outperforms the compared models. Since the proposed models reduce the input variables using MARSplines, it is easy for the optimization algorithms to optimize the BPNN parameters to get the output. This helps the proposed model to illustrate lowest errors and a better forecasting accuracy when compared to other models. Among the proposed models, it has been identified that the properties of the FPA optimization methods work better than IPSO. Statistically, to ensure the obtained outcome, the DM test was also performed. To confirm the robustness of the model, the dataset was divided into 60:40, 70:30, 80:20 and 90:10; the new findings confirmed that the MARSplines-FPA-BPNN was robust to forecast WTI and Brent crude oil futures prices with lowest MSE, MAPE and Theil U statistic values.

Overall, the results provide solid support to the proposed MARSplines-IPSO-BPNN model, which demonstrates its effectiveness among the models considered. On the whole, this study emphasized the merits of using dimension reduction methods and optimization algorithms to get better results in forecasting crude oil prices. This study coincides with Jammazi and Aloui (2012), Safari and Davallou (2018) and Li et al. (2019) who was supporting the hybrid models over single models in forecasting crude oil prices.

Theoretically, this study contributes to the development of forecasting models through the dimension reduction method to achieve lower prediction errors (Niu et al., 2020). Forecasting the future crude oil prices will impact current prices, as they will trigger trade activities (Ghoddusi et al., 2019). The application of a nonlinear regression model, i.e. MARSplines as a dimension reduction tool, helps to handle the noisy data, determining the important variables by adjusting its splines which helps the hybrid models to make lower prediction errors (Rounaghi et al., 2015; Roy et al., 2018; Chen et al., 2019). This is the first time that the nature inspired optimization algorithms (IPSO and FPA) are mixed with MARSplines and BPNN in forecasting. Like other nonlinear forecasting models, the newly proposed models are capable of handling chaotic, irregular and noisy data, so the proposed models can be used in financial market data, as well as in other time series forecasting studies (Ghoddusi et al., 2019; Gan et al., 2020; Tang et al., 2020).

The results of this research are expected to help traders, investors, firms and policy makers to make swift decisions. Besides crude oil, the proposed model can be used for forecasting other energy commodities, stocks, or economic indicators (Ghoddusi et al., 2019). By using dimension reduction methods, decision makers can learn to comprehend the significant variable which considerably influences the path of future crude oil prices. It also helps to save the running time of the model that helps decision makers to reach decisions in time (Roy et al., 2018; Chen et al., 2019), which may result in reducing the expected costs and losses associated with the forecasting process. This forecasting model can be also used by hedge fund managers and asset managers to build long-short strategies to make sufficient profits. Moreover, OPEC and OECD countries can also use the recommended models to frame policies in relevance to global crude oil prices (Tang et al., 2020). Finally, this study contributes to the development of more reliable forecasting models and anticipates that the findings will be valuable in supporting policymakers and aiding their decision making process in international energy markets (Degiannakis and Filis, 2018).

Our research is confined to comparing hybrid models built with MARSplines, IPSO, FPA and BPNN. It can be compared with the models developed by Azadeh et al. (2015), Chai et al. (2018b), Ding, (2018) and Chen et al. (2019) which may yield different results. In this study, only sixtieth minute is used, future researchers can take every fifth minute, tenth minute, fifteenth minute, and so on to compare the output of MARSplines-IPSO-BPNN and MARSplines-FPA-BPNN. They can attempt to build a new hybrid model (s) with new released data mining algorithms and nature inspired algorithms and compare the performances with MARSplines-IPSO-BPNN and MARSplines-FPA-BPNN. The researcher may embrace these models to check its forecasting efficiency in the recession era.

References

Ahmed, R.A., Shabir, A.B., 2014. Daily crude oil price forecasting model using arima, generalized autoregressive conditional heteroscedastic and support vector machines. Appl. Math. Sci. 11 (3), 48–53.
Alfi, A., Modares, H., 2011. System identification and control using adaptive particle swarm optimization. Appl. Math. Model. 35 (3), 1210–1221.
Atsalakis, G.S., Valavanis, K.P., 2009. Surveying stock market forecasting techniques-part II: soft computing methods. Exp. Syst. Appl. 36 (3), 5932–5941.
Azadeh, A., Asadzadeh, S.M., Mirzarzai, G.H., Saberi, M., 2015. An emotional learning-neuro-fuzzy inference approach for optimal training and forecasting of gas consumption estimation models with cognitive data. Technol. Forecast. Soc. Change 91, 47–55.
Berry, M.J., Linoff, G., 1997. Data Mining Techniques. John Wiley & Sons, New York.
Chai, J., Lu, Q., Hu, Y., Wang, S., Lai, K.K., Liu, H., 2018a. Analysis and Bayes statistical probability inference of crude oil price change point. Technol. Forecast. Soc. Change 126, 271–283.
Chai, J., Xing, L.M., Zhou, X.Y., Zhang, Z.G., Li, J.X., 2018b. Forecasting the WTI crude oil price by a hybrid-refined method. Energy Econ. 71, 114–127.
Chatziantoniou, I., Degiannakis, S., Filis, G., 2019. Futures-based forecasts: how useful are they for oil price volatility forecasting? Energy Econ. 81, 639–649.
Chen, H., Liao, H., Tang, B.J., Wei, Y.M., 2016. Impacts of OPEC’s political risk on the international crude oil prices: an empirical analysis based on the SVAR models. Energy Econ. 57, 42–49.
Chen, Y., Hao, Y., 2017. A feature weighted support vector machine and K-nearest neighbor algorithm for stock market indices prediction. Exp. Syst. Appl. 340, 340–355.
Chen, Y.J., Lin, J.A., Chen, Y.M., Wu, J.H., 2019. Financial forecasting with multivariate adaptive regression splines and queen genetic algorithm-support vector regression. IEEE Access 7, 112931–112938.
Chinnnathambi, R.A., Campbell, M., Nair, A.S., Ranganathan, P., 2018, October. Investigation of price-feature selection algorithms for the day-ahead electricity market. In: Proceedings of the 2018 IEEE Electrical Power and Energy Conference (EPEC). IEEE, pp. 1–6.
Chiroma, H., Khan, A., Abubakar, A.I., Saadi, Y., Hamza, M.F., Shuib, L., Herawan, T., 2016. A new approach for forecasting OPEC petroleum consumption based on neural swarm train by using flower pollination algorithm. Appl. Soft Comput. 48, 50–58.
Dai, W., Shao, Y.E., Lu, C.J., 2018. Incorporating feature selection method into support vector regression for stock index forecasting. Neural Comput. Appl. 23 (6), 1551–1561.
de Albuquerquemello, V.P., de Medeiros, R.K., da Nóbrega Besarria, C., Maia, S.F., 2018. Forecasting crude oil price: does exist an optimal econometric model? Energy 155, 578–591.
Debnath, K.B., Mourshed, M., 2017. Forecasting methods in energy planning models. Renew. Sustain. Energy Rev. 88, 297–325.
Degiannakis, S., Filis, G., 2018. Forecasting oil prices: high-frequency financial data are indeed useful. Energy Econ. 76, 388–402.
Diebold, F.X., Mariano, R.S., 1995. Comparing predictive accuracy. J. Bus. Econ. Stat. 13 (3), 253–263.
Ding, Y., 2018. A novel decompose-ensemble methodology with AIC-ANN approach for crude oil forecasting. Energy 154, 328–336.
Ding, Z., Liu, Z., Zhang, Y., Long, R., 2017. The cognition effect of international crude oil price fluctuations on Chinese stock market investor sentiment. Appl. Energy 187, 27–36.
Drachal, K., 2016. Forecasting spot oil price in a dynamic model averaging framework—are the determinants changed over time? Energy Econ. 60, 35–46.
EIA, 2019. Annual Energy Outlook 2019 with Projections to 2050. Technical Report. US Department of Energy, Washington, DC, pp. 32. Available at. https://www.eia.gov/outlooks/aeo/pdf/aeo2019.pdf Accessed on 12 November, 2019.
Engelbrecht, A.P., 2005. Fundamentals of Computational Swarm Intelligence. John Wiley & Sons, Ltd, Hoboken.
Fan, L., Pan, S., Li, Z., Li, H., 2016. An ICA-based support vector regression scheme for forecasting crude oil prices. Technol. Forecast. Soc. Change 112, 245–253.
Friedman, J.H., 1991. Multivariate adaptive regression splines. Ann. Stat. 19 (1), 1–67.
Gan, L., Wang, H., Yang, Z., 2020. Machine learning solutions to challenges in finance: an application to the pricing of financial products. Technol. Forecast. Soc. Change 153, 119928. https://doi.org/10.1016/j.techfore.2020.119928. Accessed on 01 April, 2020.

Ghodsi, H., Creamer, G.G., Rafaizadeh, N., 2019. Machine learning in energy economics and finance: a review. Energy Econ. 81, 709–727.

Guo, X., Li, D., Zhang, A., 2012. Improved support vector machine oil price forecast model based on genetic algorithm optimization parameters. Aaeer Proc. 1, 525–530.

Gupta, R., Wohar, M., 2017. Forecasting oil and stock returns with a VAR model using over 150 years off data. Energy Econ. 62, 181–186.

Han, J., Pei, J., Kamber, M., 2011. Data Mining: Concepts and Techniques, 3rd ed. Elsevier, San Francisco.

Hamid, M.H., Shahri, A., 2017. May, May. Wavelet regression model in forecasting crude oil price. In: Proceedings of the AIP Conference Proceedings. 1842 AIP Publishing, 030019.

He, K., Tso, G.K., Zou, Y., Liu, J., 2018. Crude oil risk forecasting: New evidence from multiscalar analysis approach. Energy Econ. 76, 574–583.

Herrera, G.P., Constantinop, M., Tabak, B.M., Pistori, H., Su, J.J., Narzananawa, A., 2019. Long-term forecast of energy commodities price using machine learning. Energy, 179, 214–221.

Hou, A., Suardi, S., 2012. A nonparametric GARCH model of crude oil price return volatil- ity. Energy Econ. 34 (2), 618–626.

Hu, J., Du, H., Lei, Y., 2018. August. Forecasting crude oil price based on EMD-wavelet- GARCH model. In: Proceedings of the International Conference on Management Science and Engineering Management. Springer, Cham, pp. 543–554.

Isham, M.F., Leong, M.S., Lim, M.H., Zakaria, M.K., 2019. A review on variational mode decomposition for rotating machinery diagnosis. In: Proceedings of the MATEC Web of Conferences. 255. EDP Sciences, pp. 02017.

Jammazi, R., Aloui, C., 2012. Crude oil price forecasting: experimental evidence from wavelet decomposition and neural network modeling. Energy Econ. 34 (3), 828–841.

Jianwei, L., Bao, Y., Ye, J., 2017. Crude oil price analysis and forecasting based on variational mode decomposition and independent component analysis. Phys. A: Stat. Mech. Appl. 484, 412–427.

Kao, L.J., Chiu, C.C., Lu, C.J., Chang, C.H., 2013. A hybrid approach by integrating wavelet-based feature extraction with MARS and SVR for stock index forecasting. Decis. Supp. Syst. 54 (3), 1228–1244.

Lahmiri, S., 2015. Comparing variational and empirical mode decomposition in fore- casting day-ahead energy prices. IEEE Syst. J. 11 (3), 1907–1910.

Li, J., Zhu, S., Wu, Q., 2019. Monthly crude oil spot price forecasting using variational mode decomposition. Energy Econ. 83, 240–253.

Ma, F., Zhang, Y., Huang, D., Lai, X., 2018. Forecasting oil futures price volatility: new evidence from realized range-based volatility. Energy Econ. 75, 400–409.

Mobin, S., Feronaut, F., 2006. Forecasting nonlinear crude oil futures prices. Energy J. 27 (4), 81–95.

Mohapatra, S.K., Mohapatra, S., 2008. Forecasting the differences between various commercial oil prices in the Persian Gulf region by neural network. Energy 36 (7), 3979–3984.

Nademi, A., Nademi, V., 2018. Forecasting crude oil prices by a semiparametric Markov switching model: OPEC, WTI, and Brent cases. Energy Econ. 74, 757–766.

Naser, H., 2016. Estimating and forecasting the real prices of crude oil: a data rich model using a dynamic multinomial choice (DMC) approach. Energy Econ. 56, 75–87.

Niu, T., Wang, J., Lu, H., Yang, W., Du, P., 2020. Developing a deep learning framework with two-stage feature selection for multivariate financial time series forecasting. Expert Syst. Appl. 148, 113237. https://doi.org/10.1016/j.eswa.2020.113237. Available atAccessed on 01 April, 2020.

Pan, L., Feng, X., Sang, F., Li, L., Leng, M., Chen, X., 2019. An improved back propagation neural network based on complexity decomposition technology and modified flower pollination optimization for short-term load forecasting. Neural Comput. Appl. 31 (7), 2679–2697.

Papadimitriou, T., Gogas, P., Statilakis, E., 2014. Forecasting energy markets using sup- port vector machines. Energy Econ. 83, 574–583.

Polanco-Martínez, J., Abadie, L., 2016. Analyzing crude oil spot price dynamics versus companies listed on Tehran stock exchange using multivariate adaptive regression splines, extreme learning machine, a hybrid model of MARS and ELM. Renew. Sustain. Energy Rev. 82 (3), 4256–4268.

Papadimitriou, T., Gogas, P., Stathakis, E., 2014. Forecasting energy markets volatility using GARCH models: can multivariate models beat univariate models? Energy Econ. 34 (6), 2167–2181.

Wang, T., Yang, J., 2010. Nonlinearity and intraday efficiency tests on energy futures markets. Energy Econ. 83, 240–253.

Zhang, W., Qu, Z., Zhang, K., Mao, W., Ma, Y., Fan, X., 2017. A deep learning ensemble approach for crude oil price forecasting. Eng. Appl. Artif. Intell. 64, 136–149.

Zhang, X.S., 2012. September. Flower pollination algorithm for global optimization. In: Proceedings of the International Conference on Unconventional Computing and Natural Computation. Springer, Berlin, Heidelberg, pp. 240–249.

Zhang, Y., Dui, W., Tang, L., 2016. A novel decomposition ensemble model with extended extreme learning machine for crude oil price forecasting. Eng. Appl. Artif. Intell. 47, 110–121.

Zheng, Z., Ma, F., Wei, Y., 2019. Out-of-sample prediction of the oil futures market volatil- ity: a comparison of new and traditional combination approaches. Energy Econ. 11, 819–1120.

Zhang, G.P., Patuwo, B.E., Hu, M.Y., 2001. A simulation study of artificial neural net- works for nonlinear time-series forecasting. Comput. Oper. Res. 28 (4), 381–396.

Zhang, J.L., Zhang, Y.J., Zhang, L., 2015. A novel hybrid method for crude oil price forecasting. Energy Econ. 49, 649–659.

Zhang, W., Qu, Z., Zhang, K., Mao, W., Ma, Y., Fan, X., 2017. A combined model based on COMEDIAN and modified flower pollination algorithm for wind speed forecasting. Energy Convers. Manag. 136, 439–451.

Zhao, C.L., Wang, B., 2014. Forecasting crude oil price with an autoregressive integrated moving average (ARIMA) model. In: Proceedings of the Fuzzy Information and Engineering and Operations Research and Management. Springer, Berlin, Heidelberg, pp. 275–286.

Zhao, L., Cheng, L., Wan, Y., Zhang, H., Zhang, Z., 2015. A VAR-SVM model for crude oil price forecasting. Int. J. Global Energy Issues 38 (1–3), 126–144.

Zhao, Y., Li, J., Yu, L., 2017. 2016. A hybrid learning ensemble approach for crude oil price forecasting. Energy Econ. 66, 9–16.

Zou, Y., Lu, T., Tso, G.K., He, K., 2020. Risk forecasting in the crude oil market: a mul- ticluster convolutional neural network approach. Phys. A: Stat. Mech. Appl. 541, 123360. https://doi.org/10.1016/j.physa.2019.123360. Available atAccessed on 01 April, 2020.