Author: Christos Vasilopanagos

Date: 4 April 2025

Background & theory

Since the 1970s, the global oil price has exhibited escalating volatility, partly due to the introduction of futures trading, which intensified market speculation (Gharib et al., 2021). Recent years have witnessed fluctuations spurred by increasing demand in developing nations and augmented supply, particularly from new production in the United States. Instances of economic recession, such as the 2008 downturn, witness considerable augmentation in oil reserves due to unanticipated declines in demand. The unforeseen impact of the ongoing COVID-19 pandemic sent shockwaves through the oil market, precipitating a sharp decline in prices (Bandyopadhyay and Kaushik 2021). This upheaval reached an unprecedented level when the benchmark for US crude oil plunged into negative territory in late April 2020, alongside a significant drop in the price of Brent Crude, the global benchmark (Bouazizi et al., 2024).

Inventories serve as a mechanism to equilibrate such a shock in supply and demand of oil. Surplus oil production surpassing immediate consumption necessitates storage for future utilization. Facilities like refineries and storage depots stockpile crude oil and derivative products such as gasoline and heating oil, ensuring readiness for periods of heightened demand. The volume of stored oil can exert notable influence on forthcoming prices, as perceptions of future demand fluctuations prompt adjustments in inventory levels (Kilan et al., 2014; Ye et al., 2005). Moreover, nations including the US uphold emergency oil reserves to mitigate potential disruptions in supply chains. The onset of the COVID-19 pandemic has notably disrupted crude oil markets, amplifying susceptibilities in oil-exporting developing economies. The concomitant decline in oil prices elevated the prospect of macroeconomic and fiscal crises, along with social unrest, given the pivotal role of oil in these countries' trade and fiscal dynamics. Crude oil storage has become more diverse, with storage hubs emerging in non-OECD countries (Nima, 2021). These market changes affect storage operators' strategies, shaping oil market dynamics in different regions. Challenges persist in quantifying the relationship between crude inventories and market structures, given the lack of data on convenience yield, marginal storage costs, and inventory shadow prices. While recent advancements offer improved data accuracy, differences between theory and empirical observations persist. Alternative theories, such as those proposing non-linear relationships between inventory levels and storage values, challenge conventional storage models' validity, highlighting the ongoing debate within the industry (Daniel and Wiegratz, 2023). Despite the advancements in machine learning that have culminated in better predictive capabilities, opacity pertains as a fundamental bottleneck surrounding predictive modelling as well as choosing between the appropriate degree of model complexity in crude oil trading (Petropoulos et al., 2022; Rudy and Sun, 2018).

With respect to the above, our contribution through this article can be summarized as follows:

- *A simplistic workflow of models designed for increased accuracy when predicting regional crude oil inventory combining linearity via multiple regression and non-linearity via the MLP and NARX modelling respectively.*
- *A comparative study of the models aiding in the development of appropriate theoretical trading strategies with the necessary market indices.*

1.1 Regression analysis (MLR)

Regression analysis has been historically utilized extensively as a fundamental statistical method both in econometrics and for predictive modeling in the crude oil market. The linear relationship between one dependent and one or more independent variables is examined where, in the case of crude oil trading, this can be the price of feedstock, i.e. crude oil, and the respective inventory, commercial, strategic or operational. Previous studies have showcased the significance of regression modelling as tool for assessing crude oil market dynamics using methods such as weighted least squares and Multi-Step Mixed Data Sampling LASSO regression proving regression models still perform well despite the availability of the latest modelling approaches to assess the crude oil market, such as Google Trends, co-integration tests, Granger causality analysis and machine learning (ML) (Qin et al.,2023; Wang et al., 2022; Li et al., 2022; Algahtani et al., 2020; Hardy et al., 2004). The standard regression equation is given below:

$$y = a + bx + \epsilon \quad (1)$$

where, y represents the dependent variable being predicted or explained. The constant a signifies the value of y when x equals 0, b denotes the coefficient of x, indicating the slope of the regression line and how much y changes for each change in x and x stands for the independent variable that predicts or explains the value of y. Finally, ϵ denotes the error term, representing the discrepancy in predicting the value of y given the value of x (Sezer et al., 2017).

1.2 Neural Networks and Multilayer Perceptrons (MLP)

Computational intelligence has a history in trading systems with neural networks being a popular choice. Neural network algorithms have been developed to construct profitable trading systems, and optimize technical analysis for stock trading (Chen et al., 2003). In addition to forecasting technical indicators neural networks, for instance, probabilistic (PNNs), have been used to enhance investment strategies trained with historical data having shown such models can yield higher returns compared with other strategies (Xu et al., 2019). In terms of predicting ability, neural network models such as Deep Belief Networks (DBN) have been shown to capture non-linearity, improving forecasting accuracy when compared with statistical models, such as Autoregressive Moving Average (ARMA) (Taud, Mas, 2018).

A neural network is a set of combined neurons forming sets of layers. The first layer of an MLP can be mathematically expressed as follows (Eugen, 2012):

$$b_j = \sum_{i=0}^{D} W_{ji}^{(1)} x_i, \;\; j = 1,2, \dots, M, \qquad (2)$$

where activations are defined as b_j and weights as $W_{ji}^{(1)}$, where (1) denotes the first layer of the MLP. A sigmoid non-linear activation function transforms the activations shown mathematically below:

$$z_j = h(b_j) = \frac{1}{1 + e^{-b_j}}, \qquad (3)$$

Where z_j corresponds to the output of a basis function that can be interpreted as that of a hidden unit. A subsequent second layer, formed of hidden units combined in a linear fashion, that carry out activation of the K output units. This can be shown as follows:

$$a_k = \sum_{i=0}^{M} W_{kj}^{(2)} z_i, \;\; k = 1,2 \dots, K. \; (4)$$

With $z_i = 1$, corresponding to a bias and $W_{kj}^{(2)}$ and (2) corresponding to weights and the second subsequent layer respectively. An activation function is applied to the output units. The model performs a loss minimization through iterative back-propagation that allows the optimization of the model. The error function is defined as the sum of squares of the error, i.e., predicted minus true values, mathematically below:

$$E = \sum_{n=1}^{N} E_n, \quad (5)$$

In the case of performing this programmatically via Python, activation is done via the Rectified Linear Unit (ReLU), applied to an output and allowing the introduction of non-linearity to the model. This is to avoid vanishing gradients as it is a constant gradient for positive inputs. The ReLU function returns the value 0 when given a negative input and a value when given a positive one. This can be expressed as follows:

$$F(x) = \max(0, x). \text{ (6)}$$

Specifically Multilayer Perceptron models (MLPs), serve as an extension of a feedforward neural network model, with a structure consisting of an input, one or an arbitrary number of middle/hidden layers and an output layer. The input layer serves as an input receiving layer and receives the initial signal for processing, while the output layer carries out the tasks of predicting and classifying. The model's true computational core lies in the middle, hidden layers, positioned between the input and output. Data movement is performed in a forward direction, progressing from the input to the output layer. MLP's neurons are trained using backpropagation learning algorithms to provide approximate continuous functions and as such, become useful for items that lack linear separability. The challenge with MLP modelling is how to update the weights of the inputs to artificial neurons for the hidden units as well as how to adapt weights in its hidden layer(s), or more precisely, how to calculate the delta rule. The delta rule utilizes the variance between the desired target activation (i.e., target output values) and the actual obtained activation to facilitate the learning process. The backpropagation algorithm, a generalization of the delta rule, is based on gradient descent to minimize the sum squared difference between the target and the actual network outputs. This algorithm follows a two-stage process: a feedforward phase, where an input vector is applied and the signal propagates through the network layers, and a feedback phase, where the error signal is fed back through the network layers to modify the weights, effectively minimizing the error across the entire training set and navigating the error surface in weight-space.

1.3 Nonlinear AutoRegressive Modelling with eXogenous inputs (NARX)

The Nonlinear AutoRegressive model with eXogenous inputs (NARX) represents a recurrent dynamic network characterized by feedback connections that encompass multiple layers of the network (Mamta, 2023). Derived from the linear AutoRegressive with eXogenous inputs (ARX) model, widely utilized in time-series modeling, the NARX model introduces nonlinearities and recurrent connections, thereby extending the modeling capabilities beyond traditional feedforward architectures. The (NARX) has gathered significant attention in recent years, emerging as a robust solution to address the complexities posed by nonlinearity in various systems including finance, engineering and science, with time-varying characteristics. Leveraging the strength of artificial neural networks (ANN) and optimization algorithms, NARX modelling provides the option of handling dynamic inputs expressed by time-series datasets, offering enhanced applications in a number of fields such as quantitative investment forecasting. NARX advantage lies in its ability to predict future instances for any input signal, irrespective of the underlying process that initializes the time series. This property makes it particularly suitable for state estimation tasks, where accurate predictions are essential amidst continuous fluctuations. Regarding its performance in trading and the capital markets, NARX has been previously used in the context of stock price predictions as well as a support vector machine for algorithmic trading, predicting future values of financial time series (Cheon et al., 2020, also see Banerjee et al., 2019 and references therein). The fundamental NARX equation is given below:

$$y(t) = f(y(t-1), y(t-2), \dots, y(t-ny), u(t-1), u(t-2), \dots, u(t-nu)), \quad (7)$$

where the next value of the dependent output signal $y(t)$ is determined based on previous values of the output signal and previous values of an independent (exogenous) input signal, whereas $u(t)$ represents the lagged values of the exogenous input u up to nu time steps back. To implement the NARX model, a feedforward neural network can be employed to approximate the function f. The resulting network structure typically consists of two layers in a feedforward configuration, accommodating a vector AutoRegressive with eXogenous inputs (ARX) model, allowing for multidimensional input and output signals.