Integration of Multi-Frequency Ensemble Machine Learning Models Towards Crude Oil Price Forecasting

Farid Javadnejad, Ph.D.[*]

* Geospatial Engineering Consultant, Aerogeospatial LLC, 1401 21st Street, Suite R, Sacramento, CA 95811, e-mail: *nejad.fj@gmail.com*.

1. Introduction

Crude oil and other refined liquid products from fossil fuels are critical contributors to the world economy. Petroleum has been the largest energy source for all countries. Its products run vehicles, heat buildings, and produce electricity. Moreover, various industries use petroleum as a raw material to produce intermediate or end-user products that we use daily (EIA 2022b; Lu et al. 2021; Deng, Ma, and Zeng 2021; Kilian and Murphy 2014). In 2019, global petroleum consumption neared 100 million barrels per day (Table 1).

Table 1. The largest oil consumers and their share of total world consumption (EIA 2022b)

Ranking	Country	Million barrels per day	Share of world total
1	United States	20.54	20%
2	China	14.01	14%
3	India	4.92	5%
4	Japan	3.74	4%
5	Russia	3.70	4%
6	Saudi Arabia	3.18	3%
7	Brazil	3.14	3%
8	Canada	2.63	3%
9	South Korea	2.60	3%
10	Germany	2.35	2%
	World total	100.23	

1.1. Problem statement

Crude oil prices are difficult to predict accurately due to the number of influencing factors and the highly complex behavior of such influences. Global economic and social activities can be substantially impacted by fluctuations in crude oil prices. Threfore, despite challenges for prediction of oil price, accurate oil price forecasting is crucial for decision-making support for the manufacturing, logistics, and government sectors to guide industrial and social policies and practices (Kilian and Murphy 2014; Deng, Ma, and Zeng 2021; Lu et al. 2021).

1.2. Background

Brent, West Texas Intermediate (WTI), Dubai/Oman, and Shanghai crude oil prices are the major benchmarks of the crude oil market and are reported is USD per barrel unit. Factors such as supply and demand, financial markets and economics, politics, global events, renewable energy and alternative resources, new resources and development of new oil extraction technologies, social & environmental policies, and consumption patterns may influence the crude oil market dynamics. Such impacts and resultant price fluctuations might be very complex and may occur at different frequencies.

Classical econometric models such as random walk, autoregressive integrated moving average (ARIMA), error correction model (ECM), generalized autoregressive conditional heteroscedasticity (GARCH) model are used for crude oil price prediction. Recently, machine learning (ML) methods such as artificial neural network (ANN) and support vector machine (SVM) are used for the crude oil price prediction, which provide powerful tools to model nonlinear behavior or crude oil market dynamics (Jammazi and Aloui 2012; Lanza, Manera, and Giovannini 2005; Hou and Suardi 2012; Basiri 2015; Yu, Zhao, and Tang 2017; Murat and Tokat 2009; Kilian and Murphy 2014; Javadnejad 2012).

1.3. Objectives

In this work, an ML model is proposed to predict crude oil price using multiple infulencing factors. The predictions are casted on multiple time-series to consider for complex factors that imact the market dynamics in differenct frequencies.

This report is structured as follows. Section describes the metholdogy, where Sub-section 2.1 summarizes the datasets that are used in this study. Sub-section 2.2 describes the data preparation and data wrangling procedures. In Sub-section 2.3, the exploratory data analysis and feature engineering approcahes for ML training are described. Sub-section 2.4 covers the ML pre-processing, training, models selection, model metrics. In Sub-section 2.5, the final results for model training and validation are presented, as well as the predictions for 6-month time frequencies. We present the discussion of our results in Section 3. Finally, in Section 4, we summarise our recoomendaiton for future work.

2. Methodology

2.1. Datatets

The factors that influence the crude oil market dynamics include supply and demand, financial markets, politics, global events, alternative resources, development technologies, policies, and consumption patterns (Hamilton 2008; Hamilton 2009; Kilian and Murphy 2014; Zhao, Li, and Yu 2017; Lu et al. 2021; Wang, Wu, and Yang 2015). We use the crude oil prices of West Texas Intermediate (WTI) benchmark as the target feature. To take into account the aforementioned influencing factors a total of 32 feature variables were selected from publicly accessible data sources (EIA 2022a; FRED 2022; Investing 2022; WSJ 2022). Table 2 provides a list of the selected features, a description about each feature, and the sources of data.

Table 2. Selected dataset of feature variables for crude oil price

Category	Symbol	Variable	Unit	Source
Crude Oil Price	WTIPUUS	West Texas Intermediate Crude Oil Price	dollars per barrel	EIA
Supply	COPR_OPEC	Crude Oil Production, Total OPEC	million barrels per day	EIA
	PAPR_NONOPEC	Crude Oil Production, Total non-OPEC	million barrels per day	EIA
	INTL.55-1-WORL-TBPD	Crude Oil Production, NGPL, and other liquids production, World	thousand barrels per day	EIA
	COPRPUS	Crude Oil Production, U.S.	million barrels per day	EIA
Replacement Cost	RNGWHHD	Henry Hub Natural Gas Spot Price	dollars per million btu	EIA
Demand	PATC_OECD	Liquid Fuels Consumption, Total OECD	million barrels per day	EIA
	PATC_NON_OECD	Liquid Fuels Consumption, Total non-OECD,	million barrels per day	EIA
	FEDFUNDS	Federal Funds Effective Rate	percent, not seasonally adjusted	FRED
	IGREA	Index of Global Real Economic Activity	index, not seasonally adjusted	FRED
	CICPIUS	US Consumer Price Index (CPI): All Commodities	index, 1982-1984=1.00	EIA
	USACPIENGMINMEI	US Consumer Price Index (CPI): Energy for the United States	index 2015=100, not seasonally adjusted	FRED
	WPCPIUS	US Producer Price Index (PPI): All Commodities	index, 1982=1.00	EIA
	WP57IUS	US Producer Price Index (PPI): Petroleum	index, 1982=1.00	EIA
	EA19PIEAMI01GPM	roducer Price Index (PPI) of Euro Area (19 Countries)	index 2015=100, not seasonally adjusted	FRED
	ZOMNIUS	US Manufacturing Production Index (PMI)	index, 2017=100 (seasonally adjusted)	EIA
Inventory	PASC_OECD_T3	Petroleum Inventory, Total OECD	million barrels, end-of-period	EIA
	PASXPUS	Petroleum Inventory, U.S. Total	million barrels, end-of-period	EIA
	COSQPUS	US Crude Oil Inventory: Strategic Petroleum Reserve (SPR)	million barrels, end-of-period	EIA
	COSXPUS	US Crude Oil Inventory: Non-SPR	million barrels, end-of-period	EIA
Monetary Market	RTWEXBG	Real Broad Dollar Index	index Jan 2006=100, not seasonally adjusted	FRED
	DXY	US Dollar Index (DXY)	index	Investing
	DEXUSEU	U.S. Dollars to Euro Spot Exchange Rate (DEXUSEU)	U.S. dollars to one euro, not seasonally adjusted	FRED
Stock Market	SPX	S&P 500 Index	index	WSJ
	DJI	Dow Jones Industrial Index	index	WSJ
	COMP	NASDAQ index	index	WSJ
Commodity Market	Gold_Future	Gold Futures Historical Data	dollar per ounce	Investing

	Copper_Future	Copper Futures Historical Data	dollar per pound	Investing
Policy Uncertainty	GEPUCURRENT	Global Economic Policy Uncertainty Index: Current Price Adjusted GDP	index, not seasonally adjusted	FRED
Technology	MGWHUUS	Refiner Wholesale Gasoline Price	cents per gallon	EIA
	DSWHUUS	Diesel Fuel Refiner Wholesale Price	cents per gallon	EIA
	BREPUUS	Brent Crude Oil Spot Price	dollars per barrel	EIA

65

2.2. Data Cleaning and Data Wrangling

67   We used Jupyter Notebook 6.5.2 (Kluyver et al. 2016) and Python 3.9.15 (Python Software

68   Foundation 2022) to process the data. The features in Table 2 were read through APIs (if available)

69   or were downloaded directly from the data source. The data were initially set to be imported with

70   monthly intervals or averaged to monthly values, then were limited to the target time frame of

71   January 2000 and December 2022. The features were index based on their date values and then all

72   merged together on the date values to create corresponding feature values for each month. Figure

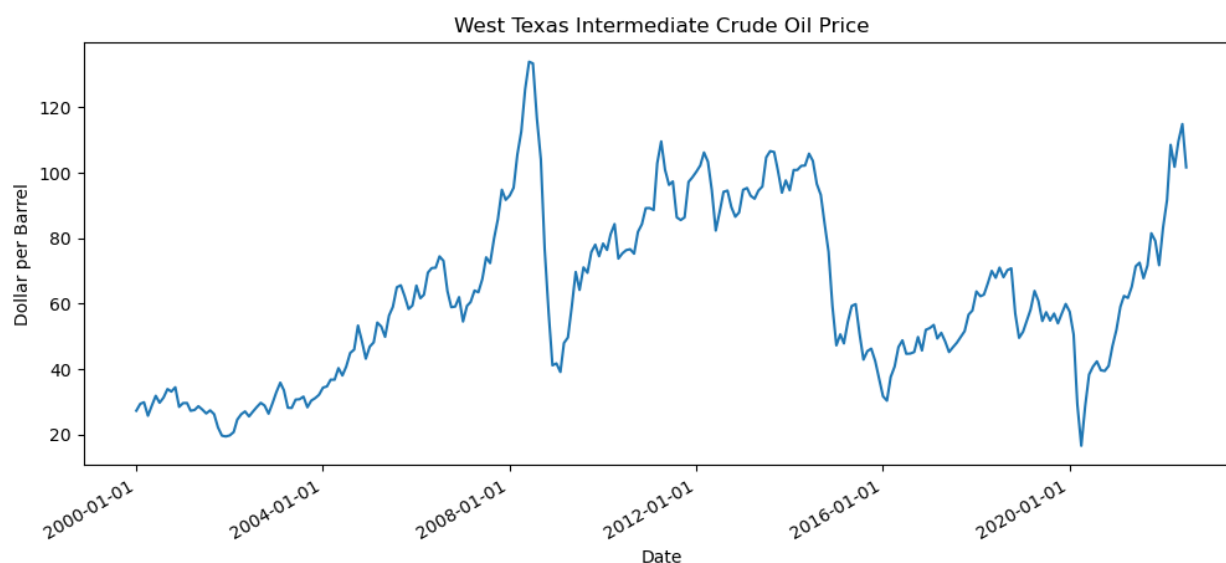73   1 shows the monthly West Texas Intermediate (WTI) crude oil price in the target time frame.

74

75   **Figure 1. West Texas Intermediate (WTI) crude oil price**

76

77   We used Pandas 1.5.2 (McKinney 2011) that is an open-source, simple, powerful, and

78 flexible library for data analysis and data manipulation, Pandas is for Python programming

79 language. The final dataset sized (274, 32) that represent 274-month records (rows) for 32 feature

80 variables (columns).

81   An important step of data wrangling is dealing with missing data. Table 3 shows the

82 summary statistics of missing data in feature variables. Missingno (Bilogur 2018) is also useful

83 tool that provides a series of visualisations for presence and distribution of the missing data within

84 a pandas dataframe. Figure 2 visually shows the distribution of the missing. To handle the missing

85 data, features with more than the 10% of missing data were dropped from the dataset. The columns

86 that had less than 1% missing features were imputed by using back and forward fill methods. For

87 the remaining missing data between 1% and 10%, the rows for all features were dropped to create

88 a dataset with no missing data. After treating missing data, the final dataframe sized (271, 30).

89

90   **Table 3. Summary statistics of missing data in feature variables**

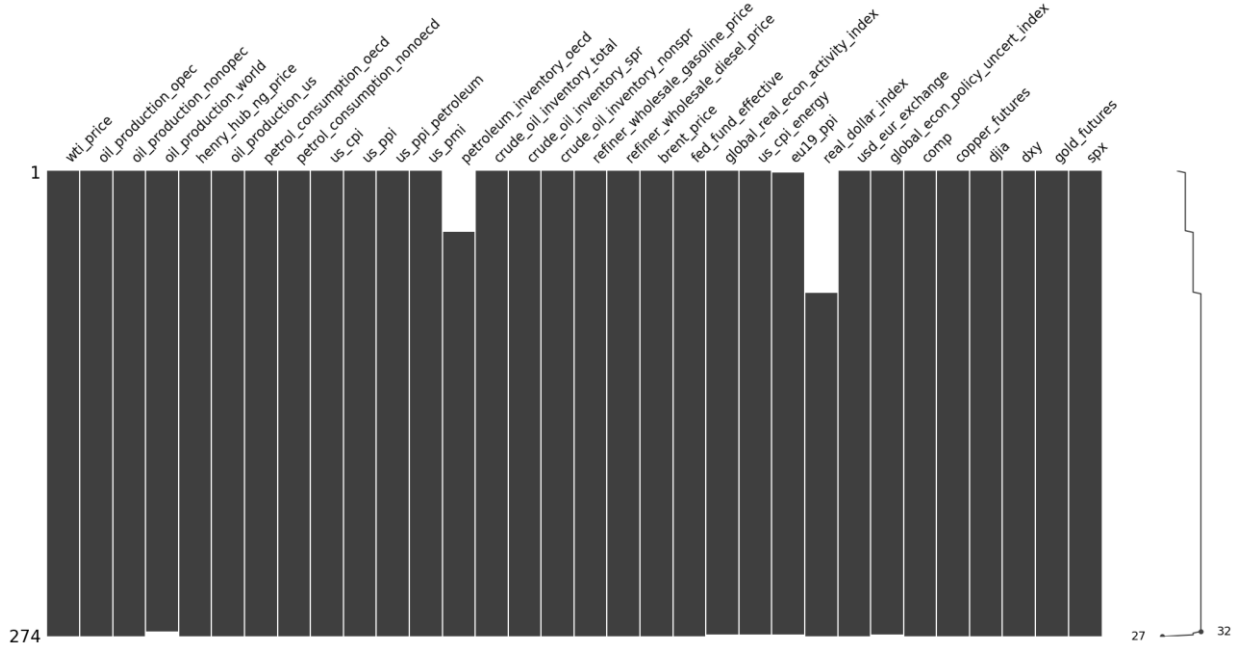| Variable	Count	Percentage
oil_production_world	3	1.1%
petroleum_inventory_oecd	36	13.1%
global_real_econ_activity_index	1	0.4%
us_cpi_energy	1	0.4%
eu19_ppi	2	0.7%
real_dollar_index	72	26.3%
global_econ_policy_uncert_index	1	0.4%

91

Figure 2. Missing data plot

2.1. Exploratory Data Analysis and Feature Engineering

WTI-Brent Spread is defined as difference between Brent and the WTI Crude Oil spot prices (Eq. 1). WTI is produced from the Gulf of Mexico basin and is the benchmark for oil price in American market. On the other hand, Brent Oil is produced from the North Sea and Atlantic basin and is the pricing benchmark for European market. While, the prices for these two benchmarks are highly correlated; difference between the prices often reflect technical, supply/demand or geopolitical issues.

$$\text{WTI-Brent spread} = \text{WTI spot price} - \text{Brent spot price} \tag{1}$$

The Crack spread is defined as the price difference between crude oil and its refined oil (Eq. 2 and 3), reflecting the supply and demand relationship between the crude oil market and its refined product market (Wang, Wu, and Yang 2015).

$$\text{WTI crack spread} = 3 \times \text{WTI spot price} - 2 \times \text{Gasoline Price} - 1 \times \text{Diesel Fuel Price} \tag{2}$$

$$\text{Brent crack spread} = 3 \times \text{Brent spot price} - 2 \times \text{Gasoline Price} - 1 \times \text{Diesel Fuel Price} \tag{3}$$

7

104   After calculating the spreads, Brent price, refiner wholesale gasoline price, and refiner

105   wholesale diesel price were dropped from the dataset.

106   Figure 3 shows the histograms of initial feature variables and feature engineered variables.

107   Figure 4 includes scatterplots of feature variable against the target variable of WTI oil price. Figure

108   5 and Figure 6 show the Pearson correlation coefficients matrix, and predictive power score (PPS)

109   matrix for feature variables, respectively.

110
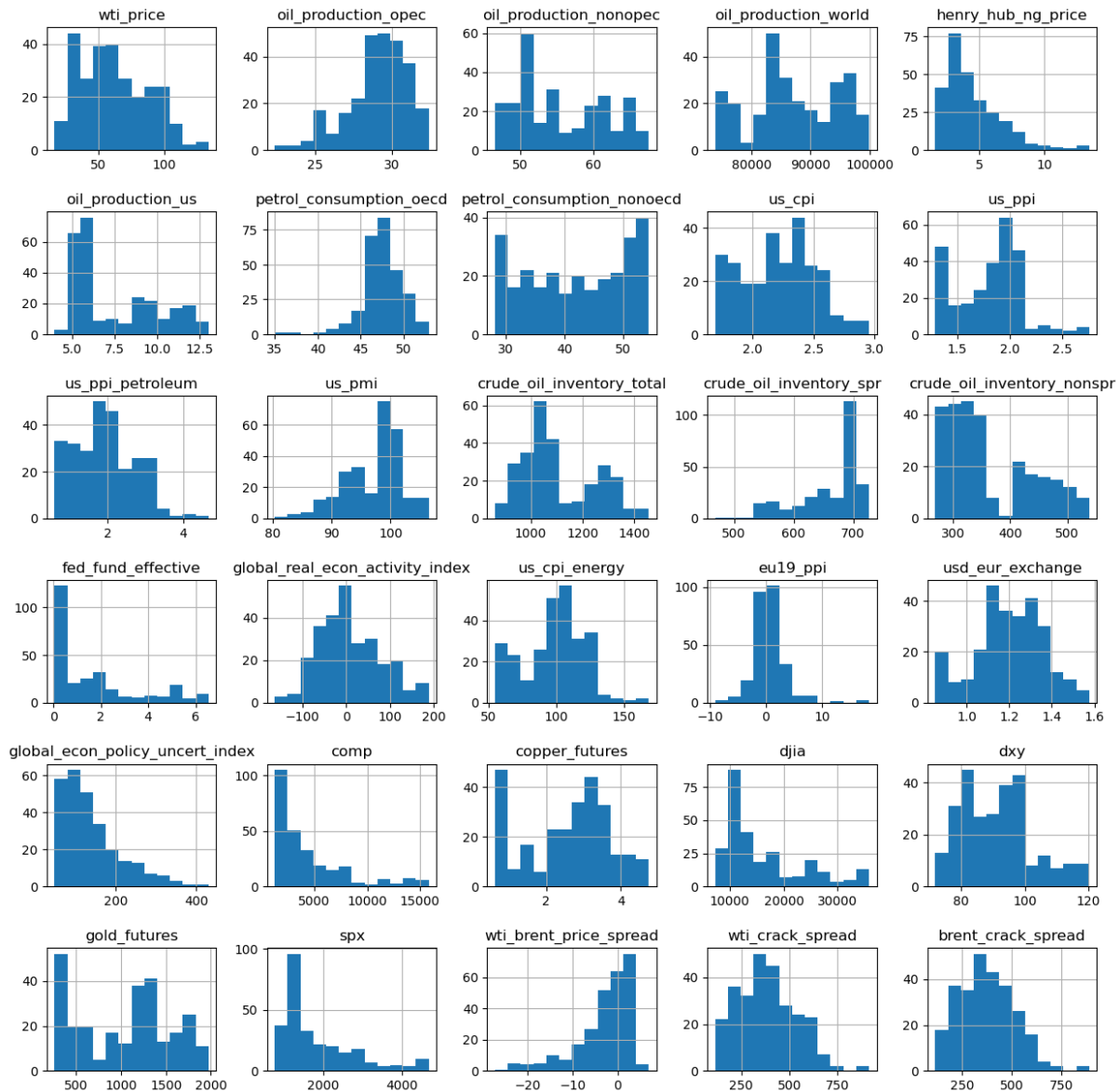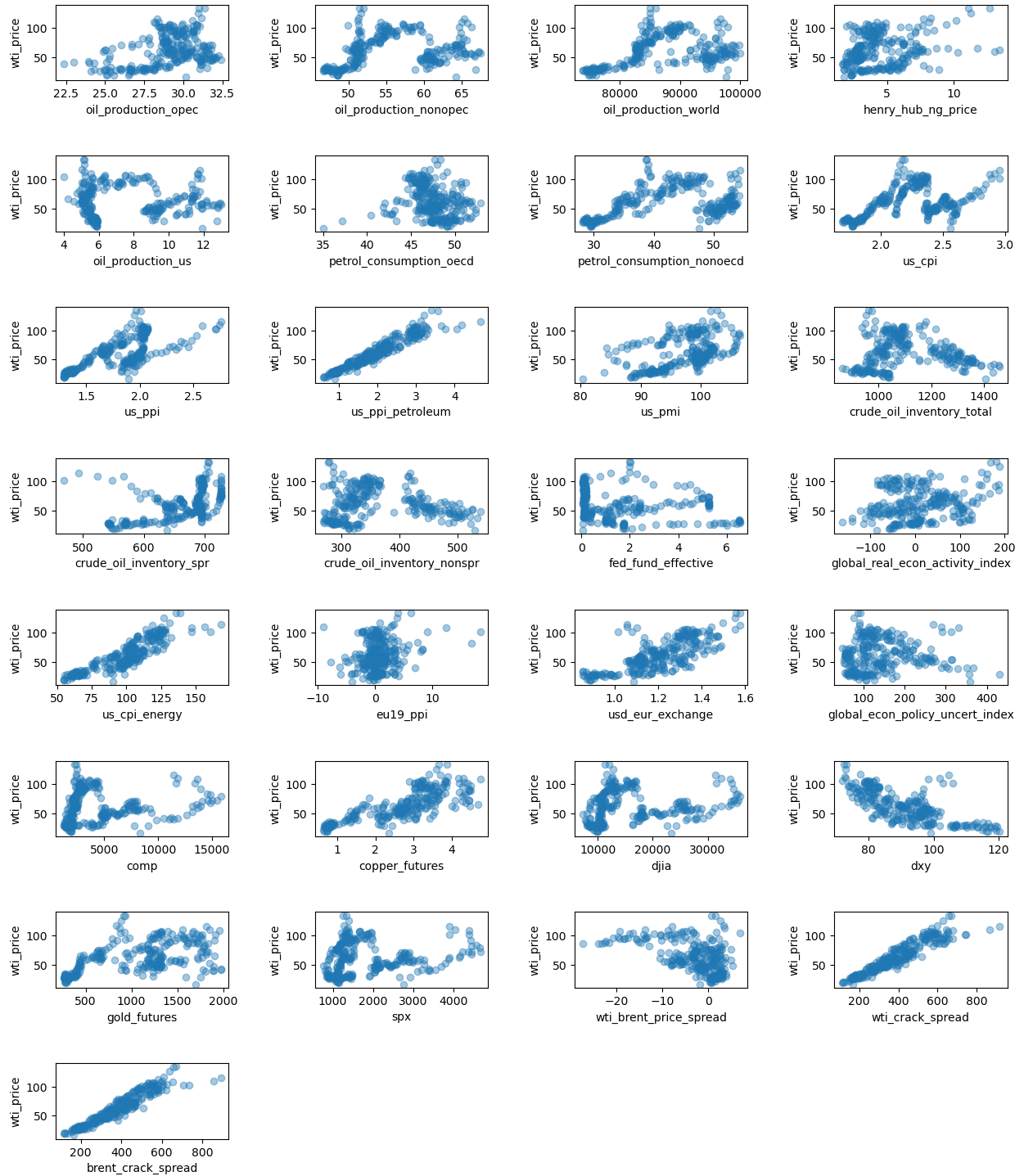

111   **Figure 3. Histograms of feature variables**

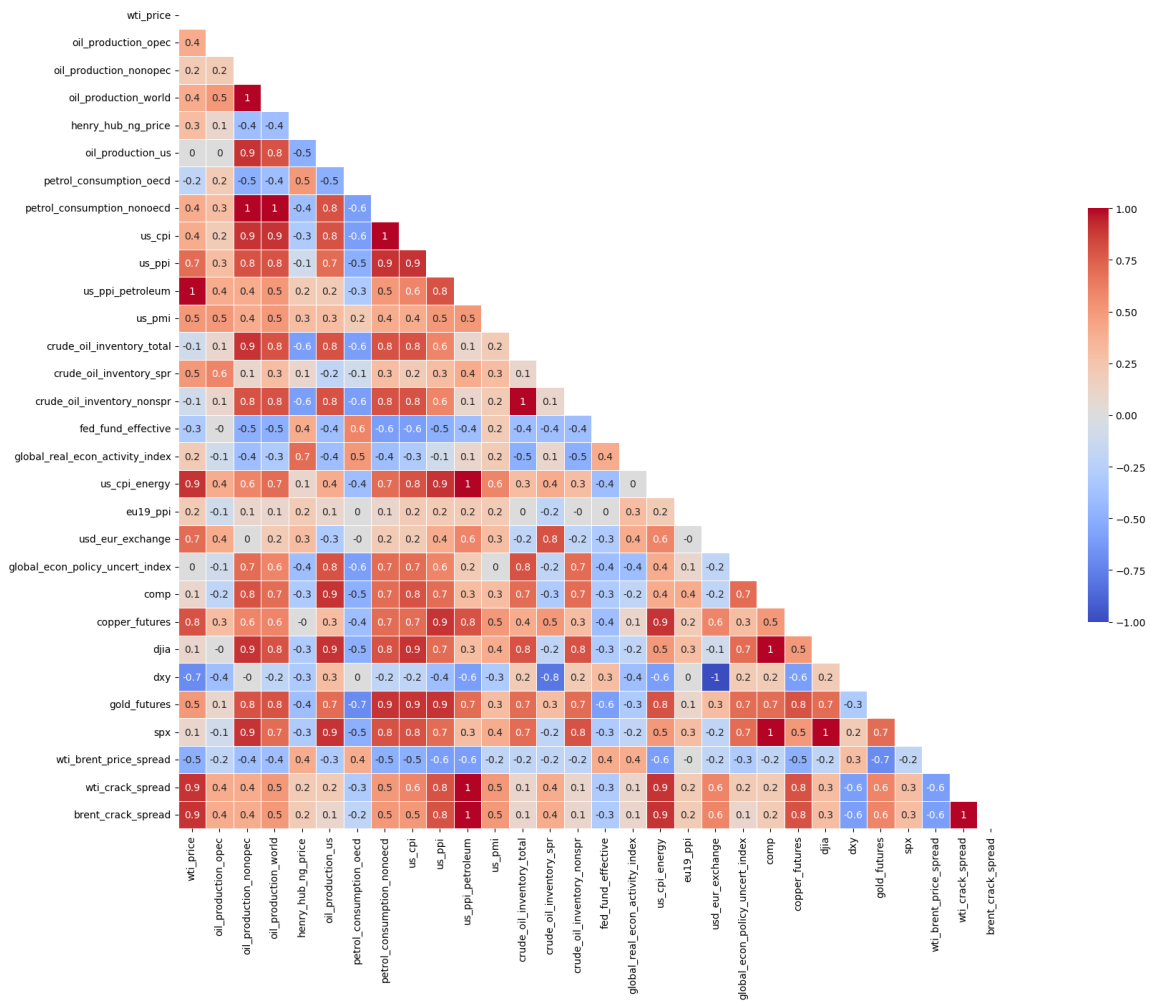**Figure 4. Scatter plots of feature variable against the target variable**

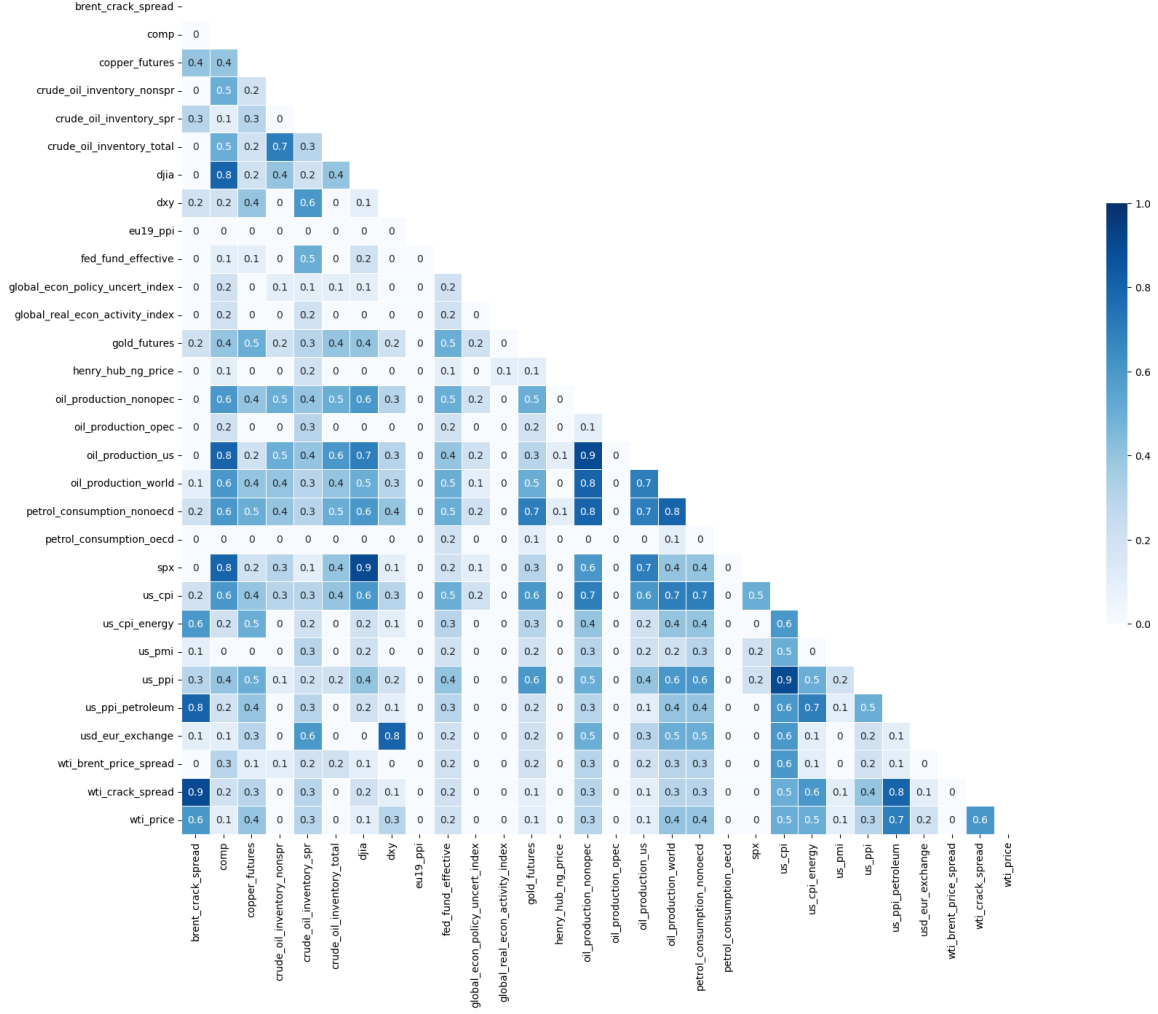**Figure 5. Pearson correlation coefficients matrix for feature variables**

Figure 6. Predictive Power Score (PPS) matrix for feature variables

2.2. Pre-processing and Training

- *Pre-processing*

The objective of this time-series analysis is to predict the WTI Price (dependent variable) using the independent variables for a time period in the future, where the independent variables are also unknow. To handle this, a data preparation step is needed before modelling so that the dependent value is shifted in time to be associated with the independent variables of previous time period. The time shift operation is formulated in Eq. 4:

$$Y_t = f(X_{t-n}, \beta_t) + e_{t-n} \tag{4}$$

11

127   where Y_t is the dependent variable at time t, X is the independent variable, n is the shift operator,

128   β is the regression coefficient, and e is the residual term.

129   We performed time shifting for frequency of $n = \{1, 2, 3, 4, 5, 6\}$ months. When we shift

130   the dependable variable in time, we end up having n rows with missing values. So, it is necessary

131   to handle the missing values by dropping the rows (as described is Sub-section 2.2).

132   After performing time shifting, the dependent variable will have a new set of independent

133   variables that are specific to that frequency, and in each frequency a different variable may have

134   stronger correlation in the WTI Price. In real world scenario, this can be described and the time-

135   lag for an effect to make an impact on the WTI prices. For example, changes in some features may

136   impact the WTI prices in short term, while some other features may impact the dependent variable

137   in a longer term. With the proposed methodology is expected to capture features with impact

138   frequencies of 1 to 6 months. Figure 7 shows the bar chart of sorted PPS coefficients of

139   independent variables against WTI oil price for 1-to-6-month time shift scenarios.
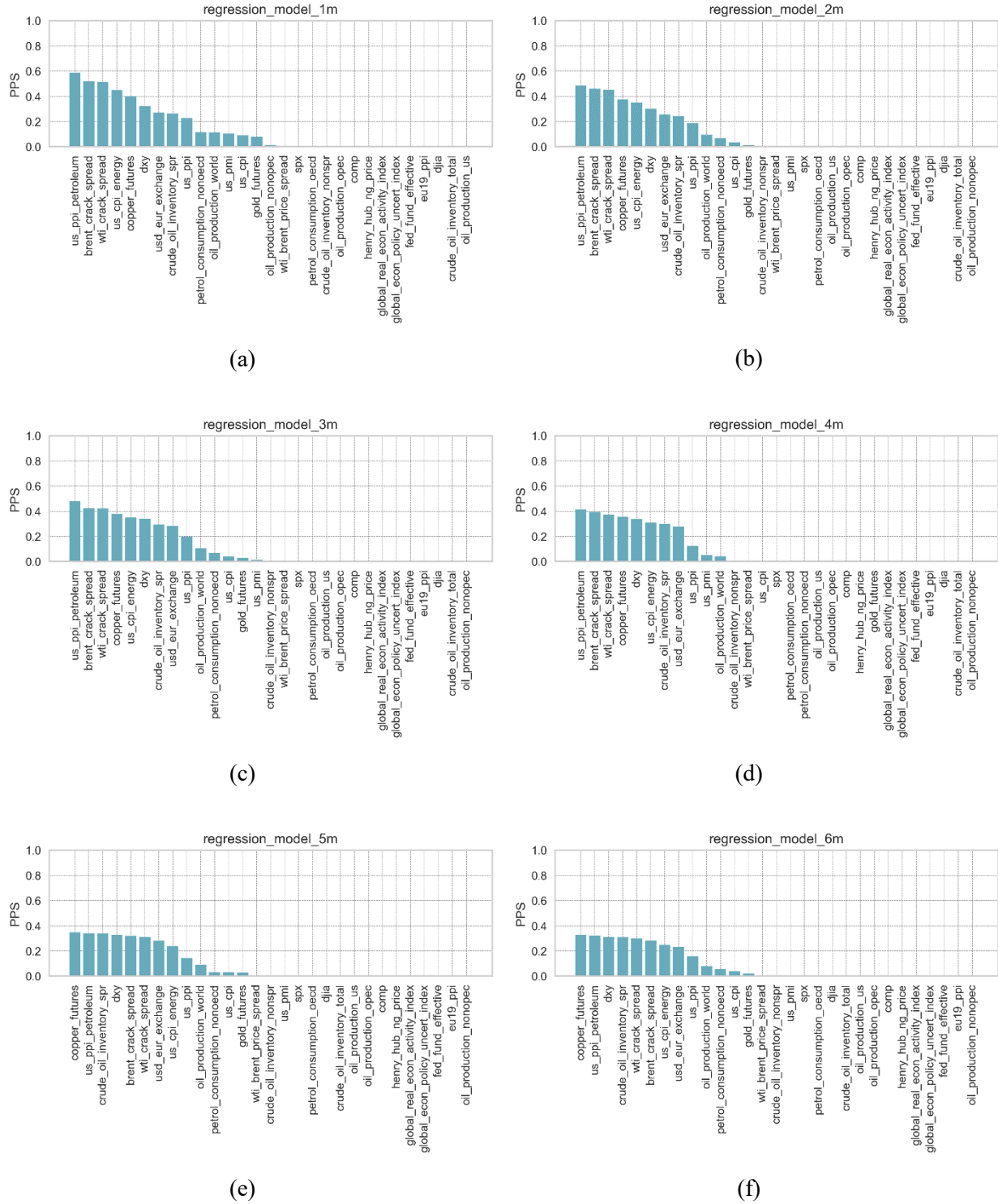
140

(a)

(b)

(c)

(d)

(e)

(f)

Figure 7. Bar chart of sorted PPS coefficients of independent variables against WTI oil price multi-frequency time scenarios; a) 1-month, b) 2-month, c) 3-month, d) 4-month, e) 5-month, f) 6-month

145   Training a model on one dataset and testing it on the same data is a methodological mistake

146   that causes overfitting. The model will learn from the provided sample and adopt to the training

147   data, but it may fail if a different dataset is feed to the model. By partitioning the data into train/test

148   splits, without letting the model to learn about the test split, it is possible to avoid overfitting. Also,

149   the performance of the model can be independency assessed by developing quality metrics based

150   on prediction of the test data (Pedregosa et al. 2011). The train/test split is also critical when

151   optimizing the model estimators by evaluating different settings ("hyper-parameters") via cross-

152   validation (CV). This approach is also called k-fold CV, where the training set is split into k

153   smaller sets. Then the model is trained using $k - 1$ folds as training data and the resulting model

154   is validated on the remaining part of the data to optimize the model. Finally, the final evaluation

155   is made based on the performance of the unseen test data (Pedregosa et al. 2011; Refaeilzadeh,

156   Tang, and Liu 2009). The train/test split and hyper-parameterization is discussed in training sub-

157   section.

158   Normalization or scaling of datasets is a commonly required for modelling. As previously

159   presented, the data has various units; however, the variables should be scaled into a common unit.

160   This is usually achieved by removing the mean value of each feature, and dividing the by their

161   standard deviation. It is recommended to find scaling parameters from the train then apply to the

162   test dataset to practically keep them unseen. Figure 8 shows boxplot of scaled train data.
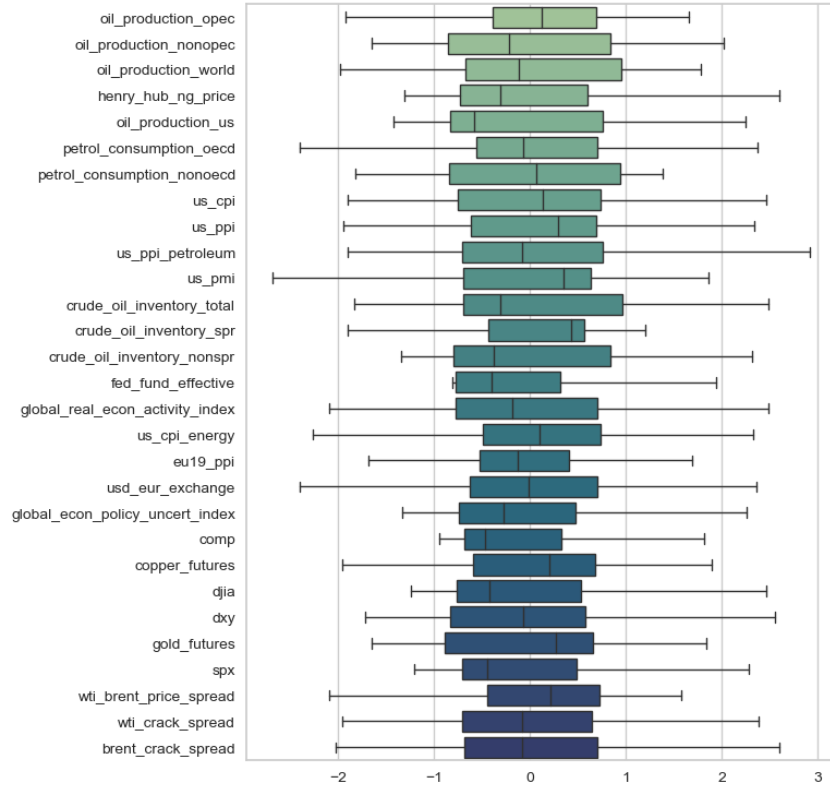
Figure 8. Boxplots of scaled train datasets

- *Training*

We use PyCaret, an open-source low code Python library that automates machine learning (AutoML) models to construct and deploy the models (Moez 2022). The library manages twenty-five different algorithms for regression, such as Extra Trees Regressor (E.T.), Gradient Boosting Regressor (GBR), Extreme Gradient Boosting Regressor (XGB), Random Forest Regressor (R.F.), Linear Regression (L.R.), AdaBoost Regressor (ADA), and eighteen other algorithms for classification. We compare the performance of twenty-five AutoML models based on coefficient of determination (R^2), Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). (Ardabili, Mosavi, and Várkonyi-Kóczy 2020).

PyCaret offer settings for train/test splits, missing value imputation methods, scaling, normalziation, etc. Table 4 shows the Pycaret regression session settings, where the data is configured to train/test split by 70/30, and transformation and normalization to be applied.

15

177 **Table 4. PyCaret regression session settings**

Description	Value
Session id	786
Target	wti_price
Target type	Regression
Data shape	(270, 30)
Train data shape	(188, 30)
Test data shape	(82, 30)
Numeric features	29
Preprocess	True
Imputation type	simple
Numeric imputation	mean
Categorical imputation	constant
Low variance threshold	0
Transformation	True
Transformation method	yeo-johnson
Normalize	True
Normalize method	zscore
Fold Generator	KFold
Fold Number	10

178

179   After setting the PyCaret session, preformed the initial modeling on multiple time

180 frequencies. For each scenario we compared the performance of ML models with 5-fold of cross-

181 validation generator and stored the top 5 best performance models to be later used to build

182 ensemble models. Table 5 shows the performance of ML models on the predicting WTI price for

183 1-month frequency scenario.

184

Table 5. Performance of ML models for 1-month frequency scenario

	Model	MAE	MSE	RMSE	R^2	RMSLE	MAPE
et	Extra Trees Regressor	4.51	40.47	6.32	0.95	0.12	0.08
gbr	Gradient Boosting Regressor	5.10	48.24	6.85	0.94	0.12	0.09
lightgbm	Light Gradient Boosting Machine	5.22	53.37	7.25	0.93	0.12	0.09
rf	Random Forest Regressor	5.30	55.06	7.38	0.93	0.13	0.10
lr	Linear Regression	5.90	56.68	7.49	0.93	0.16	0.12
ada	AdaBoost Regressor	5.80	59.57	7.63	0.92	0.13	0.11
huber	Huber Regressor	5.81	58.89	7.64	0.92	0.16	0.12
ridge	Ridge Regression	5.94	59.68	7.69	0.92	0.16	0.12
br	Bayesian Ridge	6.02	62.60	7.88	0.92	0.16	0.12
lasso	Lasso Regression	6.26	68.05	8.22	0.91	0.16	0.12
knn	K Neighbors Regressor	5.89	73.74	8.42	0.90	0.14	0.11
omp	Orthogonal Matching Pursuit	6.31	75.90	8.70	0.90	0.17	0.12
en	Elastic Net	6.86	79.07	8.86	0.89	0.17	0.13
dt	Decision Tree Regressor	6.64	84.37	9.07	0.88	0.15	0.12
par	Passive Aggressive Regressor	8.66	132.63	10.90	0.82	0.31	0.19
llar	Lasso Least Angle Regression	12.88	249.70	15.76	0.67	0.28	0.26
dummy	Dummy Regressor	23.21	756.93	27.45	-0.01	0.48	0.50

Then, we performed hyper-parameterization by dynamically optimizing the top 5 models based on RMSE value through 120 iterations with 5-fold CV (totaling 600 fits). Next, we select the optmized top five models for each time-frequency and build ensemble models. Ensemble methods benefit different training algorithms for increasing the training accuracy for reaching a higher testing accuracy to substantially improve the accuracy of the integerated model (Ardabili, Mosavi, and Várkonyi-Kóczy 2020). The most common ensemble methods are votting regressor and stacking regressor. Votting regressor uses a majority vote to build consensus of final prediction values. Stacking uses meta-learning to create multiple base estimators to generate the final prediction (An and Meng 2010; Džeroski and Ženko 2004).
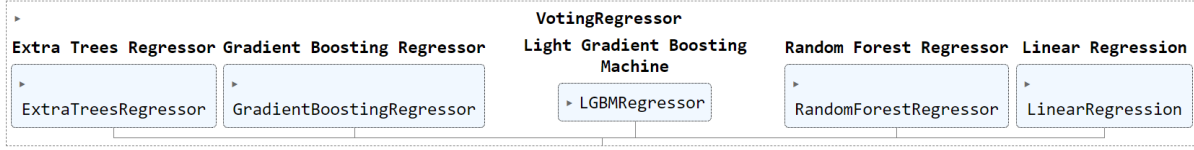
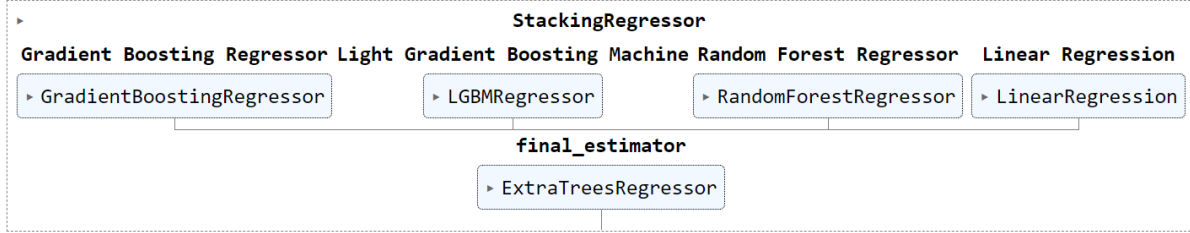Figure 9 shows voting and stacking regression ensemble model of the top 5 ML models, and Table 6 summarizes the performance metrics on test datasets of the ensemble models for 1-month frequency WTI price prediction.

(a)

(b)

199   **Figure 9. Ensemble models of top 5 ML models for 1-month frequency scenario; a) voting**
200   **regressor, and b) stacking regressor**

201
202   After comparing the performance of the ensemble model on test datasets, the best model

203   was stored and the final model for that scenario. Table 6 and Figure 10 show side-by-side

204   performance evaluation for voting and stacking regressor of 1-month scenario, where stacking

205   regressor was found to have better performance in this case.

206

207   **Table 6. Performance metrics on test datasets for voting and stacking regressor on 1-month**
208   **frequency scenario**

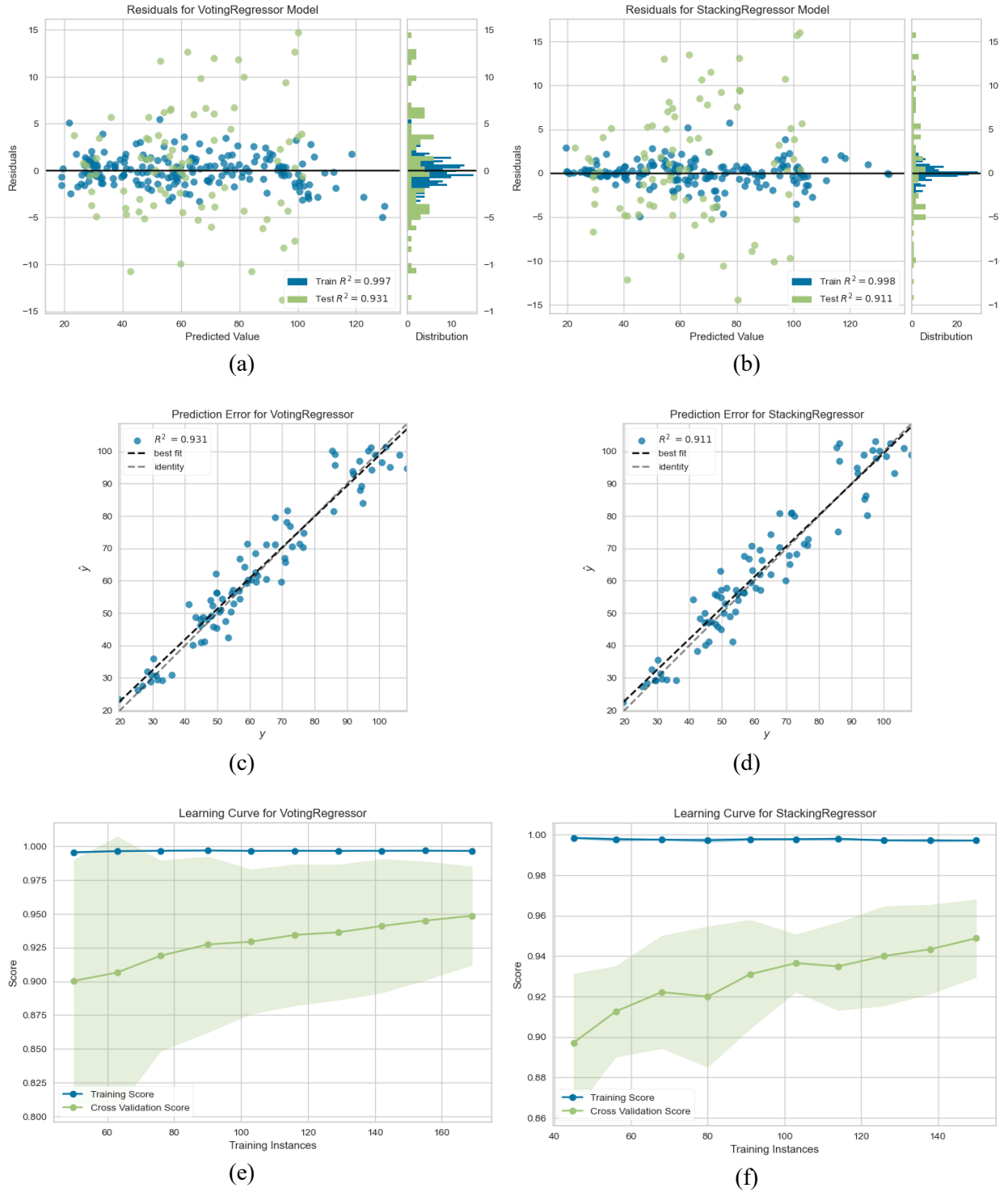| Ensemble model	Metric	MAE	MSE	RMSE	R2	RMSLE	MAPE
Votting Regressor	Mean	4.5984	38.9276	6.1874	0.9486	0.1129	0.0865
	Std	0.5388	10.0889	0.8024	0.01	0.0343	0.0234
Stacking Regressor	Mean	4.5974	39.7895	6.1574	0.9475	0.1106	0.0848
	Std	0.8137	17.5949	1.3699	0.0212	0.041	0.0281

209
210
211

212

Figure 10. Size-by-side evaluation of ensemble model performances of 1-month frequency scenario; a) residuals for voting regressor, b) residuals for stacking regressor, c) prediction error for voting regressor, d) prediction error for stacking regressor, e) learning curve for voting regressor, and d) learning curve for stacking regressor

19

218   The aforementioned steps on training, optimization, ensemble modeling was performed for

219   1- to 6-month time frequency scenario and the best performance model and it performance metrics

220   stored for final modeling.

221   **2.3. Modelling**

222   Table 7 shows the final selected ensemble models for the 6 frequency scenarios, where

223   voting regressor was used to build the 2- and 4-month frequncies and stacking regressor was used

224   for the other four scenarios. For the final models, the performances were analyzed on the entire

225   dataset including both test and train sets to calculate the performance metrics (Table 7).

226   **Table 7. Performance of final models on entire dataset**

| Model	Ensemble	MAE	MSE	RMSE	R2	RMSLE	MAPE
1-month	Stacking Regressor	0.9828	1.9863	1.4094	0.9971	0.0246	0.0164
2-month	Voting Regressor	1.4974	4.0136	2.0034	0.9941	0.0385	0.0273
3-month	Stacking Regressor	1.3449	4.1664	2.0412	0.9938	0.0344	0.0224
4-month	Voting Regressor	1.8882	6.3462	2.5192	0.9905	0.0503	0.0346
5-month	Stacking Regressor	1.2967	5.0315	2.2431	0.9925	0.0553	0.025
6-month	Stacking Regressor	1.2893	5.3536	2.3138	0.9920	0.0585	0.0253

227

228   Figure 11.a shows the distribution of actual and predicted values for WTI prices using the

229   final model for 1-month scenario. Figure 11.b presents the linear relationship between the actual

230   and predicted values from the final model for 1-month scenario. Figure 11.c is the scatter plot of

231   residuals and the predicted values for the same scenario. Similar plots can be generated for the rest

232   of the five scenarios to visually show the prediction performance of each model.

233
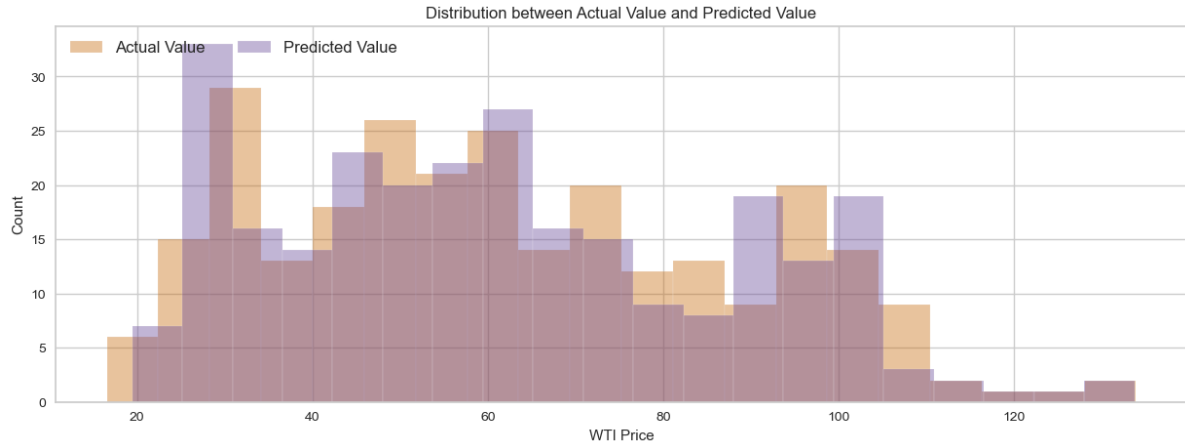
(a)
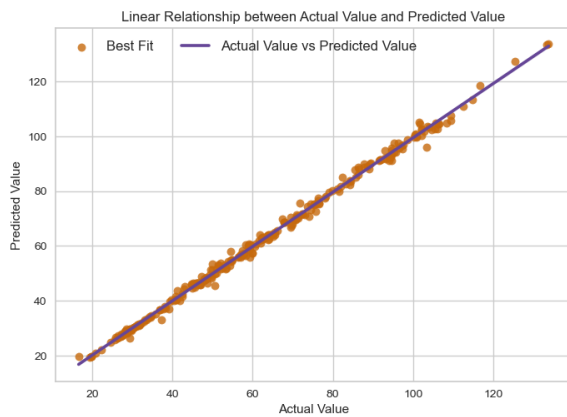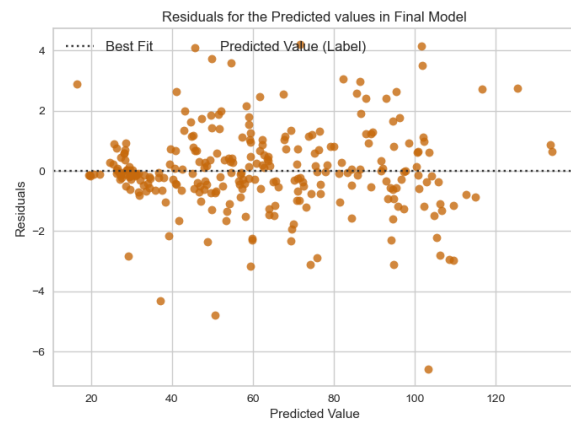
(b)   (c)

Figure 11. The performance of final model for 1-month scenario; a) distribution between actual vs predicted values, b) linear relationship between actual and predicted values, and c) residuals for the predicted values

The predicted values using final models for multi-frequency scenarios are plotted as time-series in Figure 12 together with the actual values of WTI prices. Apparently, the of minimum and maximum turning points of the oil price introduced larger errors to the models e.g., 2007/01, 2008/06, 2009/01, 2015/03, 2018/12, 2020/04, and 2022/05 (Figure 12.a).

Figure 12.b is a zoomed view of Figure 12.a that shows the recent WTI prices. As presented in the "data cleaning" section, dealing with the missing data required dropping a number of records

21

244   from the entire dataset; therefore, 2022-07 is the last data point in the cleaned dataset, while the

245   WTI price records already exists. The time-series in Figure 12 are extended by 6 data points to be

246   able to show the 6-month ahead prediction based on last data point. That is why each multi-

247   frequency series expands by additional data points beyond the last data point, e.g., 6 months

248   extension for 6-month frequency model.

249

(a)

(b)
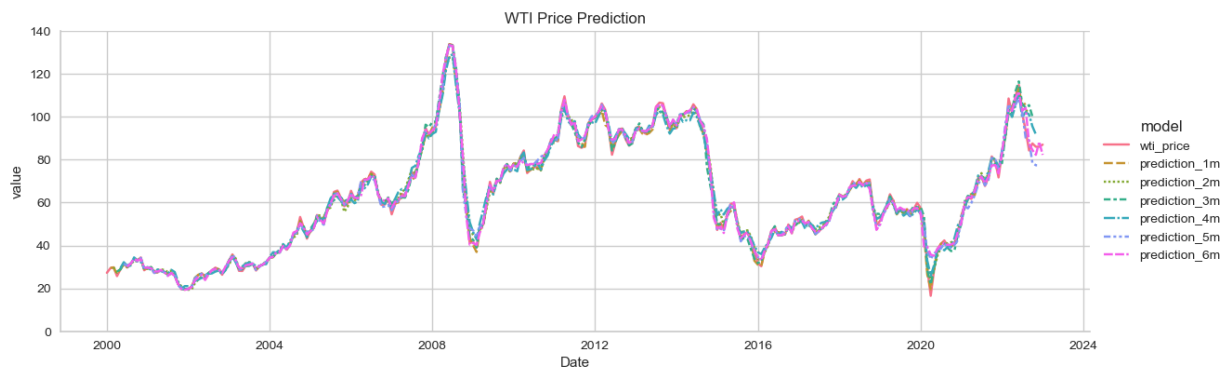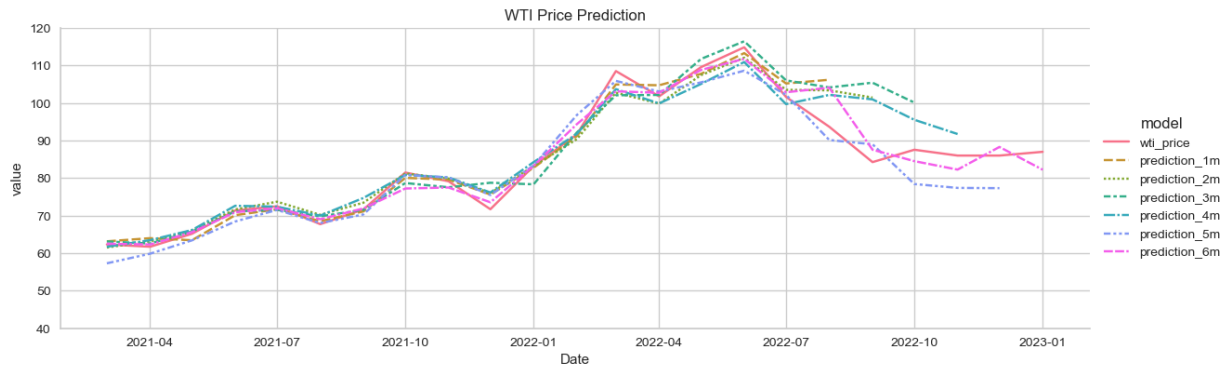
250   **Figure 12.**

251
252   As mentioned before, fluctuations in multiple factors may influence the crude oil market

253   dynamics, while the impact can be very complex and may occur at different frequencies. To

254   account for this multi-frequency impact, we developed multi-frequency modelling approach. Here,

22

255   we use the multiple predicted values for future prices for the crude oil and calculate a weighted

256   average prediction value (Eq. 5).

$$\bar{X} = \frac{\sum_{i=1}^{n} w_i X_i}{\sum_{i=1}^{n} w_i} \tag{5}$$

257   where, $\bar{X}$ is the weighted arithmetic mean for the new prediction data point, n is the number of

258   values to be averaged, X_i is the data value to be averaged, and w_i is the weight value that is

259   calculated using Eq. 6:

$$w = 1/\overline{RMSE} \tag{6}$$

260   where, $\overline{RMSE}$ is the average RMSE of the multiple models at the prediction data point that is

261   calculated using Eq. 7.

$$\overline{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} RMSE_i^2}{n}} \tag{7}$$

262   where, $RMSE_i$ is the RMSE for each model at the prediction data point.

263   To plot the lower and upper bounds of the prediction, the RMSE for each data point is used

264   to calcite the confidence intervals of the prediction using Eq. 8:

$$C.I. = \pm\, z_{1-0.01/2} \times RMSE \tag{8}$$

265
266   Figure 13 is the visualization of final averaged model, with upper and lower bound

267   confidence interval, and the actual values for WTI price in the unmodeled future prices.
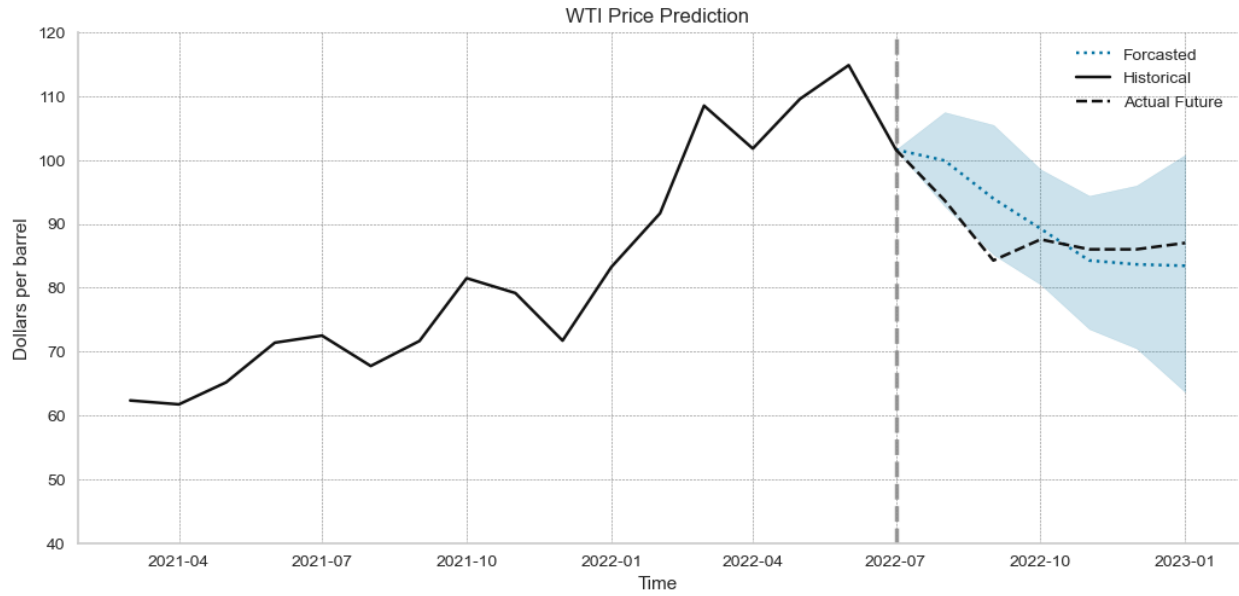
Figure 13. Visualization of final averaged model, with an upper and lower bound confidence interval, and the actual values for WTI price in the unmodeled future price.

3. Results and Discussion

- And end-to-end workflow for predicting the WTI crude oil pricing using AutoML modelling is introduced.

- We used data from publically aviable resources to extract values for 33 feaures for the timeline between Jan 2000 to December 2022.

- After data cleaning and missing data imputation, final dataset sized (274, 32) that represent 274-month records for 32 feature variables.

- We exploted the data using explorotary data analysis techniques and presented the relationa and correlations between the featues. We also created three new featues by feature rengineerring techniques.

24

- We used PyCaret, to pre-process, split test/train sets, and normalize the data data. Then we used the library to build, deploy and evaluate auto machine learning (AutoML) models on the data

- We build 1-6 month multi-frequency models WTI price predicitn. For each scenario we compared the performance of ML models with hypermaramerization and stored the top 5 best performance models to build ensemble models

- Finally, we used the multiple predicted values for future prices for the crude oil and calculate a weighted average prediction value and plotted with lower and upper bound confiencede internals.

- Plotting the actuial WTI against the predicted values shows that the model is susccesful in forecasting the oil price.

4. Recommendations and Future Work

- For future studies, it is rcoommdend to perform modeling using other machine learning method such as long short-term memory (LSTM), random walk (RW), autoregressive integrated moving average models (ARMA), elman neural Networks (ENN), ELM Neural Networks (EL), Walvet Neural Networks (WNN) and generalized regression neural network (GRNN).

References

An, Kun, and Jiang Meng. 2010. "Voting-Averaged Combination Method for Regressor Ensemble." In *International Conference on Intelligent Computing*, 540–546.

Ardabili, Sina, Amir Mosavi, and Annamária R. Várkonyi-Kóczy. 2020. "Advances in Machine Learning Modeling Reviewing Hybrid and Ensemble Methods." In , 215–227. doi:10.1007/978-3-030-36841-8_21.

Basiri, Mohammad Hossein; Javadnejad, Farshid; Saeedi, Azita. 2015. "Forecasting Crude Oil Price with an Artificial Neural Network Model Based on a Regular Pattern for Selecting of the Training and Testing Sets Using Dynamic Command-Line Functions." In *24th International Mining Congress and Exhibition of Turkey-IMCET'15*.

Bilogur, Aleksey. 2018. "Missingno: A Missing Data Visualization Suite." *The Journal of Open Source Software* 3 (22). The Open Journal: 547. doi:10.21105/JOSS.00547.

Deng, Chao, Liang Ma, and Taishan Zeng. 2021. "Crude Oil Price Forecast Based on Deep Transfer Learning: Shanghai Crude Oil as an Example." *Sustainability* 13 (24): 13770. doi:10.3390/su132413770.

Džeroski, Saso, and Bernard Ženko. 2004. "Is Combining Classifiers with Stacking Better than Selecting the Best One?" *Machine Learning* 54 (3). Springer: 255–273.

EIA. 2022a. "Opendata - U.S. Energy Information Administration (EIA)." https://www.eia.gov/opendata/.

EIA. 2022b. "International Energy Statistics, Total Oil Production." *U.S. Energy Information Administration*.

FRED. 2022. "St. Louis Fed Web Services: FRED® API." https://fred.stlouisfed.org/docs/api/fred/.

Hamilton, James. 2008. *Understanding Crude Oil Prices*. Cambridge, MA. doi:10.3386/w14492.

Hamilton, James. 2009. *Causes and Consequences of the Oil Shock of 2007-08*. Cambridge, MA. doi:10.3386/w15002.

Hou, Aijun, and Sandy Suardi. 2012. "A Nonparametric GARCH Model of Crude Oil Price Return Volatility." *Energy Economics* 34 (2): 618–626. doi:10.1016/j.eneco.2011.08.004.

Investing. 2022. "Investing.Com - Stock Market Quotes & Financial News." https://www.investing.com/.

Jammazi, Rania, and Chaker Aloui. 2012. "Crude Oil Price Forecasting: Experimental Evidence from Wavelet Decomposition and Neural Network Modeling." *Energy Economics* 34 (3): 828–841. doi:10.1016/j.eneco.2011.07.018.

Javadnejad, Farshid. 2012. "Presenting a Model for Prediction of Crude Oil Price Based on Artificial Intelligent Hybrid Methods and Time-Series." Tarbiat Modares University.

Kilian, Lutz, and Daniel P. Murphy. 2014. "THE ROLE OF INVENTORIES AND SPECULATIVE TRADING IN THE GLOBAL MARKET FOR CRUDE OIL." *Journal of Applied Econometrics* 29 (3): 454–478. doi:10.1002/jae.2322.

Kluyver, Thomas, Benjamin Ragan-Kelley, Fernando Pérez, Brian E Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, et al. 2016. *Jupyter Notebooks: A Publishing Format for Reproducible Computational Workflows*. Vol. 2016. https://jupyter.org/.

Lanza, Alessandro, Matteo Manera, and Massimo Giovannini. 2005. "Modeling and Forecasting Cointegrated Relationships among Heavy Oil and Product Prices." *Energy Economics* 27 (6): 831–848. doi:10.1016/j.eneco.2005.07.001.

Lu, Quanying, Shaolong Sun, Hongbo Duan, and Shouyang Wang. 2021. "Analysis and Forecasting of Crude Oil Price Based on the Variable Selection-LSTM Integrated Model." *Energy Informatics* 4 (S2): 47. doi:10.1186/s42162-021-00166-4.

McKinney, Wes. 2011. "Pandas: A Foundational Python Library for Data Analysis and Statistics." *Python for High Performance and Scientific Computing* 14 (9). Seattle: 1–9. https://pandas.pydata.org/.

Moez, Ali. 2022. "PyCaret: An Open Source, Low-Code Machine Learning Library in Python."

Murat, Atilim, and Ekin Tokat. 2009. "Forecasting Oil Price Movements with Crack Spread Futures." *Energy Economics* 31 (1): 85–90. doi:10.1016/j.eneco.2008.07.008.

Pedregosa, F, G Varoquaux, A Gramfort, B Michel V. and Thirion, O Grisel, M Blondel, R Prettenhofer P. and Weiss, et al. 2011. "Scikit-Learn: Machine Learning in Python." *Journal of Machine Learning Research* 12: 2825–2830.

Python Software Foundation. 2022. "Python." Python Software Foundation (PSF). https://www.python.org/.

Refaeilzadeh, Payam, Lei Tang, and Huan Liu. 2009. "Cross-Validation." *Encyclopedia of Database Systems* 5. Springer: 532–538.

Wang, Yudong, Chongfeng Wu, and Li Yang. 2015. "Forecasting the Real Prices of Crude Oil: A Dynamic Model Averaging Approach." *SSRN Electronic Journal*. doi:10.2139/ssrn.2590195.

WSJ. 2022. "The Wall Street Journal - Breaking News, Business, Financial & Economic News, World News and Video." https://www.wsj.com/.

Yu, Lean, Yang Zhao, and Ling Tang. 2017. "Ensemble Forecasting for Complex Time Series Using Sparse Representation and Neural Networks." *Journal of Forecasting* 36 (2): 122–138. doi:10.1002/for.2418.

Zhao, Yang, Jianping Li, and Lean Yu. 2017. "A Deep Learning Ensemble Approach for Crude Oil Price Forecasting." *Energy Economics* 66 (August): 9–16. doi:10.1016/j.eneco.2017.05.023.