CSC 522 Fall'23 ALDA Midterm Report Project Group 5

Raj Madhu
North Carolina State University
North Carolina
rmadhu3@ncsu.edu

Soham Bapat
North Carolina State University
North Carolina
sbapat2@ncsu.edu

Pankaj Thakur
North Carolina State University
North Carolina
pmanoha@ncsu.edu

Kanv Khare
North Carolina State University
North Carolina
kkhare@ncsu.edu

Abstract

This research paper aims to contribute to the field of oil price prediction by merging diverse data sets encompassing critical dimensions of this complex phenomenon. Focusing on West Texas Intermediate crude oil prices, a prominent global oil benchmark known for its superior quality, our study integrates historical price data, global demand and supply trends, geopolitical events, and environmental factors into a comprehensive analysis. Our core hypothesis posits that certain key features, such as the prices of gold, copper, euros, and the indices NASDAQ and SP500, play a pivotal role in forecasting oil prices. To achieve this goal, we employ a variety of modeling techniques, each utilizing distinct sets of features, in order to elucidate the most accurate predictive model. This research not only seeks to enhance our understanding of the intricate dynamics influencing crude oil prices but also holds the potential to inform critical decision-making processes for a wide array of stakeholders in the global energy landscape.

1 Introduction

In an era characterized by the profound significance of oil exploration and exploitation within the global economy, the ability to predict oil prices has become an essential endeavor. The stability and prosperity of numerous individuals, industries, and government entities hinge on the precise forecasting of crude oil prices, a task made daunting by the inherently volatile and ephemeral nature of this critical commodity. A myriad of factors, spanning economic conditions, global events, and the prices of precious metals and agricultural products, can exert their influence on oil prices. However, establishing a steadfast correlation between these factors and crude oil prices remains a formidable challenge. With the intricacies of predicting oil prices becoming increasingly apparent, this research endeavors to investigate the factors that influence these prices. It employs various feature selection and elimination techniques to discern the variables impacting crude oil prices, and subsequently, it evaluates the efficacy of these methods through the implementation of machine learning models for time-series data prediction. This study aims to amalgamate diverse datasets related to oil prices, encompassing historical price data, global demand and supply trends, geopolitical events, and environmental factors. Furthermore, it hypothesizes that specific features, such as the price of gold, copper, euros, and indices like NASDAQ and SP500, play a pivotal role in predicting oil prices. By exploring a range of techniques and feature sets, this research endeavors to determine the most accurate models for forecasting oil prices, underpinned by the West Texas Intermediate crude oil benchmark, renowned for its exceptional quality and global significance.

2 Data Handling

One of the main part of this project is to collect data that some correlation with WTI oil prices. We read various research papers [1] [4] [5] and searched on google to find some of the features that might have some correlation with Oil prices. The details about these features and their respective data collection is given in Table 2.

2.1 Data Collection

The data about WTI oil prices was collected from US Energy Information Administration website and from Yahoo finance. We research how different papers collected their data on oil prices and we found that most used the official US government website to collect their data so we also decided to use that. We also decided on other factors that might correlate with oil prices and we got all those from the Yahoo Finance website. We built a data fetching bot in python that can fetch data from the Yahoo Finance Website and save that data into a CSV file. The feature list and their source is given in Table 2:

2.2 Data Preprocessing

2.2.1 Handling Null or Missing Values

- Upon collecting data for each feature, the first step involves a thorough examination to identify and address any null or missing values.
- Dealing with missing data is crucial for maintaining the integrity of the dataset and ensuring accurate analyses.

2.2.2 Method for Handling Missing Data

- The chosen method for handling missing data involves using the ’bfill()’ method from the Pandas library.
- The ’bfill()’ method stands for backward fill and replaces null values with the most recent non-null value in the dataset.
- This method helps in maintaining the temporal order of the data, especially when dealing with time-series information.

2.2.3 Extraction of Columns

- After addressing missing values, the ’Date’ and ’Value’ columns are extracted from each individual dataset.
- The ’Date’ and ’Value’ columns are likely considered essential for subsequent analysis and are singled out for further processing.

2.2.4 Merge Operatin Based on ’Date’ Column:

- The extracted ’Date’ and ’Value’ columns from each dataset are used in a merge operation.
- The merge operation is performed based on the ’Date’ column, suggesting that the datasets are combined or consolidated based on the common dates.
- This step is crucial for creating a unified dataset that incorporates information from multiple sources.

2.2.5 Resulting Dataset:

- The outcome of the merge operation is a dataset that now comprises 13 different features along with the ’Date’ column.
- These 13 features are presumably a combination of the initially collected features from different datasets.

Table 2: Feature descriptions and their source

Feature	Details	Source
WTI_dollar_per_barrel	WTI Crude Oil Spot Price is the price for immediate delivery of West Texas Intermediate grade oil, also known as Texas light sweet. It, along with Brent Spot Price, is one of the major benchmarks used in pricing oil. WTI in particular is useful for pricing any oil produce in the Americas.	EIA
copper_close	The close price of Copper	Yahoo Finance
dji_index	The Dow Jones Industrial Average, Dow Jones, or simply the Dow, is a stock market index of 30 prominent companies listed on stock exchanges in the United States.	Yahoo Finance
gold_close	The close price of Gold.	Yahoo Finance
eur_close	The ratio of US Dollar to Euros.	Yahoo Finance
Henry Hub Natural Gas	Henry Hub Natural Gas Spot Price in Dollar per Million Btu.	Yahoo Finance
rub_close	The ratio of US Dollar to Russian Ruble.	Yahoo Finance
silver_close	The close price of Silver.	Yahoo Finance
nasdaq_close	The NASDAQ Composite is a stock market index that includes all companies listed on the NASDAQ stock exchange, featuring technology and various other sectors. It serves as a benchmark for the overall performance of the NASDAQ market.	Yahoo Finance
SP500	The S&P 500, or Standard & Poor's 500, is a prominent stock market index in the United States, tracking the performance of 500 of the largest publicly traded companies. It's a key indicator of the U.S. stock market and economic conditions.	Yahoo Finance
pal_close	The close price of Palladium.	Yahoo Finance
corn_close	Corn futures are financial contracts that allow traders to buy or sell a specified quantity of corn at a predetermined price on a future date. These futures contracts are traded on commodity exchanges, such as the Chicago Board of Trade (CBOT) in the United States.	Yahoo Finance
heat_close	Heating oil futures are derivative contracts that allow traders and investors to speculate on or hedge against the future price movements of heating oil.	Yahoo Finance

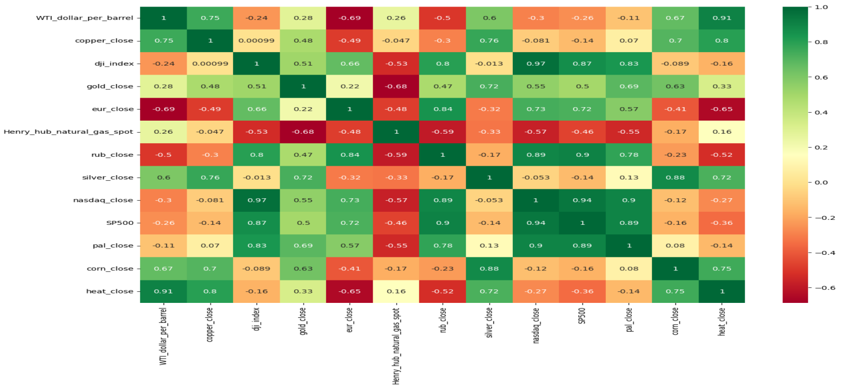
3 Methods

3.1 Feature Selection Methods

3.1.1 Heatmap / Correlation Matrix

A correlation matrix is a square matrix that displays the pairwise correlations between variables in a data-set. It provides a numerical measure of the strength and direction of the linear relationship between pairs of variables. To find the correlation matrix we calculate the Pearson correlation coefficient (r) between two variables and use that value in the matrix to show correlation between those two variables. The numeric value of 'r' can be between -1 to 1, where a negative value close to -1 suggest a strong negative correlation, value closer to 0 suggests a no correlation and a positive value closer to 1 suggest a strong positive correlation. The equation for Pearson correlation coefficient is given in Equation 1

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X}).\sum_{i=1}^{n}(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}.\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}} \tag{1}$$



	WTI_dollar_per_barrel	copper_close	dji_index	gold_close	eur_close	Henry_hub_natural_gas_spot	rub_close	silver_close	nasdaq_close	SP500	pal_close	corn_close	heat_close
WTI_dollar_per_barrel	1	0.75	-0.24	0.28	-0.69	0.26	-0.5	0.6	-0.3	-0.26	-0.11	0.67	0.91
copper_close	0.75	1	0.00099	0.48	-0.49	-0.047	-0.3	0.76	-0.081	-0.14	0.07	0.7	0.8
dji_index	-0.24	0.00099	1	0.51	0.66	-0.53	0.8	-0.013	0.97	0.87	0.83	-0.089	-0.16
gold_close	0.28	0.48	0.51	1	0.22	-0.68	0.47	0.72	0.55	0.5	0.69	0.63	0.33
eur_close	-0.69	-0.49	0.66	0.22	1	-0.48	0.84	-0.32	0.73	0.72	0.57	-0.41	-0.65
Henry_hub_natural_gas_spot	0.26	-0.047	-0.53	-0.68	-0.48	1	-0.59	-0.33	-0.57	-0.46	-0.55	-0.17	0.16
rub_close	-0.5	-0.3	0.8	0.47	0.84	-0.59	1	-0.17	0.89	0.9	0.78	-0.23	-0.52
silver_close	0.6	0.76	-0.013	0.72	-0.32	-0.33	-0.17	1	-0.053	-0.14	0.13	0.88	0.72
nasdaq_close	-0.3	-0.081	0.97	0.55	0.73	-0.57	0.89	-0.053	1	0.94	0.9	-0.12	-0.27
SP500	-0.26	-0.14	0.87	0.5	0.72	-0.46	0.9	-0.14	0.94	1	0.89	-0.16	-0.36
pal_close	-0.11	0.07	0.83	0.69	0.57	-0.55	0.78	0.13	0.9	0.89	1	0.08	-0.14
corn_close	0.67	0.7	-0.089	0.63	-0.41	-0.17	-0.23	0.88	-0.12	-0.16	0.08	1	0.75
heat_close	0.91	0.8	-0.16	0.33	-0.65	0.16	-0.52	0.72	-0.27	-0.36	-0.14	0.75	1

Figure 1: Correlation Matrix

From the Figure 1 we can see that the following features have correlation with WTI oil prices: **heat_close (0.91), copper (+0.75) and eur_close (-0.69)**

3.1.2 Xgboost Feature Importance

XGBoost (Extreme Gradient Boosting) is a popular machine learning algorithm for both classification and regression tasks. It is an ensemble learning method that uses decision trees as base learners. Feature importance in XGBoost refers to determining the significance of each feature (predictor or input variable) in making predictions. Knowing which features are important can help one understand the model better and potentially improve its performance. It's based on the gradient boosting framework. Boosting is a technique that combines weak learners to create a strong learner. XGBoost builds an ensemble of decision trees sequentially, with each tree trying to correct the errors made by the previous ones. The feature importance is calculated by weights. The weight importance is based on the number of times a feature is used to split data across all trees in the ensemble. So it measures the relative frequency with which a feature appears in tree nodes as a split criterion. Features that are used frequently to make splits are generally considered more important because they contribute more to the final predictions.

As seen in the Figure 2, the top features that have high correlation with WTI oil prices are:**Henry_hub_natural_gas_spot**, **Copper_close**, **heat_close**.

3.1.3 Feature Selection using Genetic Algorithms

We use Genetic Algorithms as another method for feature selection. We started with an initial population size of 30, with each individual in the population representing a set of features that we

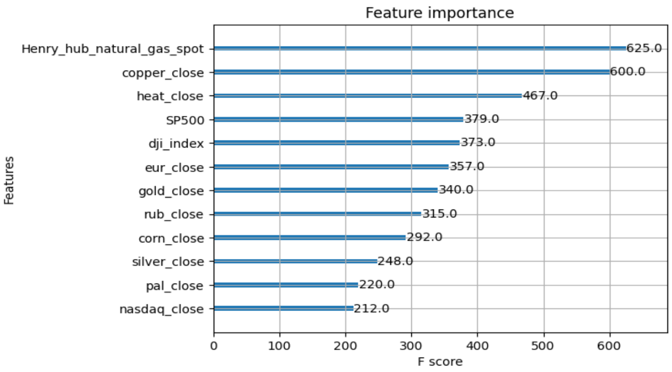


Figure 2: XGBoost Feature Importance

could potentially use for our model. This population was randomly generated. Next, we evaluated the fitness of each of the individuals in our population. This was done using three-fold cross validation with mean squared error (MSE) being our scoring metric that we were trying to minimize. A RandomForestRegressor was used as the model for evaluation. After evaluating the population, the genetic algorithm was run for twenty-five generations. For each generation, the individuals in the population underwent crossover and mutation based on their fitness. Crossover consisted of individuals being combined and occurred with a probability of 60%. Mutations randomly flipped features and had a 20% probability of occurring. We chose these numbers to encourage crossover, and also allow for genetic diversity via mutations. The fitness of the new population was then re-evaluated, and this process was repeated until the target of twenty-five generations had been reached. The features we got from this method of feature selection were **’copper close’, ’dji index’, ’nasdaq close’, ’pal close’, and ’heat close’.**

3.2 Feature Elimination Methods

3.2.1 Recursive Feature Elimination(RFE)

In Recursive Feature Elimination or RFE the model is trained recursively on all the features of the dataset and the least important features are eliminated. This process is continued until the desired number of features is reached. In our project, we employed the Random Forest Regressor as the model for training on the complete set of features, for feature ranking. Since Random Tree Regressor is a tree-based model, the significance of a feature was assessed based on its contribution to the reduction of impurity or error.

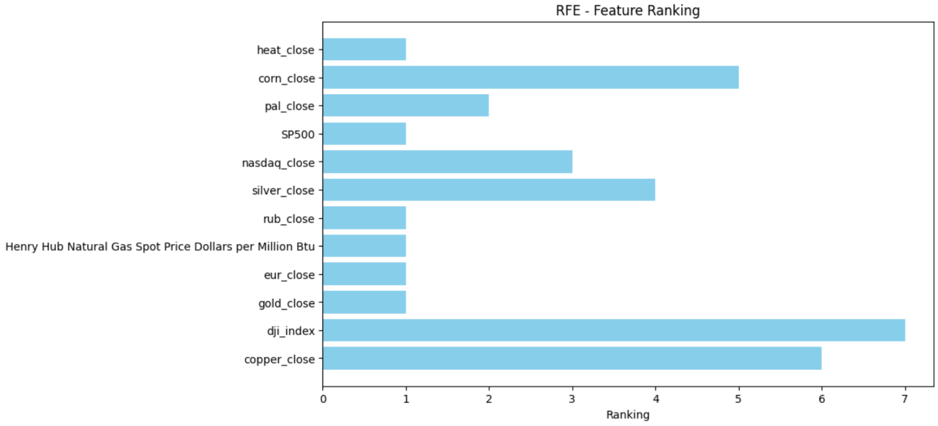


Figure 3: RFE Feature Importance

As seen in the above Figure 3, the top features with rank 1 are: **Gold_close**, **Eur_close**, **Henry_hub_natural_gas_spot**, **Rub_close**, **SP500**, **Heat_close**

3.2.2 Least Absolute Shrinkage and Selection Operator (Lasso) Method

Lasso, or Least Absolute Shrinkage and Selection Operator adds a penalty term to the linear regression cost function, which includes the absolute values of the coefficients. This penalty encourages the model to prefer a simpler model with fewer non-zero coefficients, effectively leading to automatic feature selection.
The Lasso Cost function is given by:

$$\text{Cost}(\beta) = \sum(Y - X\beta)^2 + \alpha \sum |\beta_i|$$

Here, β represents the coefficients of the model, and the second term $\alpha \sum |\beta_i|$ is the L1 penalty term.

Upon running the Lasso method, the following were selected as the best features: **Eur_close, Heat_close, copper_close, dji_index, henry_hub_natural_gas_spot_price**.

4 Machine Learning Model

After careful evaluation and reviewing research papers like [2] and [3] , we decided to use an ensemble of Random Forest model and KNN model.

4.1 Random Forest Model

Random Forest is an ensemble learning model that is widely used for both classification and regression tasks in machine learning. It involves constructing a multitude of models and combining them to make more accurate and robust predictions. Particularly, random Forest builds a forest of trees. Each tree is trained independently on a random subset of the data. It combines the predictions of these multiple decision trees to improve overall accuracy and robustness. This ensemble approach helps mitigate overfitting, increases model generalization, and provides a reliable method for handling noisy data. In our case we used the **RandomForestRegressor**() from the Sklearn library. We trained the model on **128 estimators** that is the model will build **128 decision** trees and predict the final continuous value.

The reason for picking this model was because this model gives high accuracy, it is robust to overfitting, handles Non-Linearity and Complex Relationships and also handle missing values.

4.2 KNN

K-Nearest Neighbors (KNN) is a lazy learning model that uses stored training data to predict future entries based. It looks for the k nearest points in the feature space of the training data, and classifies entries based on where they fall in the feature space in relation to the existing data points. For our KNN model, we used KNeighborsRegressor from the Sklearn library with a k value of 10 (meaning we looked at the 10 nearest neighbors). We used the Minkowski metric to compute distance between points and a 80/20 split between training and testing. The final model estimates a continuous value for a new data entry, using existing data points.

4.3 Ensemble Model

We decided to use an Ensemble of KNN and Random Forest model as it increases the accuracy by combining the predictions of multiple models, ensemble methods can reduce errors and improve overall performance. Ensemble models are also less prone to overfitting compared to individual models. Combined models tends to generalize well to new, unseen data. They are also capable of capturing complex relationships in the data and have the advantage of diversity among individual models. If each model in the ensemble has a different perspective or learns different aspects of the data, the ensemble as a whole can perform better than its individual components. They are also computationally efficient as instead of training one big complex model, we just have to train 2 simple models and combine them. It also saves a lot of time! The Mean Squared Error (MSE) that we got using the Ensemble model better than using individual models.

5 Results

Method	Features Selected	Average Mean Square Error
Heat Map	Heat Close, Copper Close, Euro Close	37.075
XGBoost	Henry Hub Natural Gas Spot Price Dollars per Million Btu, Copper Close, Heat Close	16.321
Generic	Copper Close, DJI Index, NASDAQ Close, PAL Close, Heat Close	30.434
Lasso	Copper Close, DJI Index, Henry Hub Natural Gas Spot Price Dollars per Million Btu, Eur Close, Heat Close	69.09
RFE	Gold Close, Eur Close, Henry Hub Natural Gas Spot Price Dollars per Million Btu, Rub Close, SP500, Heat Close	3.33

Table 3: Test Results

The table presents the results of different feature selection methods along with the corresponding features selected and Mean Square Error (MSE) values. The methods include Heat Map, XGBoost, Generic, Lasso, and Recursive Feature Elimination (RFE). The MSE is a measure of the average squared difference between predicted values and actual values, where a lower MSE indicates better model performance.

The Recursive Feature Elimination (RFE) method stands out as it yielded the lowest MSE of 3.33 among all methods. This suggests that the selected features from RFE contribute significantly to improving the model's predictive performance.

The features selected by RFE (Gold Close, Eur Close, Henry Hub Natural Gas Spot, Rub Close, SP500, Heat Close) seem to be more strongly correlated with the output variable (oil price) compared to features selected by other methods. This is evident from the substantially lower MSE, emphasizing the importance of these features in capturing the underlying patterns in the data.

6 Conclusion

Our project focuses on identifying key features that significantly impact the prediction of oil prices. Initially, we curated a comprehensive dataset by gathering various factors known to influence oil prices. This dataset comprised 12 features, including West Texas Intermediate crude oil prices, as follows: copper price, DJI index, Henry Hub natural gas, euro, gold price, ruble price, silver price, NASDAQ index, heat close, corn price, and palladium price.

To analyze this collected data, we employed both feature selection and elimination techniques. Our objectives were twofold: first, to determine the most effective method, and second, to identify the most influential features. Mean squared error served as the parameter for comparing these methods. Through experiments with the heatmap method, XGBoost feature importance method, genetic algorithm method, Lasso method, and Recursive Feature Elimination (RFE) method, we discovered that the RFE method yielded the lowest mean squared error, specifically 3.33.

In the RFE approach, each feature is assigned a rank based on its impact in minimizing the error value, and the feature with the lowest rank is systematically eliminated. This process continues until the desired number of features remains—in our case, six. The selected features were: `gold_close`, `eur_close`, `henry_hub_natural_gas_spot`, `rub_close`, `SP500`, and `heat_close`. RFE turns out to be the best approach because it is an iterative method to evaluate the features relevant to the target variable.

Subsequently, we constructed an ensemble model combining KNN and Random Forest, calculating the Mean Squared Error (MSE) for each individual model as well as the ensemble model. The ensemble model produced an MSE of 3.33. Overall, our study provides valuable insights for forecasting oil prices using these finalized features.

References

[1] Zhongpei Cen and Jun Wang. Crude oil price prediction model with long short term memory deep learning based on prior knowledge data transfer. *Energy*, 169:160–171, 2019.

[2] Yanhui Chen et al. Forecasting crude oil prices: A deep learning based model. *Procedia Computer Science*, 122:300–307, 2017.

[3] Junhui Guo. Oil price forecast using deep learning and arima. In *2019 International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)*, 2019.

[4] Xuan Zhang. Dynamic correlations between crude oil futures prices. *Energy Research Letters*, 3(1), 2022.

[5] Yingrui Zhou et al. A ceemdan and xgboost-based approach to forecast crude oil prices. *Complexity*, 2019:1–15, 2019.

7 Github Link

The code for the project can be found here:

https://github.ncsu.edu/kkhare/engr-ALDA-Fall2023-P5