Henry Hub Gas Price Forecasting

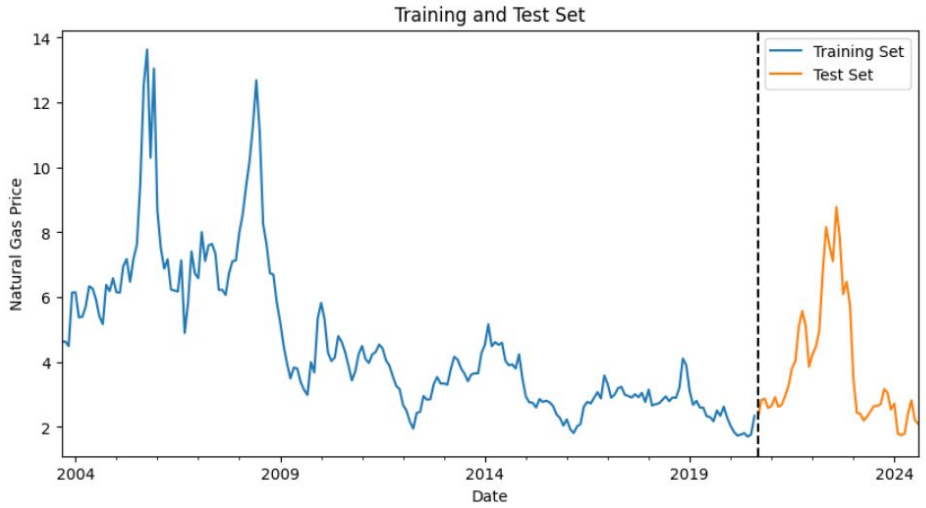
Azamat Rashidov, KAIST

- **Project Summary:**
  - This project aims to forecast the trend of Henry Hub natural gas prices over the next three months.
  - It involves identifying and analyzing potential influencing factors such as stock market activity, market behavior, and climate changes.

- **File Summary:**
  - This report presents a machine learning approach using XGBoost regression to forecast Henry Hub natural gas prices over a three-month horizon. The model incorporates diverse factors—such as market indicators, consumer confidence, and climate variables—and uses PCA to reduce dimensionality. After testing various lag configurations, a 1-period lag delivered the best performance, achieving an R^2 of 0.839, RMSE of 0.752, and MAPE of 15.36%. The model effectively captures overall trends and major fluctuations from 2020 to 2024 but underperforms during extreme events, notably the 2022 price spike. Visual analysis supports this, showing accurate predictions with occasional deviations at higher prices. Findings suggest short-term momentum is key, though further refinement is needed to better handle volatility.

Data Visualization and Processing

- The data contains missing values for many features. We address this by forward filling the missing values – use last available data for missing values. The missing values in the first row were filled with the immediate next available values in the column.
- We then plot the target data – `Natural Gas US Henry Hub Gas` column.



Data Scaling and Dimensionality Reduction

- We perform PCA to reduce the dimensionality of the dataset while preserving key variance. First, we standardize the data (excluding the target column) and determine the optimal number of principal components needed to retain the desired level of variance – 90%. Then, we apply PCA using this number of components and transform the original dataset into a set of principal components. This helps us understand the underlying structure of the data and reduce redundancy while keeping the most informative features. These are the selected features:

Most important original features (across all PCs):

Feature:	Explained Variance:
CLI_Comp_Monetary_Aggregates_IND	1.873161
CLI_Comp_Consumer_Confidence_GBR	1.624241
CLI_Comp_Construction_USA	1.594911
Chemical_Fertilizers_Manufactured_(actual_weight)_Accumulated_YoY	1.572320
Total_Natural_Gas_Underground_Storage_Volume_USA	1.567397
M_PMI_Comp_Main_Raw_Material_Purchase_Price	1.561806
CLI_Comp_Interest_Rate_Spread_JPN	1.547182
CLI_Comp_Interest_Rate_Spread_DEU	1.521297
CLI_Comp_Consumer_Confidence_USA	1.516464
Natural_Gas_Imports_From_Canada_USA	1.514590
M_PMI_Comp_Producer_Prices	1.494097
CLI_Comp_Interest_Rate_Spread_USA	1.482684
Services_PMI	1.450638
M_PMI_Comp_Expected_Production_and_Business_Activities	1.423821
M_PMI_Comp_Finished_Goods_Inventory	1.411467
Construction_PMI	1.408438

Creating Lagged Features

- We create lagged features to incorporate past information into our model, which is especially useful for time series forecasting. Lagging a feature means shifting its values down by a certain number of time steps, so each row includes values from previous periods. This helps the model learn temporal patterns and dependencies.

- We will conduct an experiment to evaluate how the number of lagged features affects model performance. To do this, we will make three separate predictions of gas prices, each using a different number of lagged features: 1, 3, and 5.

Model Selection

- XGBoost was selected for its strong performance in structured, tabular data and its ability to handle complex nonlinear relationships between features and target variables. Its built-in regularization helps prevent overfitting, which is particularly useful given the volatility and noise in natural gas price data. Its efficiency and scalability made it well-suited for experimenting with various lag configurations and large feature sets, especially after applying PCA for dimensionality reduction.

Model Training

- We train an XGBoost regression model to predict natural gas prices using lagged features. First, we create lagged versions of the features and the target variable, then split the data into training (up to August 2020) and testing sets (from September 2020). We train the model with specific hyperparameters and evaluate its performance using R^2, RMSE, and MAPE metrics. Finally, we return the trained model along with the test results and predictions.

- Although not detailed in this report, the selected hyperparameters are not arbitrary. We experimented with various combinations and chose the set that yielded better performance on the validation data.

Model Performance

- Here is the model performance for the three different lag values. As we can see, when `n_lags = 1`, the model achieves higher performance across the presented metrics. This suggests that incorporating only one lagged observation provides the most relevant past information for forecasting, while additional lags may introduce noise or redundancy.

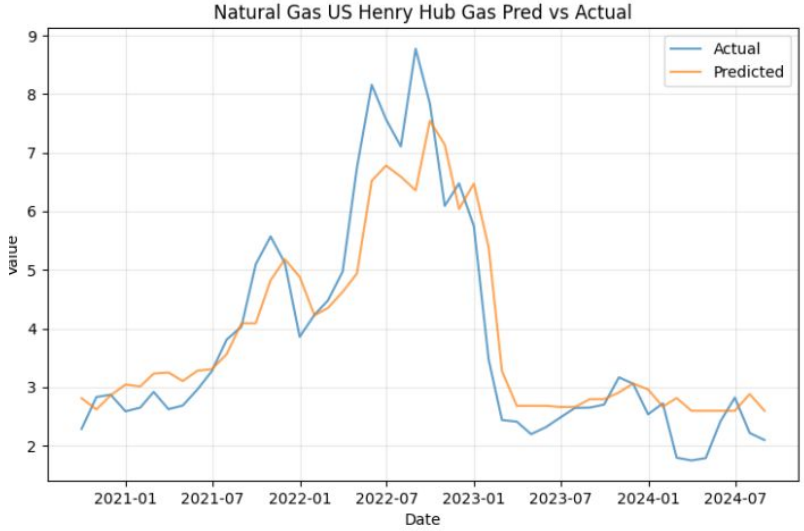
```
'Results for n_lags = 1'
R² - Train: 0.9184940525227263
R² - Test: 0.839126505279989
RMSE: 0.7518205720593814
MAPE: 0.15357304543930464
```

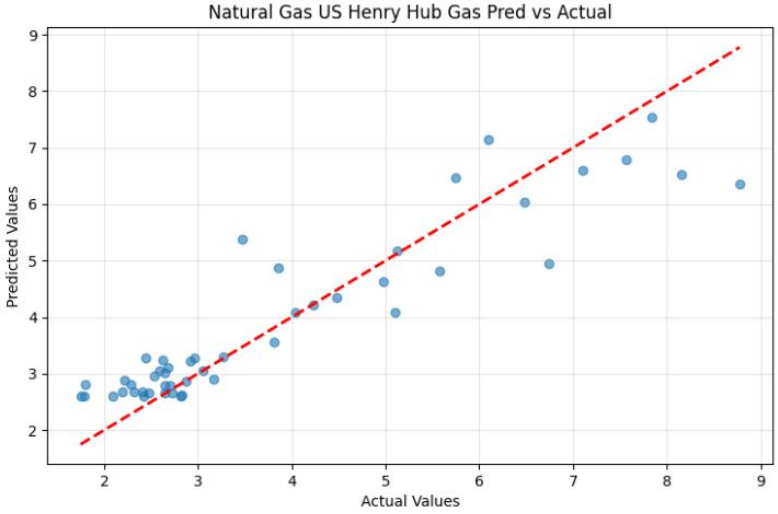
```
'Results for n_lags = 3'
R² - Train: 0.9305492700511172
R² - Test: 0.7926069938237946
RMSE: 0.8536287043930488
MAPE: 0.16425660836475128
```

```
'Results for n_lags = 5'
R² - Train: 0.9323932264073815
R² - Test: 0.7544390235072688
RMSE: 0.9288629340567183
MAPE: 0.18217537400275374
```

Best Performance Analysis

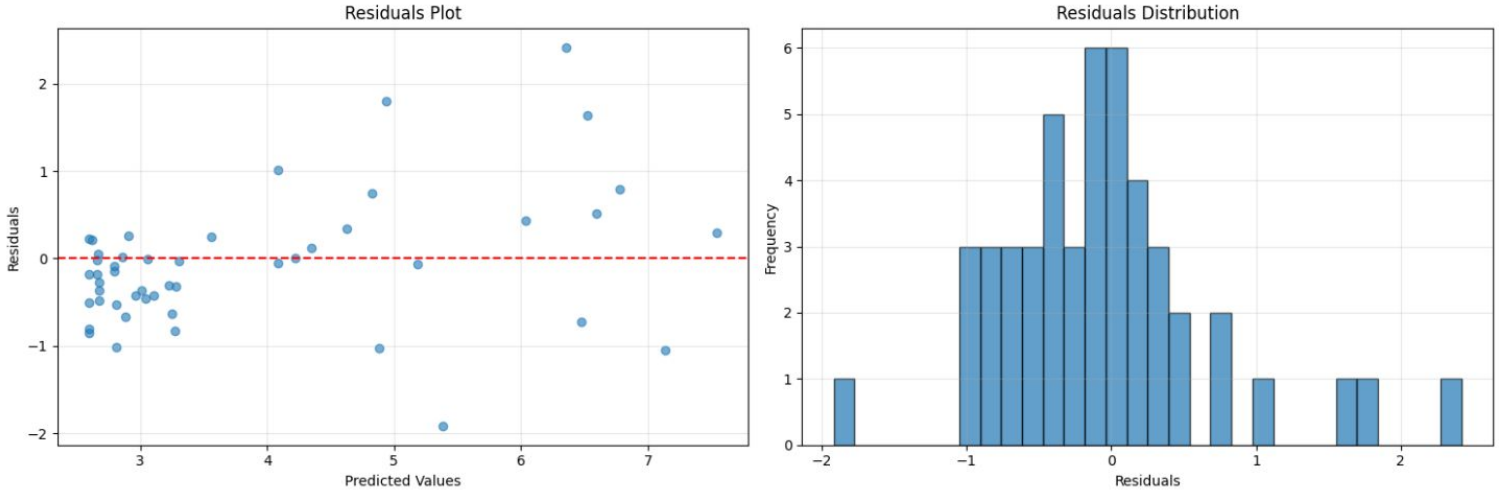
The left graph shows the predicted vs. actual values of the Natural Gas US Henry Hub Gas prices over time. We can see that the model captures the overall trend and major fluctuations in the gas prices, though there are some deviations in magnitude, especially around the peak in mid-2022. The right graph is a scatter plot comparing actual and predicted values. Each dot represents one prediction. The closer the dots are to the red dashed line (the ideal line where predicted = actual), the more accurate the predictions are. While the predictions follow the general trend, we observe that some values are underestimated or overestimated, especially at higher price levels.





Best Performance Analysis

The left plot is a residuals plot, which shows the residuals against the predicted values. Ideally, residuals should be randomly scattered around zero without any clear pattern. In this case, while many residuals are close to zero, we observe increasing variance and some noticeable outliers at higher predicted values, suggesting the model may underperform for larger gas price predictions. The right plot shows the distribution of residuals. Most residuals are concentrated around zero, forming a roughly bell-shaped distribution. This indicates that the model tends to make small errors on average, but the slight right skew suggests that a few predictions are significantly overestimated.



Shortcomings and Model Limitations

While the XGBoost model achieved reasonable R^2 scores (~0.84 on test data), several performance concerns emerge from the analysis. The model **struggles with high-price predictions**, as evidenced by increasing residual variance at higher values. The scatter plot reveals systematic underestimation during extreme price events, particularly around the 2022 peak, suggesting the model fails to capture tail risk events critical for energy market forecasting. Additionally, the residual plots exhibit clear **heteroscedasticity**, with variance increasing at higher predicted values, violating a key assumption for reliable predictions.

The significant performance drop between training (R^2 = 0.918) and test (R^2 = 0.839) scores indicates possible **overfitting**. Despite hyperparameter tuning, the model appears to have learned noise patterns from the training data that don't generalize well to future periods. This is further supported by the right-skewed residual distribution, suggesting the model makes systematic errors when encountering price patterns not well-represented in the training data.