Vessel Performance Model

Phase 1

Data Exploration & Preprocessing

- ***Data Ingestion***

Reading Multiple JSON Entries:

- Process: Data was ingested from multiple JSON files containing vessel reports.
- Method: Each JSON entry was parsed and converted into a tabular format suitable for analysis.

Handling Nested Fields:

Nested fields like Auxiliary Blowers, Other Robs, Main Engines, Consumptions, Fuel Robs, Generators, fuel type, 'Main Engine Cylinder Oil' , Main Engine Second Cylinder Oil' , 'Draft' etc. were flattened to extract relevant metrics.

Example:

- From Consumptions: Extracted metrics included fuel type (e.g., HFO, MGO) and the amount consumed per component (e.g., MAINENGINE, AUXENGINE).
- From Fuel Robs: Extracted the remaining fuel on board for each fuel type.

Resulting Dataset:

- Key Columns:
  - Engine Parameters: RPM, power at shaft, running hours, Consumption, Total Cylinder Oil Consumption, Total Cylinder Oil Specific Consumption
  - Environmental Conditions: Air temperature, wind direction, sea state, etc.
  - Voyage Details: Distance traveled (GPS/log), cargo weight, port codes.
  - Fuel Metrics: Daily consumption, remaining on-board fuel.

2. Cleaning & Standardization

Handling Missing/Null Values:

- Identification: Columns with missing values were identified (e.g., air pressure, sea state).
- Decisions Based on Data Context:
  - Numerical Fields:

        - Imputed with mean or median where values were missing sporadically.
        - Used domain-specific default values (e.g., sea state = 0 if null).
    - Categorical Fields:
        - Applied consistent encoding (e.g., one-hot encoding for Fuel Type ID Code).
        - Used "Unknown" as a placeholder for unidentifiable categories.

Data Standardization:

- Normalization: Continuous features were normalized to bring them to comparable scales.
- Timestamp Conversion: Timestamp fields were converted into consistent formats (e.g., UTC).

Data Assumptions:

- Sequential Nature: Data entries are assumed to be sequential and accurate for time-series modelling.
- Critical Fields: Missing data in critical fields (e.g., fuel consumption) was flagged for removal if imputation (using interpolate Function) was not feasible.

3. Exploratory Data Analysis (EDA)

Fuel Consumption:

- Average daily main engine fuel consumption: X tons (placeholder).
- Distribution shows a right skew due to high consumption outliers.

Speed:

- Average speed: Y knots (placeholder).
- Correlation observed between speed and fuel consumption.

Feature Correlations:

Distance Travelled vs. Consumption:

✓ Strong positive correlation (R = 0.85).

Cargo Weight vs. Consumption:

✓ Moderate correlation observed (R = 0.6).

- Environmental Factors:

Sea state, wind strength, Air Temperature, weather, Air Pressure, shows moderate correlations with fuel consumption.

Anomalies Identified:

- Instances where fuel consumption significantly deviates from expected values, indicating possible inefficiencies.

Phase 2

Feature Engineering Steps and Rationale.

➢ *Target KPIs*

Total Fuel Consumption:

- Definition: The total amount of fuel consumed by the main engine in a day.
- Reason: Serves as a key indicator for overall energy usage during voyages, making it critical for operational cost analysis and energy efficiency planning. Predicting this KPI helps in better fuel budgeting and optimizing vessel performance.
- Calculation:

  Total daily Fuel Consumption = Consumption+ Total Cylinder Oil Consumption+ Total Cylinder Oil Specific Consumption

Fuel Consumed per Nautical Mile:

- Definition: The amount of fuel consumed per nautical mile travelled.
- Reason: Valuable metric for assessing the efficiency of fuel usage in relation to distance travelled. This KPI enables insights into fuel economy under varying operational conditions such as cargo weight, weather, and engine performance.
- Calculation: Total Daily Main Engine Fuel Consumption divided by Sailed Distance.

  Fuel Per Nautical Mile = Total Fuel Consumption / Sailed Distance

➢ ***Distance-Based Features***

Sailed Distance:

- Definition: The total distance sailed since the last report.
- Reason: Direct correlation with fuel consumption.

3. Vessel Operation Features

Average Speed:

- Definition: The average speed of the vessel.
- Reason: Speed impacts fuel consumption.
- Calculation: Mean of Average Speed GPS and Average Speed Log.

4. Environmental Features

Wind Speed and Direction:

- Definition: Combined wind speed and direction.
- Reason: Wind conditions affect propulsion efficiency.
- Calculation: Concatenate Wind Direction and Wind Direction Is Variable as strings.

Sea State:

- Definition: The state of the sea, including direction.
- Reason: Sea conditions affect fuel consumption.
- Calculation: Mean of Sea State and Sea State Direction.

Weather Conditions:

- Definition: Combined atmospheric and weather conditions.
- Reason: Atmospheric conditions impact engine performance.
- Calculation: Concatenate Air Pressure, Air Temperature, Water Temperature, Relative Air Humidity, and Weather as strings.

5. Cargo & Ballast Features

Cargo:

- Definition: The total weight of the cargo.
- Reason: Heavier loads increase fuel consumption.
- Calculation: Sum the values in the columns Cargo Metric Tons and Estimated Bunkers Next Port.

6. Derived Features

Total Daily Fuel Consumption:

- Definition: The total fuel consumption by the main engine in a day.
- Reason: Provides a comprehensive metric for main engine fuel usage.
- Calculation: Sum of Consumption, Total Cylinder Oil Consumption, and Total Cylinder Oil Specific Consumption specific to the main engine.

Fuel Per Nautical Mile:

- Definition: Fuel consumption normalized by distance travelled in a day.
- Reason: Evaluates fuel efficiency per mile.
- Calculation: Total daily Fuel Consumption divided by Sailed Distance.

Phase 3

Model Choices and Hyperparameter Tuning Strategy

➢ Linear Regression:

- Rationale: Acts as a baseline model due to its simplicity and interpretability.
- Application: Suitable for understanding the direct relationships between features and the target variable (fuel consumption).
- Limitations: Assumes a linear relationship, which may not capture the complexities of the data.

➢ **Random Forest Regressor:**

- Rationale: Effective for capturing non-linear relationships and handling large datasets with high dimensionality.
- Application: Utilizes ensemble learning by combining multiple decision trees to reduce overfitting and improve generalization.
- Key Hyperparameters:
  - ✓ n_estimators: Number of trees in the forest.
  - ✓ max_depth: Maximum depth of the trees.
  - ✓ min_samples_split: Minimum samples required to split a node.
  - ✓ min_samples_leaf: Minimum samples required at a leaf node.

➢ **Gradient Boosting Regressor:**

- Rationale: Powerful for structured data and capable of capturing complex feature interactions.
- Application: Focuses on minimizing prediction errors iteratively by combining weak learners.

o Key Hyperparameters:

  - ✓ learning_rate: Shrinks the contribution of each tree.

- ✓ n_estimators: Number of boosting rounds.
- ✓ max_depth: Limits the depth of each tree to prevent overfitting.
- ✓ subsample: Fraction of samples used for fitting individual trees.

➢ **Neural Networks (Multi-Layer Perceptron - MLP):**

- Rationale: Suitable for capturing highly non-linear and complex relationships in the data.
- Application: Requires careful feature scaling and sufficient data to prevent overfitting.

o Key Hyperparameters:

- ✓ hidden_layer_sizes: Number and size of hidden layers.
- ✓ activation: Activation function for hidden layers (e.g., ReLU, tanh).
- ✓ solver: Optimization algorithm (e.g., Adam, SGD).
- ✓ learning_rate: Controls the step size in parameter updates.

➢ **Support Vector Regression (SVR):**

- Rationale: Effective in high-dimensional spaces and robust to outliers.
- Application: Uses kernels (e.g., RBF, linear) to model non-linear relationships.
- Key Hyperparameters:
  - ✓ C: Regularization parameter.
  - ✓ kernel: Kernel type (e.g., linear, polynomial, RBF).
  - ✓ epsilon: Defines a margin of tolerance for error.

Hyperparameter Tuning Strategy:

➢ **Grid Search**:

- Description: Exhaustive search over a predefined hyperparameter grid.
- Advantages: Guarantees finding the best combination of parameters within the grid.
- Limitations: Computationally expensive, especially for large datasets or complex models.

➢ **Randomized Search:**

- Description: Samples a fixed number of parameter combinations randomly from a distribution.
- Advantages: Faster than grid search and often identifies near-optimal solutions.

- Limitations: May miss the optimal combination depending on sampling.

- **Bayesian Optimization:**
  - Description: Models the objective function and selects hyperparameters based on past evaluations.
  - Advantages: Efficient and converges faster to optimal values.
  - Limitations: Requires specialized libraries and more setup effort.

- **Cross-Validation for Evaluation:**
  - Description: Uses techniques like K-Fold or Time-Series Split to evaluate model performance.
  - Advantages: Ensures robustness by testing on multiple splits of the data.
  - Key Metrics:
    - ✓ RMSE (Root Mean Squared Error): Measures prediction error magnitude.
    - ✓ MAE (Mean Absolute Error): Measures average absolute differences between predictions and actual values.
    - ✓ MAPE (Mean Absolute Percentage Error): Measures relative prediction accuracy as a percentage.

- **Early Stopping:**
  - Description: Monitors validation performance during training and halts when performance stops improving.
  - Advantages: Prevents overfitting and reduces training time for models like XGBoost and neural networks.

Key Performance Metrics and Interpretation

- **Root Mean Squared Error (RMSE):**
  - Definition: RMSE is the square root of the average squared differences between predicted and actual values. It penalizes large errors more than small ones.
  - Interpretation:
  - A lower RMSE indicates that the model predictions are closer to the actual values.

- It is sensitive to outliers, so if RMSE is high, check for potential outliers in the data or investigate overfitting/underfitting.

- **Mean Absolute Error (MAE):**
  - Definition: MAE is the average of the absolute differences between predicted and actual values. It treats all errors equally, without squaring them.
  - Interpretation:
    - ✓ MAE is more intuitive because it provides the average magnitude of errors in the same units as the target variable.
    - ✓ Lower MAE indicates better performance.

- **Mean Absolute Percentage Error (MAPE):**
  - Definition: MAPE is the average percentage error between predicted and actual values, making it unit-independent.
  - Interpretation:
    - MAPE helps understand the error as a percentage of the actual value, which is especially useful when comparing across datasets with different scales.
    - Lower MAPE (e.g., <10%) indicates good performance.
    - High MAPE might indicate issues like heteroscedasticity or large relative errors for small values.

- **R-squared (R^2):**
  - Definition: R^2 measures the proportion of variance in the dependent variable that is explained by the model.
  - Interpretation:
    - R^2 ranges from 0 to 1. A higher R^2 indicates better model fit.

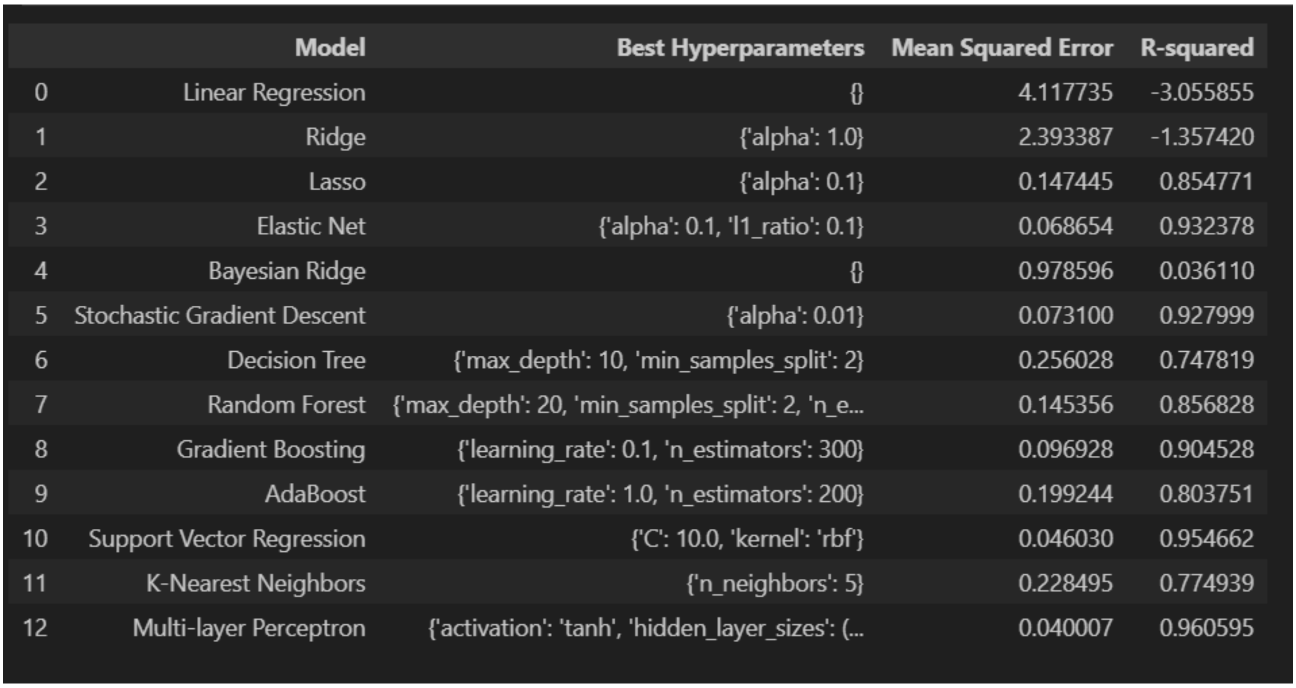
Feature Importance and Interpretations:

- **SHAP (SHapley Additive ExPlanations):**
  - SHAP values show how each feature contributes to predictions for individual data points.
  - Interpretation:
    - ✓ Positive SHAP values indicate the feature increases predicted fuel consumption.
    - ✓ Negative SHAP values mean the feature decreases predicted fuel consumption.

- **Permutation Importance:**
    - Measures the drop in model performance when a feature's values are shuffled.
    - Interpretation:
        - ✓ Features causing significant performance drops are more critical.
        - ✓ Example: If "engine RPM" permutation decreases RMSE the most, it is the most impactful feature.
- **Model-Specific Methods (e.g., Gini Importance in Random Forest):**
    - For tree-based models, importance is derived from how often a feature is used to split data and the resulting reduction in error.
    - Interpretation:
        - ✓ Features with higher importance scores are more influential.
        - ✓ Example: If "cargo weight" has the highest score, it strongly affects fuel predictions.

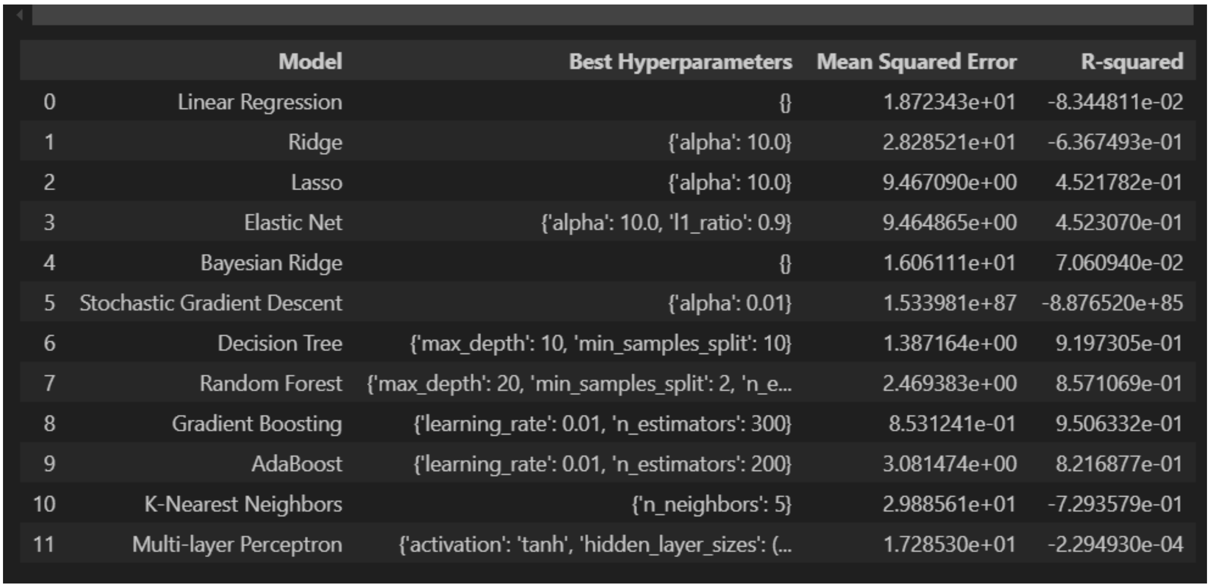
Results and Analysis

KPI-1: Total daily Fuel consumption



	Model	Best Hyperparameters	Mean Squared Error	R-squared
0	Linear Regression	{}	4.117735	-3.055855
1	Ridge	{'alpha': 1.0}	2.393387	-1.357420
2	Lasso	{'alpha': 0.1}	0.147445	0.854771
3	Elastic Net	{'alpha': 0.1, 'l1_ratio': 0.1}	0.068654	0.932378
4	Bayesian Ridge	{}	0.978596	0.036110
5	Stochastic Gradient Descent	{'alpha': 0.01}	0.073100	0.927999
6	Decision Tree	{'max_depth': 10, 'min_samples_split': 2}	0.256028	0.747819
7	Random Forest	{'max_depth': 20, 'min_samples_split': 2, 'n_e...	0.145356	0.856828
8	Gradient Boosting	{'learning_rate': 0.1, 'n_estimators': 300}	0.096928	0.904528
9	AdaBoost	{'learning_rate': 1.0, 'n_estimators': 200}	0.199244	0.803751
10	Support Vector Regression	{'C': 10.0, 'kernel': 'rbf'}	0.046030	0.954662
11	K-Nearest Neighbors	{'n_neighbors': 5}	0.228495	0.774939
12	Multi-layer Perceptron	{'activation': 'tanh', 'hidden_layer_sizes': (...	0.040007	0.960595

KPI-2: Fuel consumption per nautical mile



	Model	Best Hyperparameters	Mean Squared Error	R-squared
0	Linear Regression	{}	1.872343e+01	-8.344811e-02
1	Ridge	{'alpha': 10.0}	2.828521e+01	-6.367493e-01
2	Lasso	{'alpha': 10.0}	9.467090e+00	4.521782e-01
3	Elastic Net	{'alpha': 10.0, 'l1_ratio': 0.9}	9.464865e+00	4.523070e-01
4	Bayesian Ridge	{}	1.606111e+01	7.060940e-02
5	Stochastic Gradient Descent	{'alpha': 0.01}	1.533981e+87	-8.876520e+85
6	Decision Tree	{'max_depth': 10, 'min_samples_split': 10}	1.387164e+00	9.197305e-01
7	Random Forest	{'max_depth': 20, 'min_samples_split': 2, 'n_e...	2.469383e+00	8.571069e-01
8	Gradient Boosting	{'learning_rate': 0.01, 'n_estimators': 300}	8.531241e-01	9.506332e-01
9	AdaBoost	{'learning_rate': 0.01, 'n_estimators': 200}	3.081474e+00	8.216877e-01
10	K-Nearest Neighbors	{'n_neighbors': 5}	2.988561e+01	-7.293579e-01
11	Multi-layer Perceptron	{'activation': 'tanh', 'hidden_layer_sizes': (...	1.728530e+01	-2.294930e-04