ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume X-4/W4-2024
8th International Conference on Smart Data and Smart Cities (SDSC), 4–7 June 2024, Athens, Greece

Detecting Hull Fouling using Machine Learning Algorithms trained on Ship Propulsion Data to Improve Resource Management and Increase Environmental Benefits

Christos Petridis[1], Michael Vassilakopoulos[1]

[1]Department of Electrical and Computer Engineering,
University of Thessaly, Sekeri & Cheiden Str., 38334 Volos, Greece - {cpetridis, mvasilako}@uth.gr

Keywords: Ship Performance Monitoring, Machine Learning, Hull Fouling, Naval Empirical Rules, Environment-Friendly Solutions, Intelligent Transport Systems

Abstract

This study aims to develop a methodology to assess hull fouling based on ship propulsion data such as speed, draft and weather related data. Hull fouling is an unavoidable phenomenon in ships and results in higher fuel consumption and the maintenance frequency has be the optimal one. Despite the fact that until now this task has primarily relied on empirical rules, it turns out that it can be improved by employing machine learning techniques. Using data from clean-hull ships, we aim to isolate and consider only the weather in this study. Our goal is to replace empirical rules with machine learning, as the vast amount of data we possess can significantly aid us in this endeavor. It ends up to be a regression problem, and therefore, we experiment with several supervised algorithms using k-fold cross validation to finally select models based on ensemble methods or artificial neural networks. We propose the potential use of MLP Regressor, Random Forest Regressor and XGB Regressor since all of them yielded very good results in terms of some performance metrics. The timely detection of hull fouling can provide substantial benefits in terms of resource management and environmental sustainability.

1. Introduction

Shipping companies remain highly important in the contemporary global economy since it is estimated that around 80 percent of all goods are carried by sea (Statista, 2024). With the growth of the world economy over the past decades, the amount of cargo transported by ships has increased as well. Therefore, shipping companies dedicate resources, both time and capital, to attain optimal performance and lower operational costs. Fuel consumption and the general operational performance are very crucial in such companies.

1.1 Hull Fouling

Hull and propeller fouling can significantly impact the overall performance of a ship. The hull is the exterior surface of the ship that is in direct contact with the water. Hull fouling refers to the accumulation of marine (micro) organisms, such as barnacles and algae, on the submerged surface of a ship (Coraddu et al., 2019). This phenomenon produces an additional resistance for the ship, leading to higher fuel consumption and power demand (Song et al., 2020). One of the most famous and crucial strategies to mitigate hull fouling is dry docking.

1.2 Dry-Docking

Dry docking is very important in the maritime industry since it plays a key role in maintaining the safety, reliability, and efficiency of ships. It is about bringing a ship to a specially designed platform (shipyard) for maintenance. During dry docking, ships undergo a series of activities including, but not limited to, hull inspection and maintenance, checking the propeller and rudder, cleaning tanks, and overhauling the engine. Each company sets its own frequency for dry dockicng and it is approximately every 40 to 50 months depending on factors such as the age and the model of the ships (Kr Dev and Saha, 2015). The cost of dry-docking is a significant component of a ship's operational expenses and it is usually represented as a six or seven figure amount (Apostolidis et al., 2012). Thus, the frequency has to be as efficient as possible. Obviously, if shipping companies conduct dry docking earlier than the optimal frequency, emissions due to hull fouling would be reduced to zero, but the costs would be higher. Conversely, if they delay dry docking significantly beyond the optimal frequency, emissions due to hull fouling would increase, but the costs would be lower. Therefore, it is important to determine the ideal timing for dry docking.

1.3 Our work

To date, the detection of hull fouling is based on some empirical rules for the weather. However, nowadays we possess a vast amount of data on different operational features of ships, such as speed, power demand, and weather-related data among others. Using these data properly, along with the help of machine learning, can lead to notable progress in the early detection of hull fouling. Shipping industry significantly contributes to the greenhouse effect by emitting large quantities of carbon dioxide, nitrogen oxides and sulfur oxides (Aakko-Saksa et al., 2023). Ships, which heavily rely on fossil fuels such as heavy oil and diesel, emit these harmful gases into the air, contributing to global warming and negatively affecting air quality (Wan et al., 2018). To address these environmental issues, immediate actions are necessary and one of them is the optimal frequency for dry docking.

In this paper, we propose and implement a methodology for predicting hull fouling based on the the actual power requirements of ships due to weather conditions (MeasuredWeatherPower) rather than relying on empirical rules (EmpiricalWeatherPower). Machine Learning (ML) is incredibly beneficial in this context because it helps us to extrapolate novel rules, akin to empirical ones, based on authentic data. It is noteworthy to mention that the data are already being collected and we propose a methodology on how these data could be used, without the need for expensive equipment and high computational technology.

The rest of this paper is organized as follows. Section 2 delves into the scientific literature review relevant to this study. Section 3 describes very briefly the ships that the data refer to, explains the variables that are subsequently used and discusses pre-processing techniques. Section 4 begins with foundational information in the field of maritime, and it also highlights certain issues with current approaches. Section 5 starts with an introductory subsection to describe our machine learning approach and afterwards details the methodology we employed along with reference in the performance of some machine learning algorithms. There is also a subsection to describe the outcome of these algorithms and how it solves the problem. To summarize, we present overall conclusion in Section 6 and we also mention possible future work in Section 7.

2. Related Work

This section offers a detailed overview of the scientific literature relevant to our study. We provide a summary of data-driven techniques that are important for the needs of this paper, along with some basic information.

Senteris, Kanellopoulou and Zaraphonitis (Senteris et al., 2019) explored a data-driven approach using artificial neural networks (ANNs) and they evaluated 10 different training algorithms in the task of assessing a ship's performance and predict hull fouling based on clean-hull ships. Through their work, they consider only weather resistance and they experiment a lot with ANNs.

Filippopoulos and Stamoulis (Filippopoulos and Stamoulis, 2017) introduced a platform called Internet of Vessels (IoV) where shipping companies are able to collect and process data from on-board sensors almost in real time. Engine performance, gas emissions, trim, draft, temperature and navigation are some of the data that could be collected. They also refer to the communication section between the vessels and the shipping company that owns or monitors the vessels.

Coraddu et al. (Coraddu et al., 2019) suggested two anomaly detection methods by employing unsupervised machine learning algorithms such as Support Vector Machines and k-nearest neighbors to eliminate the need for labeled data related to hull conditions. Using data from a research vessel (i.e., The Princess Royal), their study shows the effectiveness of their approach in real operational scenarios. The proposed models help predict hull conditions in real-time, improving resources management and making maintenance decisions such as dry docking in the best frequency. Overall, their novelty detection method demonstrates satisfactory accuracy, a noteworthy outcome given that only 10 samples are required and can be easily labeled by an operator.

Gkerekos et al. (Gkerekos et al., 2019) conducted a comparative analysis of data-driven algorithms to assess their effectiveness in predicting ship main engine Fuel Oil Consumption (FOC). Their study investigated the impact of various data acquisition strategies, including traditional noon-reports and Automated Data Logging & Monitoring (ADLM) systems. They observed that Extra Trees Regressors (ETRs) and Random Forest Regressors (RFRs) outperformed other models, with ADLM systems enhancing accuracy by 5–7% and reducing data collection periods by up to 90%. Their work underlines the importance of hyperparameter optimization, diverse modeling architectures, and highlighted the comparable results of simpler models such as Linear Regression (LR).

Pedersen and Larsen (Pedersen and Larsen, 2009) adopted an approach in the ship performance monitoring domain, departing from traditional reliance on empirical or hydromechanical methods. Instead, they implemented a single hidden layer neural network (ANN) for predicting mean propulsion power, achieving remarkable accuracy. This neural network, specifically trained for the tanker ”Torm Marie” across diverse conditions, outperformed conventional methods. The success of this approach is attributed to the incorporation of key input variables such as ship's speed, relative wind speed and direction, air temperature, and sea water temperature. The departure from conventional methodologies and the successful application of neural networks not only pave the way for more precise and adaptable ship performance evaluations but also hold potential implications for assessing hull and propeller performance in real-world scenarios.

3. Ships Description and Data

For the present study, we had the privilege to have some data from a Greek shipping company, which should remain anonymized for confidentiality purposes. The data refer to some clean-hull VLCC (Very Large Crude Carrier) ships belonging to the same class (Class A), which we will denote them as VLCC 1, VLCC 2, VLCC 3, ..., VLCC 9. Table 1 refers to some characteristics of the ships we used for our study.

Deadweight tonnage (DWT)	317,048.3
Year of built	2018
Country of Built	Korea
Vessel Type	Very Large Crude Carriers

Table 1. Characteristics of the 9 ships.

3.1 Available Data

The available features in our data are shown below and they are collected mainly from on-board sensors.

ShaftPower : It refers to the actual power measured by the torque meter. It is measured in kilowatts (kW).

LogCorrectedSpeed : It refers to the ship's speed using a device known as a log meter, which computes the ship's velocity through the water calibrated based on the possible error of the sensor. It is measured in knots.

Draft : It refers to the vertical distance between the waterline and the lowest point of the hull. It represents how much of the ship is submerged in the water. It is measured in meters (m).

WaveRelativeDirection : It represents the direction from which the waves approach relative to the ship's heading. It is measured in degrees.

WaveHeight : It refers to the vertical measurement of the waves that the ship faces. It is measured in meters.

https://doi.org/10.5194/isprs-annals-X-4-W4-2024-185-2024 | © Author(s) 2024. CC BY 4.0 License.

WavePower : It expresses the power required by the ship due to waves, as per empirical rules. It is measured in kilowatts (kW).

WindSpeed : It refers to the speed of the wind. It is measured in knots.

WindPower : It expresses the power required by the ship due to wind, as per empirical rules. It is measured in kilowatts (kW).

PowerModel : It refers to the power that the ship should generate when calm sea and very mild weather are applicable. It comes from the shipyard and it is calculated from the draft and the speed of the ship. It is measured in kilowatts (kW).

CorrectedPower : It is a percentage indicating the additional power generated by the ship, based on the empirical rules for weather. It often falls within the range of 80% to 130%. To make it more clear it can be expressed as:

$$\mathit{CorrectedPower} = \frac{\mathit{ShaftPower}}{\mathit{PowerModel} + \mathit{EmpiricalWeatherPower}} \cdot 100 \quad (1)$$

where *EmpiricalWeatherPower* is the power that the ship would require based on the empirical rules due to weather.

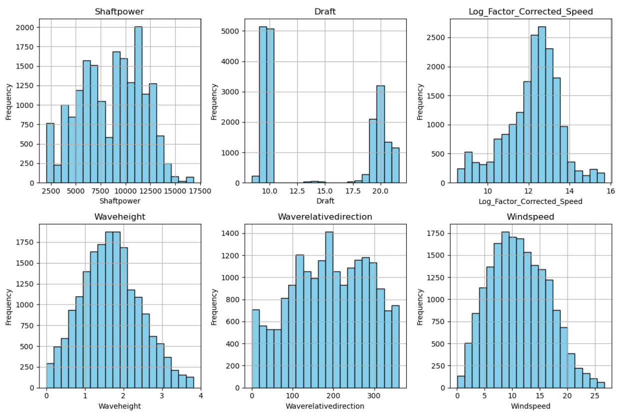

Figure 1. Histograms for each feature of our dataset.

3.2 Feature Engineering

Based on the data above, we create a new variable that will serve as the target variable in the subsequent machine learning algorithms. In particular, the new feature is as follows:

$$\mathit{MeasuredWeatherPower} = \mathit{ShaftPower} - \mathit{PowerModel} \quad (2)$$

Since we are working on clean-hull ships, *MeasuredWeatherPower* refers to the actual power due to weather conditions which is the subtraction of the power that the ship should demand in perfect weather conditions from the overall observed power demand.

Moreover, we can calculate the power demand predicted by empirical rules due to weather as:

$$\mathit{EmpiricalWeatherPower} = \mathit{WindPower} + \mathit{WavePower} \quad (3)$$

3.3 Handling Outliers

In the process of data preprocessing, addressing outliers is a common and essential practice. We cannot take for granted that the data are perfect and can be used immediately. We employ two methods in order to make our dataset as reliable as possible. The first method, renowned for its effectiveness, is the well-known Interquartile Range (IQR). The second one involves some domain knowledge setting suitable ranges for each feature. In particular, we utilize a feature from the dataset called ”*Visible*” since this column tells us whether a row has successfully passed certain checks and can be utilized in our task.

3.3.1 Interquartile Range method: It involves looking at the spread of data within a dataset. It is calculated as the difference between the third quartile (Q3) and the first quartile (Q1) in a dataset. Any data points that fall outside a certain range, calculated using the IQR, are considered outliers. To make our data more reliable, we removed such values. We have only numeric columns and thus, we calculate the Interquartile Range (IQR) for each numeric column. Then we determine lower and upper bounds based on Equations 4 and 5 respectively.

$$\mathit{lower\ bound} = Q1 - k \cdot IQR, \quad k = 1.5 \quad (4)$$

$$\mathit{upper\ bound} = Q3 + k \cdot IQR, \quad k = 1.5 \quad (5)$$

where $Q1$ = first quartile
$Q3$ = third quartile
k = a constant determining the range of values considered as outliers
$IQR = Q3 - Q1$

3.3.2 Domain Knowledge method: Applying domain knowledge in such problems is crucial and can help a lot in achieving better results. Comments and annotations from operators and naval engineers can help us retain reliable data excluding outlier values. Operators from shipping companies can provide valuable information on how ships behave in various conditions, such as different weather conditions, speeds, and loading/unloading conditions. Thankfully, domain experts have set some filters when retrieving data from company's database and therefore there is a column named *”Visible”* which indicates whether each row can be used in our analysis or not. If a row is not reliable, the reason is listed under the *”Visible”* column. If a row is reliable, TRUE is listed under the *”Visible”* column. It is noteworthy to mention that the vast majority of the rows do have TRUE under the *”Visible”* column ($\sim$ 93% of the rows have TRUE). By doing so, we have a reliable dataset that we can rely on in order to extrapolate their behavior to other data as well.

4. Methodology

4.1 Domain Knowledge

To make the main idea more comprehensible, it would be helpful to analyze some specific cases. Generally speaking, the power that a ship requires when it sails can be expressed with the following equation:

$$\mathit{Power} = \mathit{PowerModel} + \mathit{WeatherPower} + \mathit{HullFoulPower} \quad (6)$$

where *WeatherPower* is the additional power due to weather conditions and *HullFoulPower* is the additional power due to hull fouling.

Based on the above relationship (Equation 6), it is evident that when a ship has a clean hull the required power is:

$$Power = PowerModel + WeatherPower \tag{7}$$

since there is no resistance due to hull fouling. discrepancy

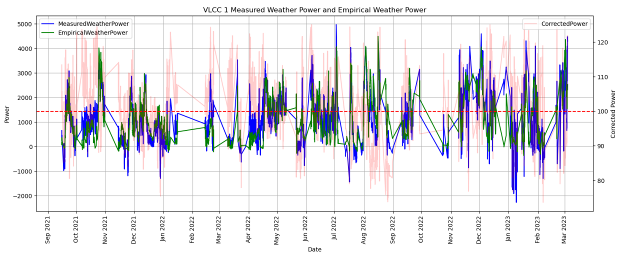

Figure 2. The influence of the discrepancy between the actual power demand due to weather and the power demand predicted by empirical rules in CorrectedPower for VLCC 1 (clean-hull).

between the empirical and the actual WeatherPower

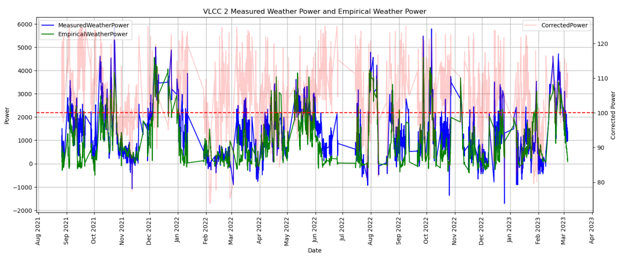

Figure 3. The influence of the discrepancy between the actual power demand due to weather and the power demand predicted by empirical rules in CorrectedPower for VLCC 2 (clean-hull).

Unitl now, the *WeatherPower* is calculated based on some empirical naval rules. These rules are functions that are based on some weather data (i.e., *WaveHeight*, *WindSpeed* etc.), and they compute the amount of power (kW) needed due to weather. However, as shown in Figure 2 and Figure 3, there is a discrepancy between the actual power demand due to weather and the power demand predicted by empirical rules due to weather. Machine learning algorithms, and especially supervised learning algorithms, are able to learn patterns on historical data and solve this problem. We aim to replace the empirical rules employing such algorithms.

It is evident that based on Equation 2, when a ship has a clean-hull along with calm sea and good weather conditions then the *ShaftPower* equals the *PowerModel* and *MeasuredWeatherPower* equals zero. This occurs because, in such cases, ships encounter no resistance due to weather conditions. Obviously, having a perfect way to compute the additional power due to weather for clean-hull ships would result in the following relationship:

$$\frac{ShaftPower}{PowerModel + accurateWeatherPower} \approx 1 \tag{8}$$

where *accurateWeatherPower* is the exact amount of power (kW) required due to weather conditions.

4.2 Issues with Current Approaches

In case that the power demand as per empirical rules (*EmpiricalWeatherPower*), as represented in Equation 1, closely aligns with the actual power demand due to weather, only in the context of clean-hull ships, the *CorrectedPower* (Equation 1) should be very close to 100%. It has to be around 100% because when the *EmpiricalWeatherPower* is accurate enough then the *ShaftPower* is almost equal to the sum *PowerModel* + *EmpiricalWeatherPower*, leading to *CorrectedPower* ≈ 100%. Following the discrepancy between the actual and the predicted by empirical rules power demand as demonstrated in previous figures (Figure 2 and Figure 3); the issue becomes more obvious in the subsequent figures below.

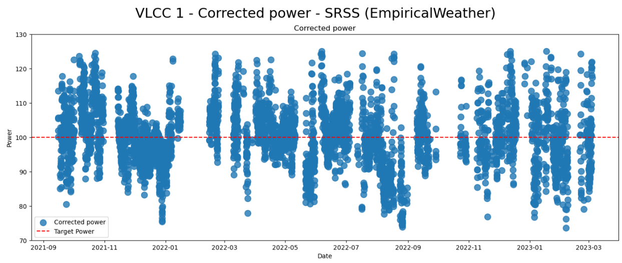

Figure 4. CorrectedPower for VLCC 1 (clean-hull).

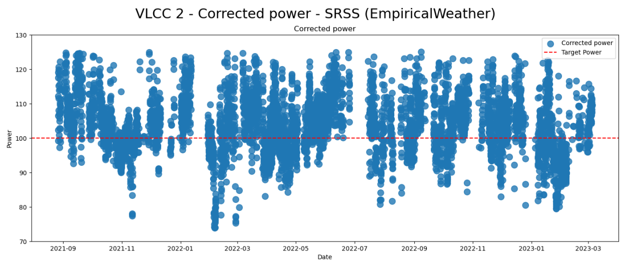

Figure 5. CorrectedPower for VLCC 2 (clean-hull).

Figure 4 and 5 clearly show that the empirical rules are not aligning with expectations. Specifically, the variable *CorrectedPower*, anticipated to be around 100 in ships with clean hulls but it diverges significantly from this expected value. This discrepancy points to potential inaccuracies of the empirical rules.

5. Machine Learning Experimentation

5.1 Machine Learning Introduction

The main challenge is the high variance around 100 in Figures 4 and 5. Data points in this kind of plots have to be as close to 100% as possible. Therefore, we propose a new variable instead of *CorrectedPower* based on the same formula as the original *CorrectedPower* (Equation 1) with a small modification. To achieve this, we introduce a new method using machine learning to compute the additional power due to weather conditions and we replace *EmpiricalWeatherPower* with our predicted number. Therefore, Equation 1 is converted to:

$$ML\ CorrectedPower = \frac{ShaftPower}{PowerModel + MLWeatherPower} \cdot 100 \tag{9}$$

where *MLWeatherPower* is the predicted power due to weather conditions. Similarly, Equation 9 refers to clean-hull data.

Specifically, we calculate the *MLWeatherPower* based on the actual data from on-board sensors. Since we possess weather-related data, such as wave and wind information, and can very easily compute the additionally required power due to weather (Equation 2), it ends up to be a regression problem. Supervised learning algorithms are very effective for tasks where the goal is to predict a continuous outcome variable, in our case this variable is the *MLWeatherPower*, also known as a dependent or target variable. In regression problems, such algorithms learn a relationship between input features and a continuous output, allowing it to make predictions for unseen data. (Choudhary and Gianey, 2017).

5.2 Regression Problem Dataset

Variable	ML scope
ShaftPower	Feature
LogCorrectedSpeed	Feature
Draft	Feature
WindSpeed	Feature
WaveRelativeDirection	Feature
WaveHeight	Feature
MeasuredWeatherPower	Target

Table 2. Machine learning dataset.

Table 2 presents the dataset used, including both the features and the target variable, to train and validate some machine learning algorithms. We do include ShaftPower in our dataset as a feature because the target variable is measured in kiloWatts and we need to perform nondimensionalization of this. Our problem is a multiple regression one since we attempt to explain a target (dependent) variable using more than one feature (independent) variable.

The target variable in Table 2 (i.e., *MeasuredWeatherPower*) comes from Equation 2 because we are based on the actual data and not on empirical rules (Equation 3). In this manner, we take into consideration only the weather dynamics since the experiment refers to clean-hull ships. Table 3 demonstrates descriptive statistics for our dataset.

5.3 Machine Learning Methodology

We experiment with various machine learning algorithms and consequently select the best one based on some performance metrics. In order to train the machine learning models, we merged the data from clean-hull ships and tried to find an accurate relationship between the features (weather data) and the power demand. It is noteworthy to mention that we were able to merge all available data because the data refer to sister ships. A sister ship is basically a ship that is almost exactly the same as another one. They look identical, have similar sizes, and share the same design features, including the layout of their hull and superstructure

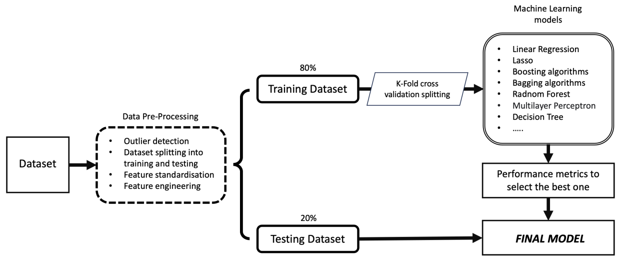

Figure 6. Suggested methodology for our task. Dataset explained in Table 2.

In Figure 6, we present an overview of how we apply machine learning. As mentioned earlier, we aim to identify the most accurate relationship between the features and the target variable, relying on actual (real-world) data.

5.4 Validation and Performance metrics

Having the data ready for machine learning training, we employed K-fold cross-validation (k=10) to validate each algorithm, based on Mean Squared Error and Coefficient of Determination.

5.4.1 K-fold Cross Validation (KCV): This is a technique used to assess the performance of a machine learning model by dividing the dataset into k subsets or ”folds.” The model is trained and evaluated k times, each time using a different fold as the testing set and the remaining k-1 folds for training. This process helps ensure a more robust evaluation of the model's performance, as it considers multiple combinations of training and testing data. In our case, the final performance metric is an average of the metrics computed in each iteration, providing a more reliable estimate of the model's generalization ability (Anguita et al., 2012).

5.4.2 Mean Squared Error (MSE): This a common metric used in machine learning to measure the average squared difference between the actual and predicted values and it is particularly useful for regression problems. The formula for Mean Squared Error is given by:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{10}$$

where n = number of data points
y_i = actual value for the i-th input data
$\hat{y}_i$ = predicted value for the i-th input data.

Merged Dataset 5 sister ships

Descriptive Statistics	Shaftpower (kW)	Draft (m)	LogCorrSpeed (kn)	WaveHeight (m)	WaveRelativeDirection (degrees)	WindSpeed (kn)	MeasuredWeatherPower (kW)
Count	18774	18774	18774	18774	18774	18774	18774
Mean	8584.7622	14.2726	12.1939	1.6721	190.3164	11.2554	1114.5076
Standard Deviation	3144.8256	5.1419	1.3318	0.7638	94.3524	5.1930	1004.3292
Minimum	2042.3228	8.4136	8.5249	0.0000	0.0064	0.0583	-2281.8985
25th Percentile (Q1)	6077.8076	9.7013	11.4863	1.1300	116.9662	7.3089	323.6064
Median (50th Percentile, Q2)	9000.5775	9.9318	12.3965	1.6500	191.0864	10.9050	969.1811
75th Percentile (Q3)	11139.5839	19.9023	13.0471	2.1600	269.3881	15.0259	1751.9017
Maximum	16822.0893	21.6322	15.6923	3.8100	359.9936	26.6501	5907.4973

Table 3. Descriptive statistics for our dataset

https://doi.org/10.5194/isprs-annals-X-4-W4-2024-185-2024 |

5.4.3 Coefficient of Determination (R^2): This is another commonly used metric in regression analysis and problems. It represents the proportion of the variance in the dependent variable that is predictable from the independent variables. The formula for R^2 is as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{11}$$

where n = number of data points
y_i = actual value for the i-th input data
$\hat{y}_i$ = predicted value for the i-th input data.
$\bar{y}$ = the mean of the actual values.

5.5 Machine Learning Models.

Model	Description
Linear Regression	Ordinary least squares linear regression
Lasso	Linear regression with L1 regularization
ElasticNet	Linear regression with a combination of L1 and L2 regularization
KNN Regressor	K-nearest neighbors regression
Decision Tree Regressor	Decision tree-based regression
Random Forest Regressor	Ensemble of decision trees for regression
Gradient Boosting Regressor	Boosted ensemble of decision trees
MLP Regressor	Multi-layer perceptron neural network for regression
XGB Regressor	Extreme Gradient Boosting regression
AdaBoost Regressor	Boosted ensemble using adaptive boosting
Support Vector Regressor (SVR)	Support vector machine for regression

Table 4. Regression Models and Descriptions

We apply K-Fold cross validation (KCV) on 11 machine learning algorithms. Table 4 provides a list of the regression models we employed along with their descriptions. Our models include ordinary least squares linear regression, L1 regularization, a combination of L1 and L2 regularization, K-nearest neighbors regression, decision tree-based regression, and ensemble methods such as random forests and gradient boosting. Additionally, we employ neural network-based approaches like multi-layer perceptron (MLP), as well as models like Extreme Gradient Boosting (XGBoost) regression, AdaBoost, and Support Vector Machine for regression (SVR). It is very important to understand that these metrics do not reflect the whole performance of the models since we want to extrapolate the knowledge from clean-hull ships and evaluate the fouling on other ships (unseen data) which are probably fouled. By saying that, we want to mention that the performance itself will be evaluated later on unseen data. However, evaluating the models with training/testing datasets based on some performance metrics is a good way to select one among all the others.

5.6 Results

The subsequent results refer to our merged dataset, which includes five sister ships. Figure 7 and Table 5 demonstrate how different regression methods perform. Overall, the selection of an appropriate regression model should consider a balance between low MSE and high R^2, while also accounting for the dataset's complexity and the model's generalization capabilities. Therefore, we firmly believe that the best models to continue with are **MLPRegressor**, **XGBRegressor** and **RandomForestRegressor**. Despite the fact that MLPRegressor is an artificial neural network we did not perform any optimization on its hyperparameters and we used the default ones (Python Documentation, n.d.). For simplicity, we will continue with **MLPRegressor** because it is the best one it terms of performance metrics (Fig. 7) but there is no significant difference with the other two well-performed algorithms and the results are the same whichever we selected for the next step of evaluation.

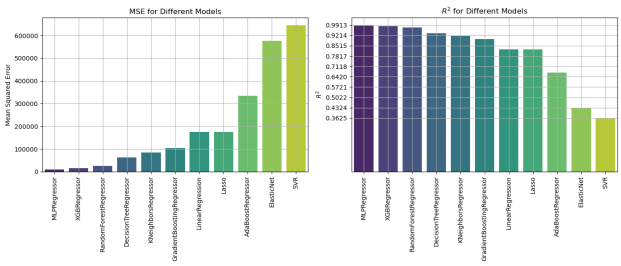

Figure 7. MSE and R^2 to evaluate their performance in testing (20%) dataset.

Model	MSE	R-squared
Linear Regression	174835.86	0.827
Lasso	174887.27	0.827
ElasticNet	577025.15	0.431
KNeighbors Regressor	83697.47	0.917
Decision Tree Regressor	63607.87	0.937
Random Forest Regressor	**25530.58**	**0.975**
Gradient Boosting Regressor	103247.62	0.898
MLP Regressor	**8836.20**	**0.991**
XGB Regressor	**15213.56**	**0.985**
AdaBoost Regressor	334610.47	0.670
SVR	646131.17	0.362

Table 5. Regression Model Performance Metrics

Figure 7 and Table 5 demonstrate how different regression methods perform. Overall, the selection of an appropriate regression model should consider a balance between low MSE and high R^2, while also accounting for the dataset's complexity and the model's generalization capabilities. Therefore, we firmly believe that the best models to continue with are **MLPRegressor**, **XGBRegressor** and **RandomForestRegressor**. Despite the fact that MLPRegressor is an artificial neural network we did not perform any optimization on its hyperparameters and we used the default ones (Python Documentation, n.d.). For simplicity, we will continue with **MLPRegressor** because it is the best one it terms of performance metrics (Fig. 7 and Tab. 5) but there is no significant difference with the other two well-performed algorithms and the results are the same whichever we selected for the next step of evaluation.

5.7 Extrapolate Machine Learning Knowledge

After evaluating various performance metrics, we opt for the MLP Regressor. Having the trained model ready, we can test its effectiveness on unseen data. Some operators from the shipping company has detected the hull fouling on a similar ship (sister ship). Thus, we do know that this dataset refers to a fouled ship and we want to evaluate and observe the performance of our model in such cases which are the cases that we need the most.

Figure 8 shows us the issues with current approaches (empirical-based) on fouled ships. In particular, it is more evident in Figure 8 where the sharp edges are circled in red. Sharp edges should be absent because it is assumed that the variable *CorrectedPower* encapsulates the weather dynamics, and any sudden changes in weather conditions should be integrated within the variable. **Please note, once again, that the CorrectedPower is not the power demand.** Therefore, sharp edges exhibit that current approaches are not very effective in estimating hull fouling.

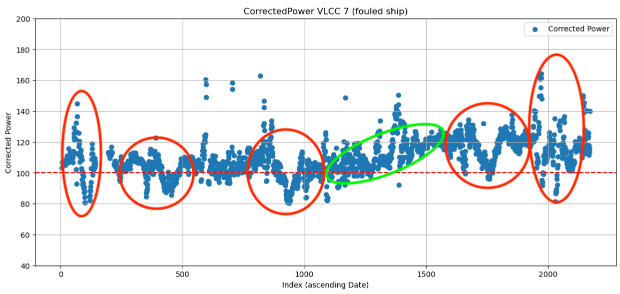

Figure 8. The sharp edges in the empirical-based CorrectedPower for fouled ships.

Having a dataset that refers to a fouled VLCC such as the one presented in Table 2 we will evaluate the effectiveness of our machine learning framework to estimate the fouling based on the new CorrectedPower that we will calculate (ML CorrectedPower as presented in Equation 9). Figure 9 demonstrates the result of our new machine learning-based CorrectedPower (i.e., ML CorrectedPower). It is clear that the most sharp edges are now absent or have been decreased and it is closer to a straight line comparing to empirical based CorrectedPower.

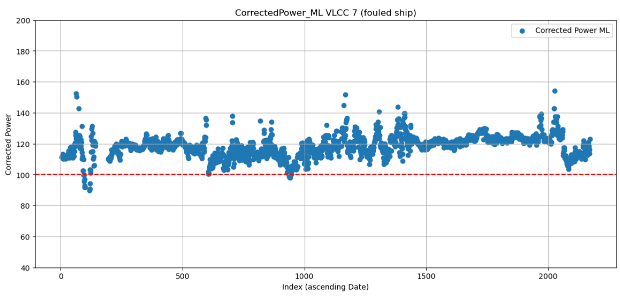

Figure 9. The sharp edges are mainly absent in the machine learning-based CorrectedPower for fouled ships.

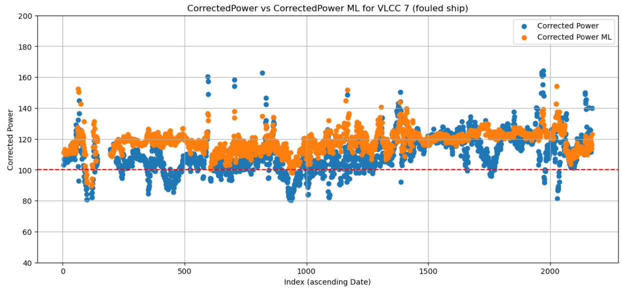

Figure 10. Comparison between the empirical and the machine learning based CorrectedPower for estimating hull fouling.

It is important to underline that the machine learning approach tells us that since the CorrectedPower is around 120, then ship's power demand is 20% greater than it should be (when clear) and further investigation is needed because it is probably fouled (dirty hull). It is evident that the new CorrectedPower is almost never around 100 and this behavior is alarming itself. Therefore, by replacing the old CorrectedPower (empirical-based) with our new one (machine learing based) shipping companies would be able to detect hull fouling timely and better monitoring the (extra) power demand.

6. Conclusions

Our data-driven approach for replacing empirical rules with machine learning algorithms yielded very good results. Performance metrics such as R^2 and MSE exhibited that the best machine learning algorithms to predict the actual power demand because of weather, in accordance with real-world data, are the ensemble methods and ANNs. MLP Regressor, Random Forest Regressor (bagging) and XGB Regressor (boosting) outperform other algorithms showing that combining different models is a powerful approach. Having a better estimation of the required power due to weather we can predict the hull condition and possible fouling based on the overall power demand. It is expected to employ a substantial amount of data when training machine learning models. For training, we used data from 5 clean-hull sister ships but it is recommended to use as many as possible. By incorporating various weather data, the models can better grasp different scenarios, resulting in more accurate predictions for each specific case.

7. Future work

While the present study has provided valuable insights, there are avenues for further exploration and improvement. We firmly believe that handling outliers is very important for such problems and therefore we believe that if there is a way to filter out outliers we could also use simple methods, such as Linear Regression, providing great interpretability. Furthermore, it would be very interesting to examine the use of date information and employ time series analysis and forecasting. Taking into consideration the date can significantly improve the results because hull fouling is strongly correlated with time. Beyond doubt, domain knowledge plays a crucial role in significantly improving the performance of machine learning models through better data pre-processing and feature engineering. Integrating specialized expertise can lead to more accurate and contextually relevant outcomes.

References

Aakko-Saksa, P. T., Lehtoranta, K., Kuittinen, N., Järvinen, A., Jalkanen, J.-P., Johnson, K., Jung, H., Ntziachristos, L., Gagné, S., Takahashi, C., Karjalainen, P., Rönkkö, T., Timonen, H., 2023. Reduction in greenhouse gas and other emissions from ship engines: Current trends and future options. *Progress in Energy and Combustion Science*, 94, 101055. https://www.sciencedirect.com/science/article/pii/S0360128522000624.

Anguita, D., Ghelardoni, L., Ghio, A., Oneto, L., Ridella, S., 2012. The 'k' in k-fold cross validation. *The European Symposium on Artificial Neural Networks*.

Apostolidis, A., Kokarakis, J., Merikas, A., 2012. Modeling the Dry-Docking Cost – The Case of Tankers. *Journal of Ship Production and Design*, 28, 134–143.

Choudhary, R., Gianey, H. K., 2017. Comprehensive review on supervised machine learning algorithms. *2017 International Conference on Machine Learning and Data Science (MLDS)*, 37–43.

Coraddu, A., Lim, S., Oneto, L., Pazouki, K., Norman, R., Murphy, A. J., 2019. A novelty detection approach to diagnosing hull and propeller fouling. *Ocean Engineering*, 176, 65-73. https://www.sciencedirect.com/science/article/pii/S0029801818313581.

Filippopoulos, I., Stamoulis, G., 2017. Collecting and using vessel's live data from on board equipment using "internet of vessels (iov) platform" (may 2017). *2017 South Eastern European Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)*, 1–6.

Gkerekos, C., Lazakis, I., Theotokatos, G., 2019. Machine learning models for predicting ship main engine Fuel Oil Consumption: A comparative study. *Ocean Engineering*, 188, 106282. https://www.sciencedirect.com/science/article/pii/S0029801819304561.

Kr Dev, A., Saha, M., 2015. Modeling and analysis of ship repairing time. *Journal of Ship production and Design*, 31(02), 129–136.

Pedersen, B. P., Larsen, J., 2009. Modeling of ship propulsion performance. *World Maritime Technology Conference WMTC2009*.

Python Documentation, s.-l. d. B. L., n.d. sklearn.neural_network.mlpregressor. https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html. Access Date: 08-04-2024.

Senteris, A., Kanellopoulou, A., Zaraphonitis, G., 2019. A machine learning approach to assess vessel performance based on operational profile. *Sustainable Development and Innovations in Marine Technologies*, 496–502.

Song, S., Demirel, Y. K., Atlar, M., 2020. Penalty of hull and propeller fouling on ship self-propulsion performance. *Applied Ocean Research*, 94, 102006. https://www.sciencedirect.com/science/article/pii/S0141118719305735.

Statista, R. D., 2024. Ocean shipping worldwide - statistics & facts. https://www.statista.com/topics/1728/ocean-shipping/#topicOverview. Access date: 01-03-2024.

Wan, Z., el Makhloufi, A., Chen, Y., Tang, J., 2018. Decarbonizing the international shipping industry: Solutions and policy recommendations. *Marine Pollution Bulletin*, 126, 428-435. https://www.sciencedirect.com/science/article/pii/S0025326X17310214.