UNIVERSITÄT
DUISBURG
ESSEN

Offen im Denken

Master Thesis

on the topic of

Modelling and optimization of ship's fuel consumption using Random Forest Regression (RFR)

Submitted to the Faculty of Engineering
of University Duisburg Essen

by

Hibatul Wafi
3021919

Betreuer:	M. T. Muhammad Fakhruriza Pradana
1. Gutachter:	Prof. Dr.-Ing. B. Noche
2. Gutachter:	Dr.-Ing. Alexander Goudz
Studiengang:	ISE General Mechanical Engineering
Studiensemester:	Summer semester 2023
Datum:	04.05.2023

Abstract

Efforts to model energy-efficient operation of shipping operations using machine-learning methods have emerged due to volatile bunker fuel prices and stringent environmental regulations. It is widely regarded that ship speed is one of the most influential factors impacting ships' fuel oil consumption and as such, accurate modelling of ship speed is paramount to ensure the accuracy of subsequent FOC prediction.

This study proposes an intuitive data-driven modelling approach, integrating Automatic Identification System and weather data for modelling of ship states and environmental conditions' impact on FOC. Grey Box Modelling approach divides the speed and FOC prediction into stages, the first stage involves the prediction of speed over ground using Random Forest Regressor. Consequently, the FOC prediction based on predicted speed employs the empirical formula by Holtrop-Mennen, maintaining adherence with established vessel knowledge.

In the presented case study, optimised SOG prediction achieves 3.94% mean absolute percentage error (MAPE) and 93.41% R^2 score. Subsequent FOC prediction from estimated speed yields 86.57% R^2 and 12.06% MAPE. The results affirm the proposed approach's viability in predicting energy-efficient ship operations.

Keywords: Energy efficient operation, Random Forest Regression, Ship speed prediction, Fuel consumption prediction, Grey Box Model, AIS.

Acknowledgments

To my family, who makes everything worthwhile

Contents

List of Tables

List of Figures

Nomenclature

Symbols with Latin letters

Symbol	Denomination	SI Unit
A_E/A_O	Propeller expanded area ratio	–
A_{BT}	Transverse area of bulbous bow	m^2
A_L	Lateral plane area	m^2
A_T	Immersed transom area	m^2
A_V	Area of ship and cargo above waterline	m^2
A_W	Waterline area	m^2
B	Breadth	m
C_B	Block coefficient	–
$C_{D_{TH}}$	Drag coefficient of thruster tunnel	–
C_F	Frictional Coefficient	–
C_M	Midship coefficient	–
C_P	Prismatic coefficient	–
C_{WP}	Waterplane area coefficient	–
CD_l	Drag coefficient for beam wind	–
CD_t	Drag coefficient for headwind	–
D	Propeller diameter	m
d_{TH}	Diameter of bow thruster tunnel	m
Fr	Froude Number	–
g	Gravity constant	kg/ms^2
$\overline{H}$	Mean wave height	m
$H_{1/3}$	Significant wave height	m
H_{10}	Highest ten percent of wave	m
H_{max}	Maximum wave height	m
H_{swell}	Swell wave height	m
$H_{windwave}$	Wind wave height	m

Symbol	Denomination	SI Unit
h_B	Height of centre A_{BT} above basis	m
i_E	Half angle of waterline entrance	°
J	Cost function	–
k	Form factor	–
k_{2_i}	Appendage form factor	–
k_n	Feature n of ML model	–
ℓ_{CB}	Longitudinal Centre of buoyancy	–
L_{BWL}	Length of bow in the waterline	m
L_{WL}	Waterline length	m
L_{PP}	Length between perpendicular	m
L_R	Length of run	m
n	Data point(s)	–
P_b	Brake power	kW
P_e	Effective power	kW
R_A	Correlation resistance	kN
R_{AA}	Additional resistance due to wind	kN
R_{AW}	Additional resistance due to wave	kN
R_{APP}	Appendage Resistance	kN
R_B	Resistance of bulbous bow	kN
R_F	Frictional Resistance	kN
R_{CALM}	Calm water resistance	kN
R_{TOTAL}	Total Resistance	kN
R_{TR}	Transom Resistance	kN
R_W	Wave resistance	kN
R^2	Coefficient of Determination	–
Re	Reynolds number	–
S_i	Partition space i	–
S	Wetted surface of bare hull	m^2
S_{App}	Wetted surface of appendages	m^2
t	Thrust deduction fraction	–
t_k	Tree threshold k	–
T	Draught	m
T_A	Moulded draught at aft perpendicular	m
T_F	Moulded draught at forward perpendicular	m

Symbol	Denomination	SI Unit
T_p	Spectral peak period	s
T_f	Spectral peak period of fully developed sea	s
u	Apparent wind velocity	m/s
u_{TW}	True wind velocity w.r.t to bow	m/s
u_W	Wind velocity	m/s
V	Displacement volume	m^3
v	Speed	m/s
v_C	Current Speed	m/s
v_G	Speed Over Ground	m/s
v_S	Speed Through Water	m/s
w	Wake fraction	–
y	Response variable	–
$\hat{y}$	Mean of response variable	–

Symbols with Greek letter

Symbol	Denomination	SI Unit
α	Heading angle	°
β	True wind angle	°
γ	True north current direction	°
δ	Cross force parameter	–
ε	Apparent wind angle	°
η_s	Shaft efficiency	–
η_h	Hull efficiency	–
η_r	Rotative efficiency	–
η_o	Open water efficiency	–
η_{TOTAL}	Total efficiency	–
ν	Kinematic viscosity	m^2/s
ρ	Density	kg/m^3
τ	Time	$ss/hh/dd$
φ	True north wind direction	°

Abbreviations

Abbreviation	Denomination
AB	AdaBoost
ADLM	Automatic Data Logging System
(T/S)-AIS	(Terrestrial/Sattelite)-Automatic Identification System
ANN	Artificial Neural Network
BBM	Black Box Model
CART	Classification and Regression Tree
CMEMS	Copernicus Marine Environment Monitoring Service
COG	Course Over Ground
DMA	Danish Maritime Authority
DS	Data set
DT(R)	Decision Tree (Regressor)
ECMWF	European Centre for Medium-Range Weather Forecast
EEOI	Energy Efficient Operational Indicator
ET(R)	Extra Tree (Regressor)
EV	Explained Variance
FOC	Fuel Oil Consumption
GB	Gradient Boosting
GBM	Grey Box Model
GHG	Green House Gas
GPS	Global Positioning System
GT	Gross Tonnage
IMO	International Maritime Organization
ITTC	International Towing Tank Conference
IQR	Interquartile Range
KNN	K-Nearest Neighbour
LASSO	Least Absolute Shrinkage and Selection Operator
LB	Light Gradient Boosting Machine
(M)LR	(Multiple) Linear Regression
MAD	Median Absolute Deviation
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error

Abbreviation	Denomination
ML	Machine Learning
MLP	Multi Level Perceptron
MMSI	Maritime Mobile Service Identity
MSE	Mean Square Error
OPEX	Operational Expenses
RF(R)	Random Forest (Regressor)
RMSE	Root Mean Square Error
RPM	Revolution per Minute
SEEMP	Ship Energy Efficiency Management Plan
SFOC	Specific Fuel Oil Consumption
SOG	Speed Over Ground
SOLAS	International Convention for Safety of Lives at Sea
STW	Speed Through Water
SVR	Support Vector (Regressor)
TWA	True Wind Angle
VHF	Very High Frequency
WBM	White Box Model
XG	eXtreme Gradient Boosting

Chapter 1

Introduction

Marine industry stakeholders are actively pursuing research on efficient ship operation. This research direction is motivated by the increasing price of fuel oil and stricter environmental regulations. The fuel aboard a ship is referred to as "bunkers" and accounts for a substantial portion of the vessel's operational expenses (OPEX). It is known that bunker fuel takes up more than 50% of voyage costs and constitutes up to 75% of the ship's total operating cost. It can be inferred that energy-efficient ship operations that could reduce fuel consumption translate to an increase in profitability **(Stopford 2008; Ronen 2011; Bialystocki and Konovessis 2016)**. Furthermore, efficient operation also means the reduction of Greenhouse Gas Emissions (GHG). The most recent report by International Maritime Organisation indicated that GHG emissions from shipping make up 2.51% of global emissions **(IMO 2020)**. This mutual motivation aligns economic benefits with environmental compliance.

With that, maritime industry stakeholder actively searches for methods to ensure energy-efficient operation. Two approaches are considered, namely technical solutions and operational solutions. Technical solutions involve modification to the vessel's structure and power system. But these solutions are expensive, and it requires engineering innovations **(Yan, Wang, and Psaraftis 2021; Li et al. 2022)**. Because of this, stakeholders look for cheaper solutions to achieve energy-efficient operations. The answer for an inexpensive approach lies in the optimisation of operational measures, it carries less cost, and it does not require initial investments. Several recommended solutions for energy-efficient operation can be found in Ship Energy Efficiency Management Plan (SEEMP).

Significant emphasis is given in this study on the optimisation of ship speed as it is widely considered that ship speed has a substantial impact on fuel consumption. Different studies indicated that fuel consumption is correlated through a third-order, non-linear function of the ship speed **(Wang and Meng 2012; Ronen 2011; Psaraftis and Kontovas 2013; Du et al. 2019)**. The significant impact of ship speed on fuel consumption is further supplemented by reports and studies stating that reducing ship speed by about $2-3$ knots could halve the operating cost of shipping companies **(Stopford 2008; Wijnolst, Wergeland, and Levander 2009)**. Hence, energy-efficient operation is commonly achieved through the practice of slow steam-

ing by shipping industry operators.

While inexpensive, optimising operational measures is not an easy and trivial task. Several factors ranging from vessel operational performance to varying weather conditions make it challenging to model the ship speed. Some fuel consumption models, which are based on historical data and ship parameters, lack generalisation capabilities, and it is sensitive towards noisy data. To address this problem, recent research turns towards data-driven approach i.e. machine learning (ML) approach to predict ship speed and fuel consumption. These studies reported success in their modelling, citing good generalisation capability and low prediction errors. Despite these successes, maritime experts find it difficult to accept models based on data-driven approach, as some data-driven models are complex as well as unintuitive and in some cases can violate basic physical knowledge of the vessel. The performance of the data-driven model is also greatly dependent on both data quantity and quality **(Yan, Wang, and Psaraftis 2021; Gkerekos, Lazakis, and Theotokatos 2019)**.

As such, prompted by volatility and ever-increasing bunker fuel price, developing a model that could accurately predict Fuel Oil consumption (FOC) could prove to be useful to maritime industry stakeholders. As stakeholders could make critical economical decisions at the most opportune moment without violating the stringent environmental regulations.

1.1 Thesis objectives

This thesis proposes an intuitive, data-driven modelling approach that considers varying ship states and environmental conditions to predict the fuel consumption of a vessel. To ensure the abundance of data during modelling, this thesis utilise data fused between Automatic Identification System (AIS) and weather data.

To achieve this, Grey Box Model (GBM) approach is selected. ML approach using tree-based regressor is considered to provide a certain degree of intuitiveness to predict speed over ground (SOG) over different journey periods using fused AIS and weather data. Predicted SOG is then converted to actual ship speed i.e. Speed Through Water (STW). STW will be used as the input for the modelling of Fuel Oil Consumption (FOC), which is carried out through Holtrop-Mennen estimation method **(Holtrop and Mennen 1978, 1982; Holtrop 1984)**, a power estimation method based on hydrodynamic laws which consider resistance forces exerted by environmental conditions.

The following Research Questions (RQs) could be raised during the development of the model :

- **RQ1**: What are the steps that should be taken to optimise the predictive performance of the model?
- **RQ2**: Is it feasible to fuse AIS data and meteorological data to accurately predict the ship's SOG and subsequently FOC of the ship?
- **RQ3**: Which approximations and empirical equations are suitable to estimate the resistance forces required to estimate the power required by the ship?

1.2 Thesis Boundaries

The following research boundaries are set throughout this thesis:

- The weather information and AIS data are assumed to be true. Any uncertainties from AIS data and weather data are neglected.
- The focus of this work is a detailed study of the performance and possible optimisation configuration of different tree-based predictors for SOG. As such, an exhaustive comparison study between different types of machine learning models will not be performed.
- In the case study, the approximation for incomplete ship parameters and dimensions is based on a similar type of ship with nearly identical dimensions.

1.3 Thesis Contributions

The GBM approach using the fusion of AIS data and weather data provides the following contributions :

- Economical and independent data source.
- Robust modelling approach that requires minimal data pre-processing and minimal model configuration.
- Comprehensible model that adheres to physical principles and hydrodynamic laws of the vessel.

1.4 Thesis Structure

The thesis is organised with the following structure:

Chapter 1 introduces the problem statement and described the objective and boundaries of the thesis. The novelty of this thesis is declared in this chapter.

Chapter 2 explains the fundamental aspects of the methodologies used to develop the Black Box Model (BBM) and the White Box Model (WBM). Section 2.1 includes literature review of relevant past and present research. The fundamentals of the tree-based model will be discussed in Section 2.2, basic explanation of the parameters used in AIS and weather data will be given in Section 2.3 and Section 2.4. Section 2.5.2 presents the empirical formulas and parameters used to estimate fuel consumption used by the ship based on various literature studies.

Chapter 3 discusses the methodology used to develop tree-based model used for SOG prediction. The discussion comprises analysis of training data, feature selection and reduction and selection of tuning parameters of the model. The methodology to estimate resistance for ship power estimation will be discussed in this chapter as well.

Chapter 4, the GBM model will be evaluated using appropriate performance metrics and their effectiveness will be discussed. The review of the strength and limitations concerning the GBM method will be discussed here.

Chapter 5 The summary of this study and reflections on the research process will be presented here.

Chapter 2

Theoretical Background

2.1 Literature Review

The literature review in Section 2.1 presents past and present research on the utilisation of machine learning methods to achieve energy-efficient operation. The concept of different modelling approaches for ship operation will be discussed in Section 2.1.1. The summary of the data source in the modelling of FOC is given in Section 2.1.2. The review of popular machine learning models used to predict FOC is presented in Section 2.1.3. The performance of tree-based models, which include random forest and extra trees in previous research will be discussed in Section 2.1.4. Brief summary of the literature review is presented in Section 2.1.6.

2.1.1 Modelling Approach for Ship Operation

According to **Haranen et al. (2016)** and **Coraddu et al. (2017)**, the modelling strategies to predict fuel consumption are classified into three categories:

White Box Models (WBM)

Based on *a priori* mechanistic knowledge and physical principles of the vessel's system. This means that the dimensions of the vessel's structure, design parameters, and propulsion plant configuration should be known.

Black Box Models (BBM)

are purely data-driven, and it is developed using data from different sailing journeys and historical observations. Contrary to WBM, this approach does not require detailed information on the vessel. This modelling approach can be further split into two categories. *Statistical Modelling* aims to find explanations for relationships between fuel consumption and different factors that affect it. *Machine Learning (ML) Modelling* focuses on the predictive capabilities of the model that could predict fuel

consumption at different points in time.

Grey Box Models (GBM)

Fuse WBM and BBM into a single model that considers both *a priori* knowledge of the vessel and historical sailing data, This method aims to complement the performance of WBM and BBM.

Each of these strategies possesses its strengths and limitations. WBMs are developed based on physical and hydrodynamics laws, as well as theories of naval architecture. They are transparent and comprehensible, making them the preferred model used by various shipping industries. However, the deterministic nature of WBMs renders them poorly suited for generalisation and adaptability. This limitation arises from the *a priori* knowledge required for vessel dimensions, parameters, and the restricted application scope of principal dimensions and form parameters. Consequently, WBMs' inability to incorporate randomness constrains their flexibility and versatility **Haranen et al. (2016); Yan, Wang, and Psaraftis (2021)**.

BBMs in general have a good fitting ability for training data and good predictive accuracy for unseen data. BBMs developed using ML approach can generalise better compared to BBMs that are based on statistical modelling **(Petersen, Winther, and Jacobsen 2012)**. BBMs are purely data-driven, which means BBMs do not require former knowledge of vessel principle dimensions and form parameters. With increasing amounts of data, better generalisation performance and handling of noisy data should be expected in a BBM. However, for the same reason, the effectiveness of BBM model is highly dependent on data quantity and quality **(Halevy, Norvig, and Pereira 2009)**. Data-driven approach means that BBMs neglect basic vessel physical knowledge and are generally complex, making it challenging to analyse and explain the reasoning behind a ML model. For these reasons, experts in shipping industries are critical of models that do not include basic vessel knowledge and those that violate concepts of the domain knowledge in serious ways **(Yan, Wang, and Psaraftis 2021)**.

Hence, GBMs are introduced to address the limitations of both WBMs and BBMs by combining the mechanistic knowledge of the ship and physical principles of the vessel's system with BBM models, which possess good predictive capability. Despite these advantages, **Yan, Wang, and Psaraftis (2021)** noted that GBM approach is not a common approach, recent research to predict fuel consumption are mainly dominated by BBM approach, specifically BBM based on ML approach.

2.1.2 Review of data source used for FOC model

The modelling of FOC using GBM requires both components of WBM and BBM. For black-box modelling using ML techniques, it is crucial to have sufficient high-quality data for accurate training **(Halevy, Norvig, and Pereira 2009)**. **Yan, Wang, and Psaraftis (2021)** categorise the data sources for FOC modelling as follows:

(Daily) Noon Report

These reports are manually prepared by the ship's chief engineer and transmitted by the ship's masters to the shipping company and shore management daily. They contain details about the daily fuel consumption, fundamental voyage particulars (such as the ship's position and load status), sailing behaviour details (such as average sailing speed and average engine RPM), and information about sea and weather conditions. While these reports offer valuable insights into ship operations, the limitation lies in the manual and daily nature of data entry, which can impact both data quality and quantity.

Sensor Data

Data obtained from onboard installed sensors on the vessel, which might encompass sensors like fuel flow sensors, Global Positioning System (GPS) receivers, and wind speed sensors. These sensors offer a solution to the data quantity issue associated with noon reports, as highlighted in the study conducted by **Gkerekos, Lazakis, and Theotokatos (2019)** in the context of daily FOC prediction. In their study, the machine learning models derived from the Automated Data Logging and Monitoring (ADLM) system demonstrate superior performance compared to models trained using noon data. This improvement amounts to around $5-7\%$ for a data collection span of 3 months using the ADLM system against 2.5 years of noon data. Nevertheless, the implementation of onboard sensors can be intricate and expensive **(Petersen 2011)**, requiring proper management of resulting sensor data to account for potential measurement errors.

AIS Data

Apart from its primary purpose as a collision avoidance system, AIS data has shown potential for applications in ship behaviour analysis and environmental assessment. The International Maritime Organization (IMO) used AIS data for the study of Greenhouse Gas (GHG) emissions **IMO (2020); Smith et al. (2015)**, employing it to estimate global shipping emission inventories. **Rakke (2016)** introduced the ECAIS methodology to compute ship emissions based on vessel information derived from AIS data and literature review. By utilising the Holtrop-Mennen approximation and relevant literature-based approximations, the vessel's propulsion power can be determined, subsequently enabling the prediction of FOC. **Kim et al. (2020)** employed

publicly accessible AIS data, ship static information, and environmental data, to estimate the Energy Efficiency Operational Indicator (EEOI) by using big data technology. In essence, research utilising AIS data is directed at achieving independence from reliance on commercial databases. Further elaboration on AIS data can be found in Section 2.3.

2.1.3 Review of ML approach to predict FOC

Modelling of FOC using *machine learning* (ML) approach generally focus on the prediction of unseen data. The general framework usually includes the collection and preprocessing of ship operational data, training and validation of the model, and evaluation and selection of the most appropriate model. Some machine learning models allow further hyperparameter tuning of the model. The data is generally split into training and testing datasets. The test dataset is used to evaluate the performance of the generated ML model on unseen data.

The study by **Yan, Wang, and Psaraftis (2021)** indicated that the majority of recent research that uses machine learning approach employs Artificial Neural Network (ANN) as the model to predict FOC. ANN models are powerful models capable of modelling nonlinear data which are based on theories on how the brain works. The outcome is modelled by an intermediate set of unobserved variables known as hidden layer **(Kuhn and Johnson 2013)**. Backpropagation neural networks, Multi Level Perceptron (MLP), and wavelet neural networks are some examples of ANN model subclasses.

ANN has shown notable performance in its attempt to predict FOC. **Petersen, Jacobsen, and Winther (2012)** reported Root Mean Square Error (RMSE) of 47.2 L/h for for fuel flow (FOC) prediction. To provide context, the fuel flow in their case study fluctuates between $1000 - 2500$ L/h. **Bal Beşikçi et al. (2016)** considered sailing speed, trim, wind, sea effects, propeller pitch, and engine rotation speed as input variables to predict FOC per hour, achieving model fit score of $R^2 = 75.9\%$ in the test set. Similar prediction performance indices have been reported in other studies utilising ANNs **(Yan, Wang, and Psaraftis 2021)**.

However, the development of ANN models is a challenging task. ANN models tend to overfit in situations where data availability is limited. Therefore, regularisation is necessary to improve model performance, but it requires an intricate balancing process during the regularisation process and unsuitable regularisation may lead to counterintuitive prediction results. Furthermore, the process of adding layers is computationally resource-expensive and does not always guarantee promising results **(Hastie, Tibshirani, and Friedman 2009)**. Moreover, from ML perspective, ANN is classified as a black box model, which makes it unintuitive and lacking in interpretability **(Géron 2019)**, this particular limitation cause shipping industry expert generally reluctant to accept the model generated using ML approach.

2.1.4 Tree-Based Model as FOC model

Concerning interpretability, modelling approaches such Linear Regression (LR), k-Nearest Neighbour (KNN) and tree-based models have shown superior interpretability in comparison to ANNs. LR can explain the effect of each input variable on the output through the coefficients. KNN searches for the nearest neighbour and their closeness is evaluated through distance measurement algorithms such as Euclidean distance. Additionally, LRs and KNNs also offer easy implementation and adequate explainability. However, both approaches suffer from sensitivity to outliers and noise in data.

This brings us to the tree-based model, a supervised, highly interpretable machine learning modelling approach capable of performing classification tasks for discrete data and regression tasks for continuous data. According to the summary of **Yan, Wang, and Psaraftis (2021)**, it is not as popular as ANN, however, some literature work and studies have indicated its benefits and performance superiority over other machine learning modelling approaches:

Soner, Akyuz, and Celik (2018) used the ferry dataset from **Petersen, Jacobsen, and Winther (2012)** to predict FOC using tree-based model, which includes bagging, random forest (RF), and bootstrap. From the test dataset, the random forest model achieved RMSE of 43.5 L/h for the fuel consumption. Which suggested improvement from ANN model from the study of **Petersen, Jacobsen, and Winther (2012)**.

Yan, Wang, and Du (2020) used random forest (RF) model to predict FOC for a voyage of a dry bulk ship using ship operational data which includes ship noon data and sea weather data from noon report and ECMWF. The prediction model considered ship sailing speed, total cargo weight and meteorological conditions for the prediction. The trained RF model obtained mean absolute percentage error (MAPE) of 7.91% for the FOC prediction and displayed superior results in comparison to Decision Tree Regressor (DTR), ANN, LASSO, and SVR.

The advantages of tree-based model are further highlighted by **Gkerekos, Lazakis, and Theotokatos (2019)**. The study compared the performance of different machine learning models to predict ship's FOC per day using both noon data and automated data logging and monitoring (ADLM) system from a bulk carrier. This research concludes that tree-based model displayed good prediction performance on both noon data and sensor-based data. ETR achieved remarkable model fit score of 89% using the noon data and 97% when using the data from ADLM system, outperforming ANN, SVR, and RFR models.

Li et al. (2022) performed more extensive research on the effects of data fusions between meteorological data, ship voyage data, and AIS data on different machine learning models to predict the ship's FOC. The study classified ETR and RFR as tree-based model which is produced by *bagging ensemble strategy*. While AdaBoost (AB),

Gradient Tree Boosting (GB), XGBoost(XG) and LightGBM (LB) are classified as tree-based models produced by *boosting ensemble strategy*. The study recommends all tree-based models that are produced by *boosting ensemble strategy* and ETR be used to model energy-efficient operation. Additionally, RFR shows the best robustness among the proposed models in the study.

Abebe et al. (2020) attempted to use the ML approach to predict SOG of the ship. In this study, AIS data and noon-report weather data from 14 tracks and 62 ships are used for model training. The generated model considered the ship draught, ship dynamic information, tonnage, and environmental conditions. The result of this study exhibited the feasibility of using AIS data and meteorological data to predict SOG of the ship and the results also further indicated the strength of the tree-based model. On the test dataset, ETR achieved the best result with R^2 score of 98.47% and RMSE of $0,234$ knots. It is also reported that ETR achieved better performance with about half of the computational cost of RFR.

2.1.5 Review of WBM for FOC prediction

To predict the FOC of a ship, WBMs generally calculate the resistances encountered by the vessel based on physics and hydrodynamic laws. The total resistance is cumulated from calm water resistance and additional resistance from wind, wave, and other external factors. This, in turn, allows for the determination of the corresponding engine power at a given speed which allows the calculation of the FOC **(Haranen et al. 2016)**.

The methods from **Guldhammer and Harvald (1974)**, **Hollenbach (1999)**, and **Kristensen and Lützen (2012)** use different formulations, assumptions, and input variables for engine power estimation. For this thesis, the main focus will be the use of the estimation method from Holtrop-Mennen **(Holtrop and Mennen 1978, 1982; Holtrop 1984)**. Holtrop-Mennen's allowable application range is suitable in most cases indicated by studies from **Rakke (2016)** and **Kim et al. (2020)**. **Rakke (2016)** used ship operational and mechanical data from various works of literature and AIS data for input variables to estimate the engine power using Holtrop-Mennen method to subsequently calculate FOC. The FOC is then used to estimate GHG emissions for different ships and the study reported about 5% error rate during model testing. **Kim et al. (2020)** successfully estimated Energy Efficiency Operational Index (EEOI) without actual FOC. The study used AIS data as well as publicly accessible weather data and ship static information.

2.1.6 Conclusion of literature review

As categorised by **Yan, Wang, and Psaraftis (2021)**, the GBM model in this thesis falls under the category of sequential GBM, Here, the BBM and the WBM are sequentially developed and then combined to form the GBM. The BBM is designed for

predictions of unseen data, and its outcomes are subsequently fed into the WBM. The use of the tree-based ML model addresses the challenge of limited interpretability faced by certain machine learning models. Furthermore, tree-based models can outperform most of the available machine learning models while providing added benefits of little requirement for data preprocessing and relatively cheap computational cost. The selection of Holtrop-Mennen as energy estimation method is justified by the application range of the methodology and reported successes in prior studies.

2.2 Tree-based model

In this section, the theory of Decision Tree (DT), Random Forest (RF) and Extra Tree (ET) will be discussed in detail in Section 2.2.1, Section 2.2.2 and Section 2.2.3, respectively.

2.2.1 Decision Tree

The principle of the decision tree as a predictor can be defined as one or more nested `if-then` statements based on a rule that partitions the data into partition space as shown in Figure 2.1. Alternatively, the partition space formed through `if-then` statements can also be visualised using a binary tree representation, which offers greater interpretability since various input responses can be represented through a single tree **(Kuhn and Johnson 2013; Hastie, Tibshirani, and Friedman 2009)**.

A decision tree consists of the following type of nodes, ***Root node*** defines the topmost node. ***Leaf nodes*** are also termed as terminal nodes, it is the node that will give the final prediction output. The ***Internal Node*** is defined as the nodes between the root node and leaf node. The process of dividing a node into successive nodes is called ***splitting***. The node that is being split is called ***parent node*** and the successive nodes that are created are called ***child nodes***. To grow a tree in a regression task, the splitting process is typically regulated by Mean Square Error (MSE). The tree growth algorithm is based on Classification and Regression Tree (CART).

To understand the principle of selection for the feature, k_t , of the parent node and splitting rule, t_k , for data partition, the following example will be presented:

For the selection of the optimal splitting rule t_k: Given a case with single feature k and response y with n data points present. The algorithm starts by looking for possible splits between two distinct data points y. This split results in two distinct partition spaces. For each partition space S_1 and S_2, the mean is calculated by dividing the sum of response y with the amount of data points n for each respective partition space S_1 and S_2.

This step is then followed by calculating the sum of squared error (SSE) of each data

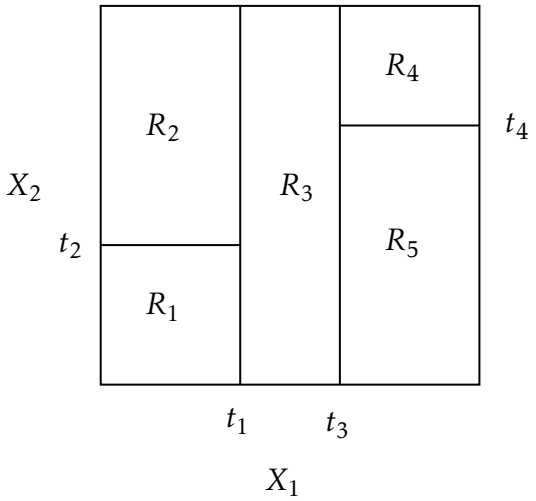

Figure 2.1: Example of partition space **(Hastie, Tibshirani, and Friedman 2009)**

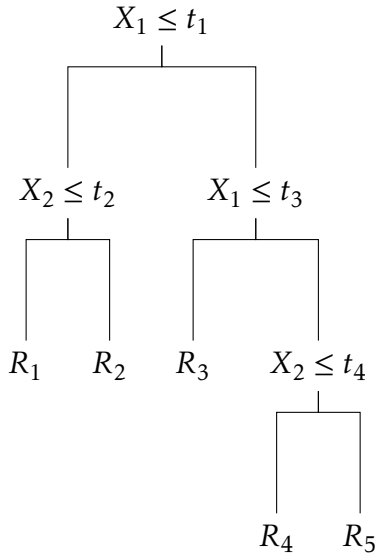

Figure 2.2: Example of partition tree **(Hastie, Tibshirani, and Friedman 2009)**

point in partition space S_1 and S_2, followed by division by the respective numbers of data points, n_{S1} and n_{S_2}, to calculate the mean sqaured error (MSE). Next, the MSE values from the partition spaces S_1 and S_2 are aggregated. This iterative process continues recursively until a threshold t_k is identified. which yields the minimum sum of MSE, this threshold will be selected as splitting rule for the parent node and correspond to the threshold that minimises the cost function $J(k, t_k)$, Here, $\hat{y}_{Si}$ represents the mean of the response variable y_{S_i} within the partition space S_i. **(Géron 2019; Kuhn and Johnson 2013)**.

$$\text{MSE}_{S_i} = \frac{1}{n_{S_i}} \text{SSE}_{S_i} \quad \textbf{where} \quad i = (1, 2) \tag{2.2.1}$$

$$J(k, t_k) = \frac{1}{n_{S_1}} \text{SSE}_{S_1} + \frac{1}{n_{S_2}} \text{SSE}_{S_2} \begin{cases} \text{SSE}_{S_i} = \sum\limits_{i \in S_i} (\hat{y}_{S_i} - y_{S_i})^2 \\ \hat{y}_{S_i} = \frac{1}{n_{S_i}} \sum\limits_{i \in S_i} y \end{cases} \tag{2.2.2}$$

For the selection of the most optimal feature for parent node t_k: Similar principle is also applied for the selection of the most optimal feature for the parent node. Consider there are k_t features, then for each respective feature $k_1, k_2, \ldots, k_t$, The MSE for each of the features is calculated following the cost function $J(k, t_k)$. The feature that can best ***minimise*** the cost function will be selected as the root node of the tree. The subsequent selections of the feature for the parent node follow the same principle. **(Hastie, Tibshirani, and Friedman 2009; Géron 2019)**. After this step, the partition space is subsequently divided into two additional regions in order to identify the next potential split that minimizes the cost function $J(k, t_k)$. This recursive process continues until either the number of samples for splitting reaches a predefined threshold or when no further reduction in MSE is possible.

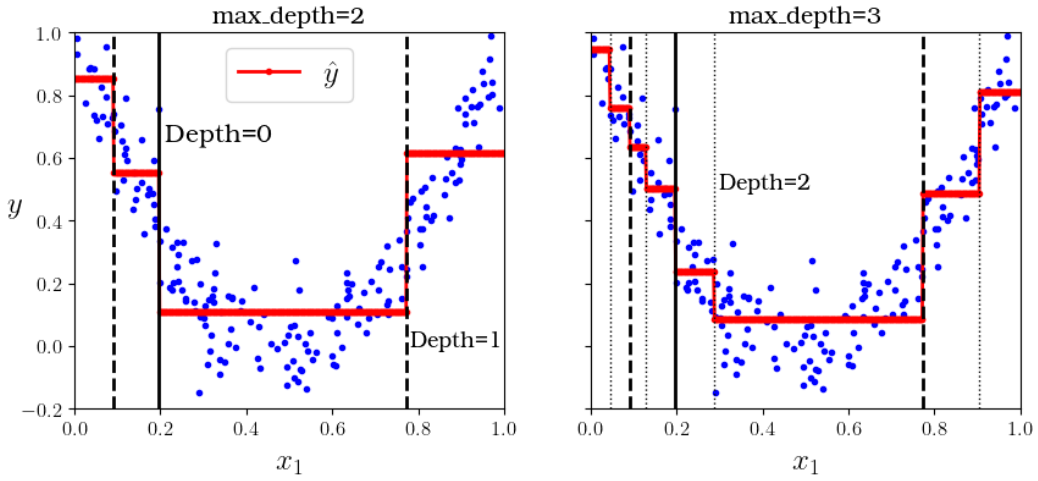

Figure 2.3: Prediction of two Decision tree regression models **(Géron 2019)**

The resulting decisions for the best possible splits can be represented using a binary tree, this makes the decision tree highly interpretable and easy to implement. The inherent logic structure from `if-then` statements means that it can handle various types of data (sparse, skewed, continuous, categorical, etc.) without the need for data pre-processing. Decision tree implicitly conducts feature selection which is a desirable trait for many modelling problems **(Kuhn and Johnson 2013)**.

However, a single decision tree suffers from overfitting when the model is unconstrained. The logical principle of `if-then` statements means that decision tree will attempt to fit the training data as closely as possible. Furthermore, a single decision tree model tends to be unstable, altering the data will cause drastic changes in the structure of the tree, there exist possibilities where completely different sets of splits might be found resulting in different interpretations **(Hastie, Tibshirani, and Friedman 2009; Kuhn and Johnson 2013)**. From Figure 2.1, it can be implied that each decision boundaries are orthogonal to an axis i.e. all splits are perpendicular to an axis and this form rectangular subspaces for each predicted value. If the relationship between predictors and response cannot be adequately defined by the rectangular subspaces, then tree-based models will suffer from larger prediction errors than other kinds of models **(Kuhn and Johnson 2013)**.

Therefore, it is necessary to regularise i.e., restrict the decision tree's freedom to grow during model training. Overfitting could be reduced by controlling the count of nodes along a path from the root node to its parent node through the `max_depth` parameter. Additionally, setting the amount of minimum number of samples a leaf node has, through `min_samples_leaf` can alleviate overfitting as well, as shown in Figure 2.4. Other regularisation techniques will be discussed in Section 3.3.2. Regularisation may result in better generalisation capability. Nonetheless, in order to attain significant improvements in the performance of the decision tree model, it is necessary to seek alternative solutions.

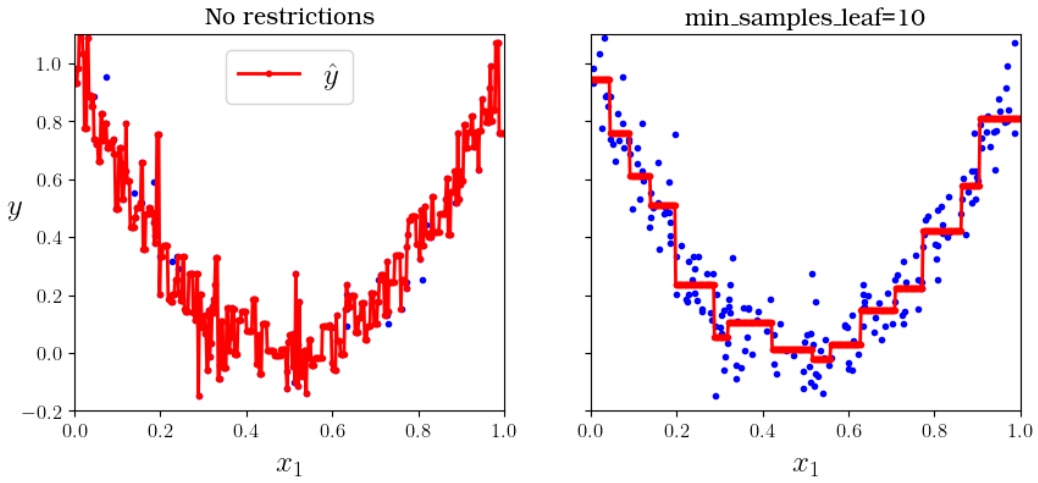

Figure 2.4: Regularising a Decision Tree regressor **(Géron 2019)**

2.2.2 Random Forest

Ensemble learning is one of the possible solutions to improve the performance of DT regressors. The main idea of ensemble learning is combining the strengths of a collection of simpler base models **(Hastie, Tibshirani, and Friedman 2009)**. The algorithm, involving the creation of bootstrap samples, random selection of splitting feature and aggregation of the prediction is termed by **Breiman (2001)** as ***Random Forest***. It involves the combination of multiple learning algorithms, known as weak learners, where each of these learners corresponds to an individual decision tree in the case of Random Forest.

The most common ensemble methods are *boosting* and *bagging*. In boosting, the learner evolves, where successive trees are dependent on the earlier trees. In bagging (short for *bootstrap aggregating*) each tree is trained using bootstrap sample of the training set i.e. this means that a sample of the training dataset is randomly selected and allowed to appear more than once[1]. Each model in the ensemble then generates a prediction from the bootstrapped sample and the predictions are aggregated across the learners **(Kam Ho 1995; Breiman 2001)**.

The performance of bagging can be further improved by reducing the correlation between trees i.e. de-correlating trees. This can be achieved by adding randomness during the tree construction process. **Dietterich (2000)** introduced the idea of random split selection, which means that a feature k will be selected from a random subset of features. From this random subset, the assignment of the feature for the parent node follows the CART algorithm described in Equation (2.2.2). Further randomness is added by exploiting the instability of a single decision tree mentioned in Section 2.2.1.

The methodology introduced in random forest address the tendency of decision tree to overfit and the issue of lack of robustness. De-correlating trees means that each

[1] This sampling technique is referred to as sampling *with* replacement

learner is independent of the other, and the combination of many independent, strong learners yields an improvement in error rates i.e. the reduction in variance and robustness against noisy response. It is also proven by **Breiman (2001)** that random forest cannot overfit, which means growing more trees should not affect the performance of random forest, albeit with a greater computational burden. Both **Kuhn and Johnson (2013)** and **Hastie, Tibshirani, and Friedman (2009)** reported that remarkable prediction results can be obtained without extensive tuning of tree parameters.

However, random forest (RF) loses the benefit of interpretability of Decision Tree (DT) model. While it remains feasible to trace the path within an individual tree, which is useful to comprehend the decisions leading to a particular prediction, the ensemble nature of random forest hinders the possibility to gain an understanding between the feature and the prediction. Nevertheless, it is still possible to quantify the impact of each feature in the ensemble through the feature importance **(Kuhn and Johnson 2013)**. Random forest also tends to perform poorly with a small number of samples **(Hastie, Tibshirani, and Friedman 2009)**.

2.2.3 Extra-Trees (Extremely Randomised Trees)

Extra-trees (Extremely Randomised Trees) is introduced by **Geurts, Ernst, and Wehenkel (2006)** to further randomise random forest and further de-correlate the trees in the forest. Unlike random forest, which selects the optimal split by selecting the best feature among randomly selected subset of features, Extra-trees selects a split at random. Extra-trees also does not bootstrap the sample[2] and uses the whole training dataset. The random selection of split means that it saves computational power and the increase in variance caused by tree de-correlation can be countered by increasing the number of trees in the ensemble.

[2] This sampling technique is referred to as sampling *without* replacement

2.3 AIS Data

2.3.1 Overview of AIS

Automatic Identification System (AIS) is an automated tracking system onboard ships that was developed automatically transmit information about a ship to other ships and coastal authorities to avoid ship collision accidents. As part of the revised new chapter V of SOLAS[3] regulation, International Maritime Organization (IMO) requires all international voyage ships of 300 gross tonnage (GT) and upwards, cargo ships with 500 GT not engaged on international voyage, and all passenger ships irrespective of size to be equipped of AIS class A equipment **(Yang et al. 2019; IMO 2015)**.

AIS uses Very High Frequency (VHF) with special protocol for communication system for information exchange between the ships. This information will be received by ships directly, buoys, Land-based (terrestrial) AIS transceivers (T-AIS) and satellites (S-AIS). The information transmitted by AIS is distinguished into three different types. **Static information** which is entered into the AIS on installation, **dynamic information**, which is automatically updated from the ship's sensors connected to AIS and **voyage-related information**, which might required manual entry and updating during the voyage. The structure of the AIS data that is relevant to this thesis is summarised in Table 2.1**(IMO 2015)**.

AIS is also further differentiated by its equipment class. The classification is based on the reporting interval and the type of information that is conveyed. **Class A** autonomously report their position within 2-10 seconds intervals, depending on the state of the ship's movement. The reporting interval is less frequent, occurring every 3 minutes, particularly when the ship is at anchor, moored, or moving at a speed slower than 3 knots. **Class A AIS** is also equipped to transmit safety-related information, meteorological and hydrological data, electronic broadcasts to mariners, and marine safety messages. On the other hand, **Class B AIS** reports at longer intervals and with lower power. Class B AIS can only receive safety-related messages and is not capable of sending them. **Rakke (2016); IMO (2015)**

[3] International Convention for the Safety of Lives at Sea

Information Item	Description
Static	
MMSI	MMSI number of vessel
Callsign	Callsign of vessel
Name	Name of the vessel
IMO	IMO number of the vessel
Length	Length of vessel
Width	Width of vessel
Ship Type	Describes the AIS ship type of this vessel
Dynamic	
Ship's position	Automatically updated from position sensor connected to AIS. Longitude and Latitude.
Position time stamp in UTC	Automatically updated from ship's main position sensor. Format: DD/MM/YYYY HH:MM:SS
Course over Ground (COG)	***If available***, automatically updated from ship's main position sensor connected to AIS.
Speed Over Ground (SOG)	***If available***, automatically updated from the position sensor connected to AIS.
Heading	Automatically updated from the ship's heading sensor connected to AIS
Navigational status	Navigational status information has to be manually entered by the Officer on Watch (OOW) and changed as necessary. For example : "*underway by engines*","*engaged in fishing*","*at anchor*".
Rate of Turn (ROT)	***If available***, Automatically updated from the ship's ROT sensor or derived from the gyro.
Voyage Related	
Ship's draught	To be manually entered at the start of the voyage using the maximum draft for the voyage and amended as required
Cargo Type	Type of (hazardous) cargo from AIS message.
Destination and ETA	To be manually entered at the start of the voyage and kept up to date as necessary.

Table 2.1: Structure of AIS data **(IMO 2015)**

It is also stated by **Yang et al. (2019)** that AIS data can be combined with data from other databases to provide additional information such as:

- Port-to-port average speed: the voyage time can be calculated from the time stamps reported by AIS data; the voyage distance can be found from corresponding navigation distance tables.
- Cargo weight which can be estimated from draught and ship size.
- Technical ship specification from fleet database which can be derived from IMO number.
- Port-to-port bunker consumption which can be estimated based on the speed, technical ship specification and distance between two ports.

2.3.2 Speed Correction

The speed displayed in AIS represents the speed over ground (SOG). However, to calculate bunker fuel consumption, the ship's actual speed, known as speed through water (STW), is required. Therefore, a correction needs to be applied to SOG to obtain STW. This correction is carried out by considering the current speed v_C and the current direction γ *with respect to True North*. In principle, STW will be greater than SOG when the ship is moving against the current, as the ship needs to compensate for the current to maintain the SOG. Conversely, STW will be less than SOG when the ship is moving in the same direction as the current.

To calculate the correction, this study will adopt the methodology proposed by Kim et al. **(Kim et al. 2020)** and Yang et al. **(Yang et al. 2020)**. The x and y components of SOG can be obtained through vector decomposition using the ship's heading angle α *with respect to True North*. Similar vector decomposition is also performed for current speed v_C, it is resolved with current direction γ *with respect to True North*:

$$v_G^x = v_G \cdot \sin(\alpha) \tag{2.3.1}$$

$$v_G^y = v_G \cdot \cos(\alpha) \tag{2.3.2}$$

$$v_C^x = v_C \cdot \sin(\gamma) \tag{2.3.3}$$

$$v_C^y = v_C \cdot \cos(\gamma) \tag{2.3.4}$$

Then the resulting equation to determine STW,v_S, including the current compensation, is given by:

$$v_S^x = v_G^x - v_C^x \tag{2.3.5}$$

$$v_S^y = v_G^y - v_C^y \tag{2.3.6}$$

$$v_S = \sqrt{(v_S^x)^2 + (v_S^y)^2} \tag{2.3.7}$$

2.3.3 Source of error in AIS

AIS data may still contain errors and inaccuracies. Manual data entry, particularly for static information and voyage-related details such as estimated time of arrival (ETA) and draught, is a primary source of errors. Instances have been observed where the Maritime Mobile Service Identity (MMSI) is shared by different ships, despite its intended uniqueness. Furthermore, data collected automatically by sensors can also be erroneous, which may arise from sensor malfunctions or improper installations **(Yang et al. 2019)**. Therefore, it is necessary to preprocess the AIS data to ensure an accurate representation of the ship's state during her voyage.

2.4 Weather data

Throughout a voyage, a vessel can encounter winds and waves originating from different directions and with varying magnitudes. These atmospheric and hydrodynamic factors can influence the vessel's trajectory during the journey, impact vessel performance elements like speed and engine power, and even affect a vessel's ability to navigate challenging sea conditions i.e. a vessel's seakeeping capabilities **(Molland 2011)**. To ensure an accurate estimation of the engine power needed by the vessel, it is important to account for different weather conditions. Therefore, this section will primarily focus on the definition of wind and wave effects and explore the correlation between some of these key parameters.

2.4.1 Definitions of weather parameters

Wind Waves and Swell

Wind Waves are also known as wind sea, wind waves are irregular and short-crested waves generated by local wind. **Swell** are waves that travel outside the wave generation area and are no longer the result of wind, they take on regular and long-crested appearance **(Holthuijsen 2007)**

Significant Wave Height, $H_{1/3}$

It is defined as the mean of the highest one-third of waves in the wave record. The distribution of wave heights can be represented by probability density function. Hence, the term "highest one-third of waves" here means the region of wave heights that belong in the upper one-third of a probability density function, this is illustrated in Figure 2.5. From this distribution, the relation between significant wave height $H_{1/3}$, the highest ten percent of waves H_{10}, maximum wave height H_{max} and average wave height $\overline{H}$ can be summarised as follows **(Bretschneider 1965; Holthuijsen 2007)**:

$$\overline{H} = 0.625 \cdot H_{1/3} \tag{2.4.1}$$

$$H_{10} = 2.03 \cdot \overline{H} = 1.27 \cdot H_{1/3} \tag{2.4.2}$$

$$H_{\text{max}} = 2 \cdot H_{1/3} \tag{2.4.3}$$

Additionally, **Bitner-Gregersen (2005)** and **Nielsen and Dietz (2020)** described the relation between the significant wave height, wind wave height and swell height through the following equation:

$$H_{1/3} = \sqrt{(H_{\text{swell}})^2 + (H_{\text{windwave}})^2} \tag{2.4.4}$$

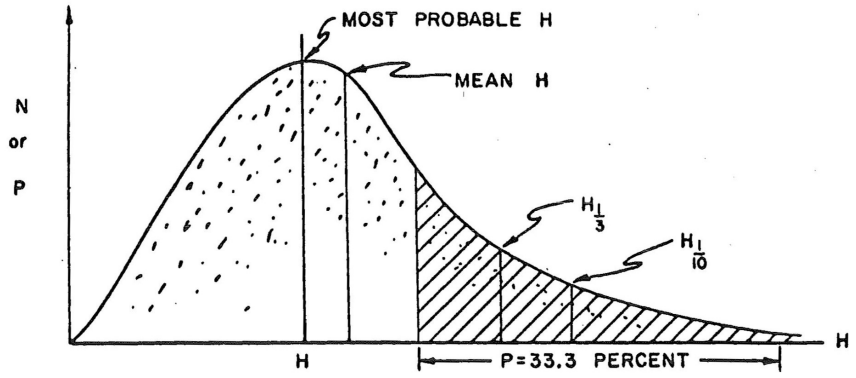

Figure 2.5: Statistical distribution of wave heights **(Bretschneider 1965)**

Wave Period

Defined as the time interval between the start and the end of a wave. Some characteristics of wave period can be derived to define wave spectrum.

Wave Spectrum

The most important form in which ocean waves are described. Wave spectrum characterises all possible observations of the waves which include wave heights, frequencies i.e. period and wave direction. For example, **Bitner-Gregersen (2005)** stated that the state of the sea can be described through the significant height $H_{1/3}$ and spectral peak period T_p with the help of Torsethaugen peak, given the spectral peak period of fully developed sea T_f and constant $a_f = 6.6$ **(Torsethaugen and Haver 2004)**.

$$T_p = a_f \cdot H_{1/3} \tag{2.4.5}$$

$$\text{Sea State (SS)} = \begin{cases} \text{Swell dominated} & \textbf{if} \quad T_p > T_f \\ \text{Wind sea dominated} & \textbf{if} \quad T_p \leqslant T_f \end{cases} \tag{2.4.6}$$

2.5 General concept of ship propulsion

A ship's bunker fuel consumption in actual operating conditions is affected by several factors including the operating parameter of the ship's engine, propeller efficiency, and encountered resistance by the ship. Furthermore, a ship's propulsion power is correlated to the sailing speed (SOG) and meteorological conditions **(Lang 2020)**. Therefore, in addition to the calm water resistance R_{CALM}, the additional resistance caused by wind R_{AA} and wave R_{AW} should be considered to estimate the total resistance of the ship R_{TOTAL}. The power needed to propel a ship forward at a given ship STW v_S, to overcome R_{TOTAL} is defined as **effective power** P_e:

$$R_{TOTAL} = R_{CALM} + R_{AW} + R_{AA} \tag{2.5.1}$$

$$P_e = R_{TOTAL} \cdot v_S \tag{2.5.2}$$

The effective power P_e is transmitted through the shaft connected to the main engine of the ship which generates power to rotate the propeller of the ship, which is termed as **brake power of the engine,** P_b. The brake power can be calculated through effective power by considering the **shaft efficiency** η_s**, hull efficiency** η_h**, relative rotative efficiency** η_r **and open water efficiency** η_o:

$$P_b = \frac{P_e}{\eta_s \cdot \eta_h \cdot \eta_r \cdot \eta_o} \tag{2.5.3}$$

The bunker fuel consumption can then be calculated by multiplying the brake power P_b with the Specific Fuel Oil Consumption (SFOC) and the operation time τ_{OP}:

$$FOC = P_b \cdot SFOC \cdot \tau_{OP} \tag{2.5.4}$$

2.5.1 Ship dimensions and form coefficients

Principal Dimension of a vessel

The summary of important ship dimensions and parameters are shown in Figure 2.6 and Figure 2.7 **(Biran and López-Pulido 2014)**:

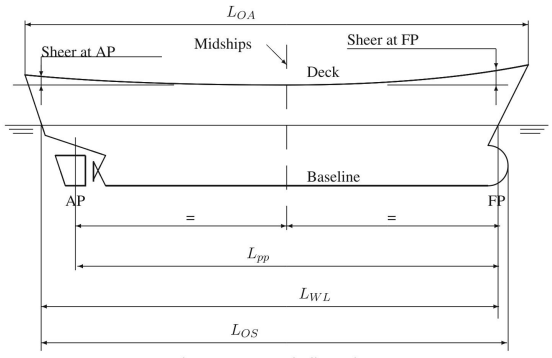

Figure 2.6: Side view of a vessel

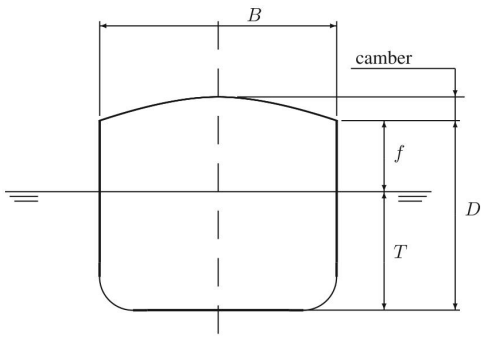

Figure 2.7: Front view of a vessel

The outer surface of the ship is usually not uniform as not all plates have the same thickness, Therefore the hull surface is measured with respect to the inner surface of the plating which is termed as *moulded surface* of the hull. All dimensions measured to this surface are defined as *moulded* dimensions whereas dimensions measured to the outer surface of the hull or of an appendage are qualified as *extreme* dimensions **(Biran and López-Pulido 2014)**.

Coefficients of form

The form coefficients are non-dimensional numbers required to classify the hulls and to find relationships between forms and their properties, the summary of some important form coefficients are summarised in Figure 2.8

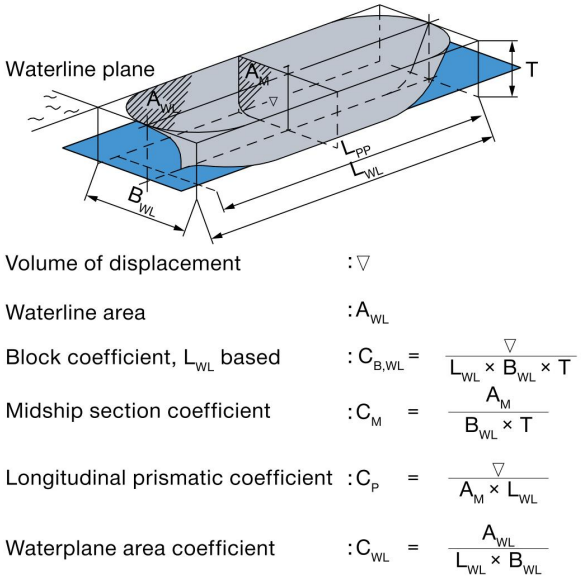

Figure 2.8: Form coefficients **(MAN 2011)**

Block Coefficient

Block Coefficient C_B is defined as the ratio of moulded displacement volume to the volume of parallelepiped (rectangular block) with dimensions L, B and T. Alternatively, **Schneekluth and Bertram (1998)** provided an estimation of the value using the Froude number within the range of $0.15 < Fr < 0.32$

$$C_b = -4.22 + 27.8\sqrt{Fr} - 39.1Fr + 46.6Fr^3 \tag{2.5.5}$$

The Froude number Fr is defined with the following equation:

$$Fr = \frac{v}{\sqrt{gL_{WL}}} \tag{2.5.6}$$

Midship Coefficient

Midship Coefficient C_M is defined as the ratio of the midship-section area A_M to the product of breadth and draught, BT. According to **Schneekluth and Bertram (1998)**, changing C_M value will have an effect on separation resistance and wave resistance. **Jensen (1994)** presented a method based on regression equation on a graph to calculate C_M:

$$C_M = \frac{1}{1 + (1 - C_B)^{3.5}} \tag{2.5.7}$$

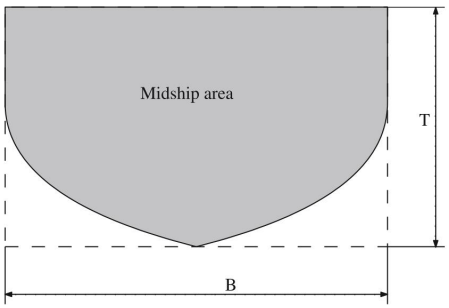

Figure 2.9: Definition of C_M **(Biran and López-Pulido 2014)**

Prismatic Coefficient

Prismatic Coefficient C_P is defined as the ratio of moulded displacement volume[4] V. It is an indicator on how much of a cylinder with constant section A_M and length L is filled with submerged hull as shown in Figure 2.10.

$$C_P = \frac{C_B}{C_M} \tag{2.5.8}$$

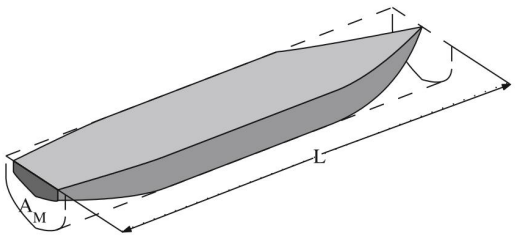

Figure 2.10: Definition of C_P **(Biran and López-Pulido 2014)**

[4] In some notations it is denoted as ∇

Waterplane area coefficient

Waterplane area coefficient C_{WP} is defined as the ratio between the ship's waterline area A_W and the product of L and B. In ship design, C_{WP} significantly impacts resistance and stability **(Schneekluth and Bertram 1998)**. **MAN (2011)** approximated that C_{WP} is 0.10 higher than C_B, alternatively **Schneekluth and Bertram (1998)** provided the following formulation for C_{WP}:

$$C_{WP} = \frac{1 + 2C_B}{3} \tag{2.5.9}$$

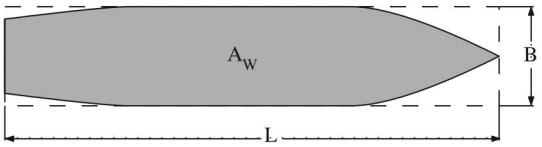

Figure 2.11: Definition of C_{WP} **(Biran and López-Pulido 2014)**

2.5.2 Holtrop & Mennen's Method

This power prediction method was applied in the late 1970s and early 1980s by J. Holtrop and G.G.J Mennen and it was based on regression analysis of vast model tests and trial data of MARIN, the model basin in Wageningen, The Netherlands. This gives Holtrop-Mennen method a wide applicability range and the only method that adopted the use of the ITTC form factor k. The resistances in this method are calculated as dimensional force. Furthermore, the method also gives estimates of hull-propeller interaction, thrust deduction, full-scale wake fraction and relative rotative efficiency **(Birk 2019)**.

Application Range

The publication from **Holtrop and Mennen (1978, 1982); Holtrop (1984)** does not provide explicit information regarding the application range of the method. However, from the experience of **Birk (2019)**, reasonable estimates from the method can be achieved for the following conditions:

$$\begin{gathered} Fr \leqslant 0.45 \\ 0.55 \leqslant C_p \leqslant 0.85 \\ 3.9 \leqslant \frac{L}{B} \leqslant 9.5 \end{gathered} \tag{2.5.10}$$

2.5.2.1 Calm water resistance

The calm water resistance R_{CALM} is broken down into several components and can be approximated using the following relation:

$$R_{CALM} = R_F(1 + k_1) + R_{APP} + R_W + R_B + R_{TR} + R_A \tag{2.5.11}$$

Parameter	Symbol	Remarks
Required Parameters		
Length in waterline	L_{WL}	
Moulded breadth	B	
Moulded mean draught	T	typically $T = \frac{1}{2}(T_A + T_F)$
Moulded draught at aft perpendicular	T_A	
Moulded draught at forward perpendicular	T_F	
Volumetric displacement (molded)	V	alternatively use the block coefficient as $C_B = V/BTL_{WL}$
Prismatic coefficient (based on L_{WL})	C_P	
Midship section coefficient	C_M	or use $C_M = C_B/C_P$
Waterplane area coefficient	C_{WP}	may have to be estimated in early design stages
Longitudinal Centre of buoyancy	ℓ_{CB}	positive forward; with respect to $L_{WL}/2$ in percent of L_{WL}
Area of ship and cargo above waterline	A_V	projected in direction of v_S
Immersed transom area	A_T	measured at rest
Transverse area of bulbous bow	A_{BT}	Measured at forward perpendicular
Height of centre A_{BT} above basis	h_B	has to be smaller than $0.6T_F$
Propeller Diameter	D	
Propeller expanded area ratio	A_E/A_0	
Stern shape parameter	C_{stern}	
Optional Parameters		
Wetted surface (hull)	S	
Wetted Surface of appendages	S_{App}	bilge keels, stabiliser fins, etc.
Half angle of waterline entrance	i_E	
Diameter of bow thruster tunnel	d_{TH}	

Table 2.2: Required and optional input parameters for Holtrop & Mennen's method according to **Birk (2019)**

Frictional Resistance R_F

R_F is calculated using the ITTC-1957 frictional resistance correlation line C_F as the basis of a representation of a resistance plate with a wetted surface area S of bare hull.

$$R_F = \frac{1}{2}\rho v_S^2 S C_F \tag{2.5.12}$$

The frictional coefficient C_F can be calculated through the Reynold number Re for a given ship speed v_S and kinematic viscosity ν:

$$C_F = \frac{0.075}{[\log_{10}(Re) - 2]^2} \quad \textbf{where} \quad Re = \frac{v_S L_{WL}}{\nu} \tag{2.5.13}$$

If not known, then the wetted surface area of bare hull S can be estimated by the following formula:

$$S = c_{23} L_{WL}(2T + B)\sqrt{C_M} + 2.38\frac{A_{BT}}{C_B} \tag{2.5.14}$$

with the factor c_{23} given as :

$$c_{23} = \left[0.453 + 0.4425C_B - 0.2862C_M - 0.003467\frac{B}{T} + 0.3696C_{WP}\right] \tag{2.5.15}$$

The flat plate resistance is subsequently adjusted by including a form factor k during the calculation of total resistance. The constant c_{14} must be determined first to calculate form factor k, which serves the purpose of capturing the impact of the aft body shape.

$$c_{14} = 1.0 + 0.011C_{stern} \quad \textbf{with} \quad \begin{array}{|l|c|} \hline \text{Aft body shape} & C_{stern} \\ \hline \text{Pram with gondola} & -25 \\ \text{V-shaped sections} & -10 \\ \text{Normal sections} & 0 \\ \text{U-shaped sections} & +10 \\ \hline \end{array} \tag{2.5.16}$$

To complete the required input for the calculation of $(1 + k_1)$, the length of run L_R can be estimated from the following equation:

$$L_R = L_{WL}\left(\frac{1 - C_P + 0.06C_P\ell_{CB}}{4C_P - 1}\right) \tag{2.5.17}$$

The formula by **Guldhammer and Harvald (1974)** can be used if ℓ_{CB} is not known:

$$\ell_{CB} = -(0.44Fr - 0.094) \tag{2.5.18}$$

Then, the form factor $(1 + k_1)$ can be determined with the constant c_{14}, the length of run L_R and input values from Table 2.2.

$$1 + k_1 = 0.93 + 0.487118c_{14}\left[\left(\frac{B}{L_{WL}}\right)^{1.06806}\left(\frac{T}{L_{WL}}\right)^{0.46106}\left(\frac{L_{WL}}{L_R}\right)^{0.121563}\left(\frac{L_{WL}}{V}\right)^{0.36486}(1 - C_p)^{-0.604247}\right] \tag{2.5.19}$$

Appendage Resistance

An appendage is defined as the addition to the main part or main structure of a vessel **(Molland 2011)**. Examples of appendages include rudders, shaft brackets, skeg and bilge keels. The form factors associated with these appendages, denoted as k_{2_i} are presented in Table 2.3. In practice, reasonable estimates can be made based on these form factors, as model tests are not the most suitable method for accurately

Appendage	k_{2_i} value
rudder behind skeg	0.2 – 0.5
rudder behind stern	0.5
twin screw rudder (slender)	1.5
twin screw rudder (thick)	2.5
shaft brackets	2.0 – 4.0
skeg	0.5 – 1.0
strut bossing	2.0 – 3.0
hull bossing	1.0
exposed shafts (angle with buttocks about 10 degrees)	1.0
exposed shafts (angle with buttocks about 20 degrees)	4.0
stabiliser fins	1.8
dome	1.7
bilge keels	0.4

Table 2.3: Approximate values for appendage form factors k_{2_i}

quantifying appendage resistance. Furthermore, the effects of appendages are typically considered as a whole and not as individual units **(Birk 2019)**.

The equivalent form factor for multiple appendages, $(1 + k_{2_i})_{eq}$ is given by:

$$(1 + k_{2_i})_{eq} = \frac{\sum_i (1 + k_{2_i}) S_{APP_i}}{\sum_i S_{APP_i}} \tag{2.5.20}$$

If bow thruster is present, the resistance due to the bow thruster tunnel R_{TH} can be obtained through:

$$R_{TH} = \rho v_S^2 \pi d_{TH}^2 C_{D_{TH}} \quad \textbf{where} \quad C_{D_{TH}} = 0.003 + 0.003\left(\frac{10 d_{TH}}{t} - 1\right) \tag{2.5.21}$$

The coefficient $C_{D_{TH}}$ defines the drag coefficient for the tunnel, and it ranges between 0.003 and 0.012. Smaller values indicate thrusters which are in the cylindrical part of the bulbous bow. The coefficient can also be estimated using the equation by **Hollenbach (1999)** in Equation (2.5.21).

With that, the appendage resistance R_{APP} can be calculated using:

$$R_{APP} = \frac{1}{2} \rho v_S^2 (1 + k_{2_i})_{eq} C_F \sum_i S_{APP_i} + \sum R_{TH} \tag{2.5.22}$$

Wave Resistance

The estimation of wave resistance R_W is dependent on Froude number Fr, and it is subdivided into three categories.[5]:

$$R_W(Fr) = \begin{cases} R_{W_a}(Fr) & \textbf{if} \quad Fr \leqslant 0.4 \\ \text{Interpolation} & \textbf{if} \quad 0.4 < Fr \leqslant 0.55 \\ R_{W_b}(Fr) & \textbf{if} \quad Fr > 0.5 \end{cases} \tag{2.5.23}$$

The wave resistance for $R_{W_a}(Fr)$ can be calculated using:

$$R_{W_a}(Fr) = c_1 c_2 c_5 \rho g V \exp\left[m_1 Fr^d + m_4 \cos(\lambda Fr^{-2}) \right] \tag{2.5.24}$$

And consequently for $R_{W_b}(Fr)$:

$$R_{W_b}(Fr) = c_{17} c_2 c_5 \rho g V \exp\left[m_3 Fr^d + m_4 \cos(\lambda Fr^{-2}) \right] \tag{2.5.25}$$

The Froude number range remaining between $0.4 < Fr \leqslant 0.55$ is determined using interpolation between equations Equation (2.5.24) and Equation (2.5.25). It is important to note that this specific range of Froude numbers (0.4 to 0.55) is generally considered uneconomical for ship operation, and ships do not typically operate within this speed range for extended periods **(Birk 2019)**.

$$R_W(Fr) = R_{W_a}(0.4) + \frac{20Fr - 0.8}{3}\left[R_{W_b}(0.55) - R_{W_a}(0.4) \right] \tag{2.5.26}$$

To compute each of the constants in Equation (2.5.24), The following equations are presented, note that the calculations for the $\cos(\lambda Fr^{-2})$ are in **Radians**:

$$c_7 = \begin{cases} 0.229577\left(\dfrac{B}{L_{WL}}\right)^{\frac{1}{3}} & \textbf{if} \quad \dfrac{B}{L_{WL}} \leqslant 0.11 \\ \dfrac{B}{L_{WL}} & \textbf{if} \quad 0.11 < B/L_{WL} \leqslant 0.25 \\ 0.5 - 0.0625\dfrac{L_{WL}}{B} & \textbf{if} \quad B/L_{WL} > 0.25 \end{cases} \tag{2.5.27}$$

$$c_1 = 2223105 c_7^{3.78613}\left(\frac{T}{B}\right)^{1.07961}(90 - i_e)^{1.37565} \tag{2.5.28}$$

i_E is defined as the half angle of the waterline entrance and the estimation can be calculated by:

$$i_E = 1 + 89e^a \tag{2.5.29}$$

and a can be obtained through:

[5] Considering the length of the equations, only the scenario where $Fr \leqslant 0.4$ will be thoroughly examined in this thesis. The formulations of R_W for other ranges of Froude number can be referenced in the studies by **Holtrop (1984)** and **Birk (2019)**

$$a = -\left[\left(\frac{L_{WL}}{B}\right)^{0.80856}\left(1-C_{WP}\right)^{0.30484}\left[1-C_P-0.0225\ell_{CB}\right]^{0.6367}\left(\frac{L_R}{B}\right)^{0.34574}\left(\frac{100V}{L_{WL}^3}\right)^{0.16302}\right] \tag{2.5.30}$$

$$c_3 = 0.56\frac{A_{BT}}{\left[BT\left(0.31\sqrt{A_{BT}}+T_F+h_B\right)\right]} \tag{2.5.31}$$

$$c_2 = e^{(-1.89\sqrt{c_3})} \tag{2.5.32}$$

$$c_{15} = \begin{cases} -1.69385 & \textbf{if} \quad \frac{L_{WL}^2}{V} \leqslant 512 \\ -1.69385 + \frac{\frac{L_{WL}}{V^{(1/3)}}-8}{2.36} & \textbf{if} \quad 512 < \frac{L_{WL}^2}{V} \leqslant 1726.91 \\ 0 & \textbf{if} \quad \frac{L_{WL}^2}{V} > 1726.91 \end{cases} \tag{2.5.33}$$

$$c_{16} = \begin{cases} 8.07981C_P - 13.8673C_P^2 + 6.984338C_P^3 & \textbf{if} \quad C_P \leqslant 0.8 \\ 1.73014 - 0.7067C_P & \textbf{if} \quad C_P > 0.8 \end{cases} \tag{2.5.34}$$

$$d = -0.9 \tag{2.5.35}$$

$$\lambda = \begin{cases} 1.446C_P - 0.03\frac{L_{WL}}{B} & \textbf{if} \quad \frac{L_{WL}}{B} \leqslant 12 \\ 1.446C_P - 0.36 & \textbf{if} \quad \frac{L_{WL}}{B} > 12 \end{cases} \tag{2.5.36}$$

$$m_1 = 0.0140407C_P - 0.03\frac{L_{WL}}{B} - 1.75254\frac{V^{(1/3)}}{L_{WL}} - 4.79323\frac{B}{L_{WL}} - c16 \tag{2.5.37}$$

$$m_4 = 0.4c_{15}\exp\left(-0.034Fr^{-3.29}\right) \tag{2.5.38}$$

Resistance of bulbous bow

The approximation of the resistance due to bulbous bow R_B can be obtained through the immersion Froude number F_{r_i} for the bulbous bow and the constant P_B which is a measure of the emergence of the bow:

$$Fr_i = \frac{v_S}{\sqrt{g(T_F - h_b - 0.25\sqrt{A_{BT}}) + 0.15v_S^2}} \tag{2.5.39}$$

$$P_B = 0.56\frac{\sqrt{A_{BT}}}{T_F - 1.5h_B + h_F} \tag{2.5.40}$$

$$R_B = 0.11\rho g(\sqrt{A_{BT}})^3 \frac{Fr_i^3}{1+Fr_i^2} e^{(-3.0P_B^{-2})} \tag{2.5.41}$$

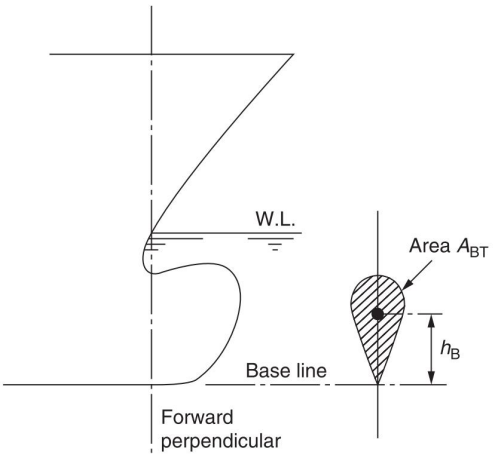

Figure 2.12: Bulbous bow definition **(Molland 2011)**

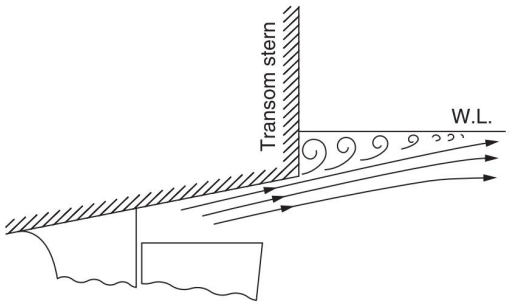

Figure 2.13: Flow around immersed transom stern **(Molland 2011)**

(Immersed) Transom Resistance

The term transom refers to the flat section located at the stern of the ship. When the transom becomes immersed in water, it leads to pressure loss, resulting in resistance. This resistance is denoted by the term R_{TR} and is associated with an immersed transom area $A_T > 0$. The transom resistance can be described as a function of the depth Froude number Fr_T:

$$Fr_T = \frac{v_S}{\sqrt{\frac{2gA_T}{(B+BC_{WP})}}} \tag{2.5.42}$$

The expression $A_T/(B+BC_{WP})$ is a measure for the average draught of the transom. When the average draught is smaller than the speed, there will be a clean separation of the flow at the transom edge and the resistance due to transom vanishes. Immersion resistance R_{TR} is considered if $Fr_T > 5$

$$c_6 = \begin{cases} 0.2(1-0.2Fr_T) & \textbf{if} \quad Fr_T < 5 \\ 0 & \textbf{if} \quad Fr_T > 5 \end{cases} \tag{2.5.43}$$

$$R_{TR} = \frac{1}{2}\rho v_S^2 A_T c_6 \tag{2.5.44}$$

Correlation allowance resistance

The resistance term R_A considers other effects that are not captured by other resistance components.

$$c_4 = \begin{cases} \frac{T_F}{L_{WL}} & \textbf{if} \quad \frac{T_F}{L_{WL}} \leqslant 0.04 \\ 0.04 & \textbf{if} \quad \frac{T_F}{L_{WL}} > 0.04 \end{cases} \tag{2.5.45}$$

The correlation allowance coefficient C_A and subsequent correlation resistance is defined as:

$$C_A = 0.00546(L_{WL} + 100)^{-0.16} - 0.00205 + 0.003\frac{L_{WL}}{7.5}C_B^4 c_2(0.04 - c_4) \tag{2.5.46}$$

$$R_A = \frac{1}{2}\rho v_S^2 C_A(S + \sum S_{APP}) \tag{2.5.47}$$

2.5.2.2 Added resistance due to wind

The magnitude of added resistance caused by wind, R_{AA}, is determined by the area of the ship superstructure and relative wind. Therefore, for a ship with large lateral areas above the water level, this added resistance due to wind can be significant. The estimation of added resistance due to wind in this thesis considers the method by **Blendermann (1994)**:

$$R_{AA} = \frac{\rho_{air}}{2}u^2 A_L C D_l \frac{\cos(\varepsilon)}{1 - \frac{\delta}{2}(1 - \frac{CD_l}{CD_t}\sin^2(2\varepsilon))} \tag{2.5.48}$$

Where u is the apparent wind velocity, A_L, the lateral plane area, ε, the apparent wind angle ($\varepsilon = 0$ in headwind), δ the cross-force parameter, and coefficients CD_t and CD_l the non-dimensional drag in beam wind and headwind. For given true wind velocity u_{TW} and true wind angle (TWA), β, The calculation for the apparent wind u and apparent wind angle ϵ is performed using the following equations:

$$u = \sqrt{u_{\mathrm{TW}}^2 + v_S^2 + 2 \cdot u_{\mathrm{TW}} \cdot v_S \cdot \cos(\beta)} \tag{2.5.49}$$

$$\frac{u_{TW}}{\sin(\varepsilon)} = \frac{u}{\sin(\beta)} \tag{2.5.50}$$

According to **Schneekluth and Bertram (1998)**, maximum wind resistance is encountered when $0° < \varepsilon < 20°$ and it is more convenient to express the longitudinal drag with respect to the frontal area A_F. Typical values for the constants are summarised in Table 2.4

$$CD_{lAF} = CD_l \frac{A_L}{A_F} \tag{2.5.51}$$

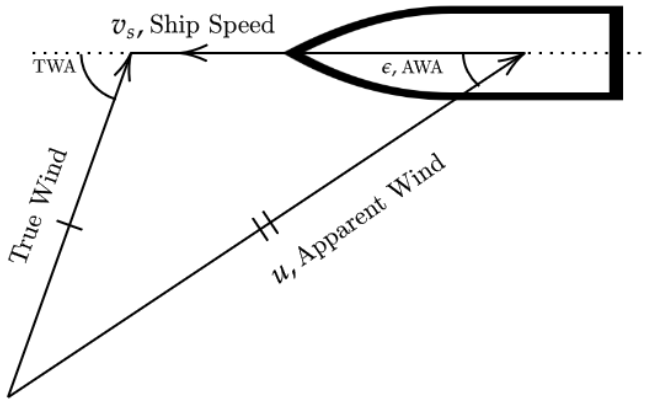

Figure 2.14: Apparent and true wind **(Knudsen 2013)**

	CD_t	CD_{lAF}	δ
Car carrier	0.95	0.55	0.8
Cargo ship, container on deck, bridge aft	0.85	0.65/0.55	0.40
Containership, loaded	0.90	0.55	0.40
Ferry	0.90	0.45	0.80
LNG Tanker	0.70	0.60	0.50
Passenger liner	0.90	0.40	0.80
Speed boat	0.90	0.55	0.60
Tanker, loaded	0.70	0.90	0.40
Tanker, in ballast	0.70	0.75	0.40

Table 2.4: Coefficients to estimate wind resistance

2.5.2.3 Added resistance due to wave

The added resistance due to wave, R_{AW} is estimated using the STAWAVE-1 method recommended by **ITTC (2014)**. This method only considers waves encountered within the bow sector i.e. within ±45° off the bow and does not consider wave correction for other encounters. Also, STAWAVE-1 is valid for the following condition:

$$\text{Significant wave height:} \quad H_{1/3} = 2.25 \leqslant \sqrt{L_{PP}/100} \tag{2.5.52}$$

$$R_{AWL} = \frac{1}{16}\rho g H_{1/3}^2 B \sqrt{\frac{B}{L_{BWL}}} \tag{2.5.53}$$

In which, L_{BWL} is the length of the bow on the water line to 95% of maximum breadth.

2.5.2.4 Efficiencies affecting brake power

Open water efficiency

The open water efficiency η_O, can be understood as the propeller working in open water conditions i.e. the propeller operates in a homogenous wake field with no hull in front of it. The curve of different propulsion devices with their respective efficiencies is summarised in the work of **Breslin and Andersen (1994)**:

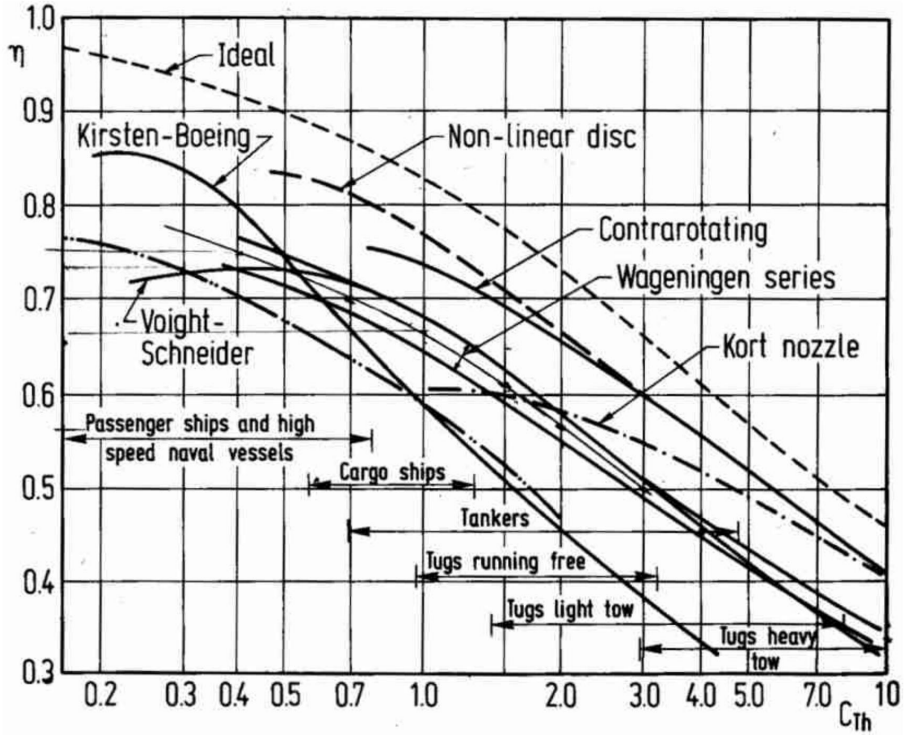

Figure 2.15: Efficiencies of various propulsion devices **(Breslin and Andersen 1994)**

Hull efficiency

The Hull efficiency η_H can be calculated using the following equation:

$$\eta_H = \frac{1-t}{1-w_S} \tag{2.5.54}$$

The term t refers to the thrust deduction fraction, which represents the thrust force required to overcome the towing resistance of the ship R_{TOTAL} and the additional resistance caused by the propeller's interaction with the hull. On the other hand, the term w_S corresponds to the wake fraction, characterising the influence of the ship's hull on the water flow into the propeller **(MAN 2011; Birk 2019)**. The following equations are presented for twin-screw vessels[6] to calculate w_S and t.

$$w_S = 0.3095C_B + 10C_V C_B - 0.23\frac{D}{\sqrt{BT}} \tag{2.5.55}$$

[6] Considering the length of the equations, the equations for single screw vessels can be obtained from **Holtrop and Mennen (1982)** and **Birk (2019)**

$$t = 0.325C_B - 0.1885\frac{D}{\sqrt{BT}} \tag{2.5.56}$$

where C_V is the viscous resistance coefficient, which combines all friction-related components of the resistance and the correlation resistance:

$$C_V = \frac{(1+k_1)R_F + R_{APP} + R_A}{\frac{1}{2}\rho v_S^2 (S + \sum_i S_{APP_i})} \tag{2.5.57}$$

Relative rotative efficiency

The relative rotative efficiency η_R can be expressed by the following ratio, with v_A defined as the arriving water velocity to propeller **(MAN 2011)**:

$$\eta_R = \frac{\text{Power absorbed in open water at } v_A}{\text{Power absorbed in wake behind the ship at } v_A} \tag{2.5.58}$$

According to **Holtrop and Mennen (1982)**, η_R for twin screw vessels can be estimated using the following formula, with P/D defined as the propeller pitch-to-diameter ratio:

$$\eta_R = 0.9737 + 0.111(C_P - 0.0225\ell_{CB}) - 0.06325\frac{P}{D} \tag{2.5.59}$$

Shaft efficiency

The shaft efficiency η_S is defined as the ratio between the power delivered to the propeller P_D and the brake power of the main engine P_B, with values ranging from $\eta_S = 0.95 - 0.99$ depending on shaft design and gear configuration.

Chapter 3

Research Methodology

This chapter will cover the methodology employed to develop the grey box model. The grey box modelling approach adopted in this thesis falls within the category of sequential GBM. Consequently, the development process is divided into two stages. The initial stage of the modelling process focuses on machine learning, specifically the development of the BBM using tree-based models. This is carried out using Python in conjunction with `Scikit-Learn` **(Pedregosa et al. 2011)**. This stage encompasses data acquisition, data preprocessing, hyperparameter optimization, and model evaluation. The training of the models involves a fusion of T-AIS data and weather data, followed by the selection of relevant features for predicting the SOG. These processes are visually illustrated in Figure 3.1.

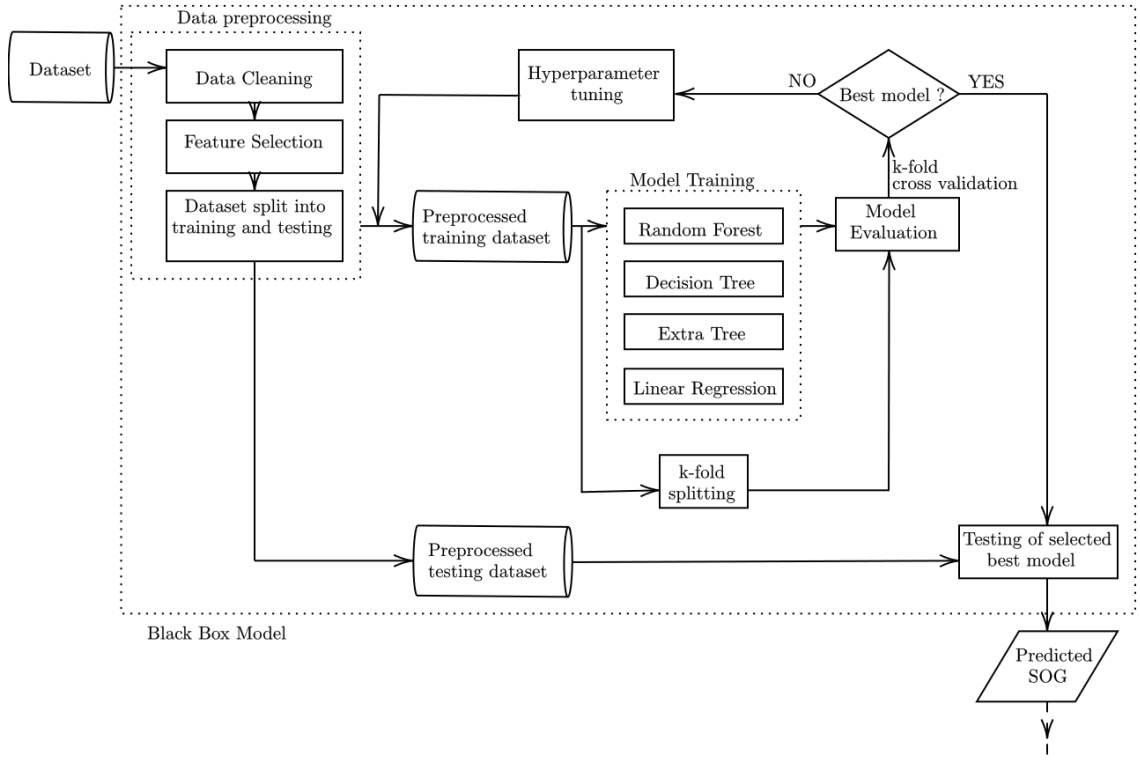

Figure 3.1: Scheme of proposed BBM methodology

The second stage of the modelling process is centred around the WBM component of the GBM. In this phase, the predicted SOG serves as input for the WBM to forecast the necessary brake power required for propelling the ship. This involves an initial conversion of SOG to STW for estimation of encountered resistance during the voyage, this then facilitates the estimation of the required power i.e. energy required to propel the ship. The framework outlined in Figure 3.2 provides a graphical depiction summarising the concepts discussed in Section 2.5.

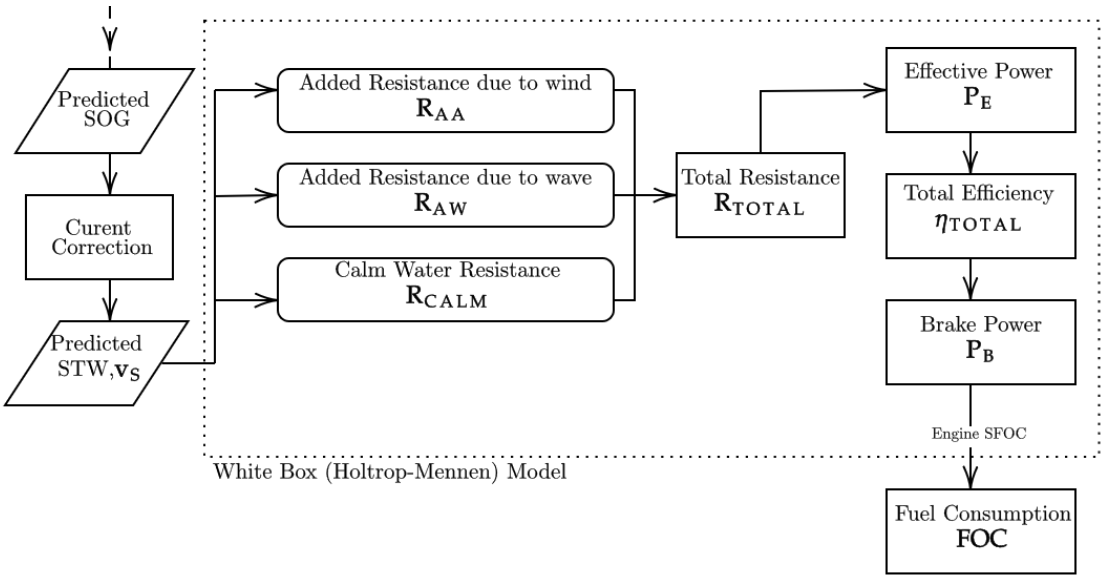

Figure 3.2: Scheme of proposed WBM methodology adopted from **Lang (2020)**

The development process of GBM is summarised in Figure 3.3. Detailed discussion regarding the development of BBM and WBM model will be discussed in the following sections of this chapter.

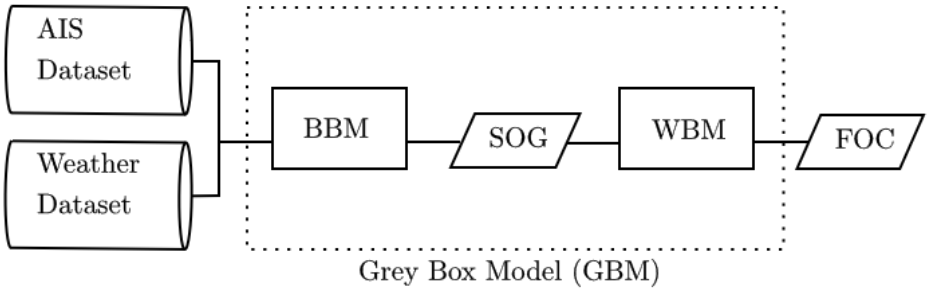

Figure 3.3: Scheme of proposed GBM methodology

3.1 Data Acquisition

The data is collected from a ferry serving between ports of Køge, Rønne, Ystad and Sassnitz, as shown in Figure 3.5 **(Commons 2010)**. The trip between Køge, Rønne takes about 5 h 30 minutes and it sails between Rønne and Sassnitz for 3 h and 20 minutes. The journey is tracked by the T-AIS system of the Danish Maritime Authority (DMA). The weather data along her sailing path are acquired from ECMWF[1] with a temporal resolution of 1 hour at a spatial granularity of 0.25° (longitude) x 0.25° (latitude), data from ECMWF provides information for wind, waves and seawater temperature. The information for current is obtained from CMEMS[2] with a temporal resolution of 3 hours at a spatial granularity of 0.25° (longitude) x 0.25° (latitude).

IMO	9812107
Type & Service	Passenger ferry
L_{OA}	158.00 m
L_{WL}	144.80 m
B (moulded)	24.5 m
T_{DESIGN}	5.70 m
T_{MAX}	5.85 m
Gross Tonnage (GT)	18,009
Deadweight (dwt)	4,830 t
Main Engines	Wärtsillä 8V31 2 x 4,880 kW
SFOC	169.4 g/kWh
Service Speed	17.7 knots
Bow Thrusters	2 x 1500 kW

Figure 3.4: Particular of M/S Hammershus

Figure 3.5: Journey of the ferry

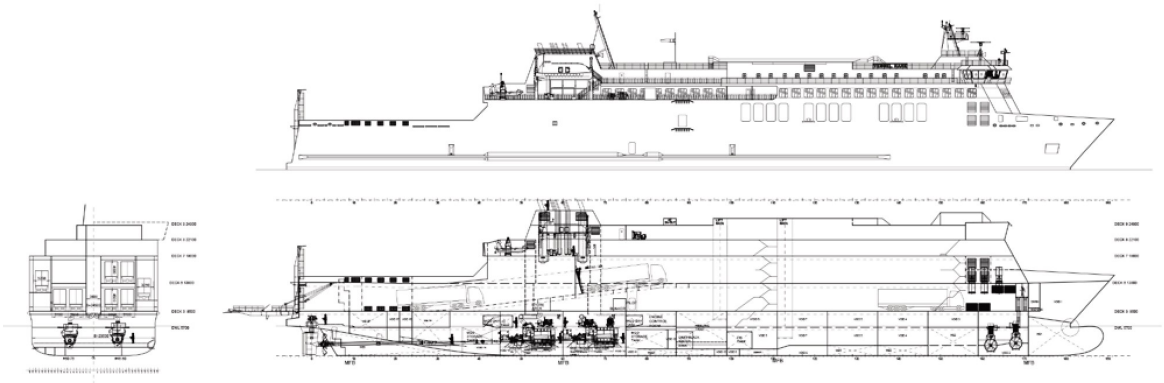

Figure 3.6: Schematics of M/S Hammershus

The resulting combined dataset maintains a temporal resolution of 1 hour. To ad-

[1] European Centre for Medium-Range Weather Forecast
[2] Copernicus Marine Environment Monitoring Service

dress the disparity between the temporal resolutions of the data from CMEMS and ECMWF, the weather information is synchronised. This synchronisation ensures that the wind, waves, seawater temperature, and sea current data align with the same weather grid and possess consistent temporal resolutions. The features **wind direction**,**swell direction**, and **wind wave direction** are oriented to true north. However, to reflect the actual direction of weather effects that are acting on the ship, these features are converted to true direction; where true direction is defined as the direction of weather effect with respect to the bow of the ship. The value ranges between 0° and 180°. Subsequently, through vector decomposition, the northward and eastward wind velocity is converted to absolute wind speed and wind direction *with respect to True North*,φ:

$$u_{\mathrm{W}} = \sqrt{(u_{\mathrm{W}_N})^2 + (u_{\mathrm{W}_E})^2} \tag{3.1.1}$$

$$\varphi = \begin{cases} 360 - \arctan(u_{\mathrm{W}_E}/u_{\mathrm{W}_N}) & \textbf{if} \quad u_{\mathrm{W}_E} > 0 \quad \wedge \quad u_{\mathrm{W}_N} < 0 \\ 180 - \arctan(u_{\mathrm{W}_E}/u_{\mathrm{W}_N}) & \textbf{if} \quad u_{\mathrm{W}_E} < 0 \quad \wedge \quad u_{\mathrm{W}_N} > 0 \\ 270 - \arctan(u_{\mathrm{W}_E}/u_{\mathrm{W}_N}) & \textbf{if} \quad u_{\mathrm{W}_E} > 0 \quad \wedge \quad u_{\mathrm{W}_N} > 0 \\ \arctan(u_{\mathrm{W}_E}/u_{\mathrm{W}_N}) & \textbf{otherwise} \end{cases} \tag{3.1.2}$$

Similarly, information of Northward and Eastward current Velocity is converted to the absolute current speed and current direction *with respect to True North* γ.

$$v_{\mathrm{C}} = \sqrt{(v_{\mathrm{C}_N})^2 + (v_{\mathrm{C}_E})^2} \tag{3.1.3}$$

$$\gamma = \begin{cases} 360 - \arctan(u_{\mathrm{W}_E}/u_{\mathrm{W}_N}) & \textbf{if} \quad v_{\mathrm{C}_E} < 0 \quad \wedge \quad v_{\mathrm{C}_N} > 0 \\ 180 - \arctan(u_{\mathrm{W}_E}/u_{\mathrm{W}_N}) & \textbf{if} \quad v_{\mathrm{C}_E} > 0 \quad \wedge \quad v_{\mathrm{C}_N} < 0 \\ 270 - \arctan(u_{\mathrm{W}_E}/u_{\mathrm{W}_N}) & \textbf{if} \quad v_{\mathrm{C}_E} < 0 \quad \wedge \quad v_{\mathrm{C}_N} < 0 \\ \arctan(u_{\mathrm{W}_E}/u_{\mathrm{W}_N}) & \textbf{otherwise} \end{cases} \tag{3.1.4}$$

The initial dataset offers data in true directions, indicating the direction in which weather conditions are encountered relative to the ship's bow. The static information from AIS data, which includes the ship's identity and navigational status, is excluded from the dataset. The original structure encompasses 27 features: 9 AIS features and 18 weather features. The layout of the initial dataset, prior to data preprocessing and feature selection, is outlined in Table Table 3.1.

3.2 Data Preprocessing

This section presents the steps taken during data preprocessing. The dataset will be subjected to data cleaning which includes identification of anomalies and missing values. A threshold for SOG is applied to ensure the model captures operating conditions in a steady state. Feature selection based on domain knowledge is executed to align with vessel domain knowledge. Subsequently, the datasets are partitioned into training and testing subsets.

Feature	Feature Name
AIS data	
Position Time Stamp [DD/MM/YYYY HH:MM:SS]	`Time`
Latitude [°]	`LAT`
Longitude [°]	`LON`
Width [m]	`width`
Length [m]	`length`
SOG [Knots]	`sog`
COG [m/s]	`cog`
Heading [°]	`heading`
Draught [m]	`draught`
Weather Data (0.5° Granularity)	
Wind Speed [m/s]	`windspeed`
True North Wind Direction, φ [°]	`truenorthcurrentdir`
Air Temperature Above Oceans [K]	`oceantemperature`
Maximum Wave Height [m]	`waveheight`
Swell Period [s]	`swellperiod`
Wind Wave Period [s]	`windwaveperiod`
Wave Period [s]	`waveperiod`
Sea Surface Temperature [K]	`surftemp`
Combined Wind Wave Swell Height [m]	`windwaveswellheight`
Swell Height [m]	`swellheight`
Wind Wave Height [m]	`windwaveheight`
Current Speed [m/s]	`curspeed`
True North Current Direction γ [°]	`truenorthcurrentdir`
True Wind Direction [°]	`truewinddir`
True Current Direction [°]	`truecurrentdir`
True Swell Direction [°]	`trueswelldir`
True Wind Wave Direction [°]	`truewindwavedir`
True Wave Direction [°]	`truewavedir`

Table 3.1: Structure of fused dataset

3.2.1 Data Cleaning

The plotted trajectory reveals an incomplete representation of the voyage between Rønne and Sassnitz. This may stem from the constraints of the T-AIS system, attributed to limited coverage within the area connecting Sassnitz and Rønne. This is shown by the plot shown in Figure 3.7. Therefore, the data plot for the journey between Sassnitz and Rønne will be excluded. Furthermore, a latitude threshold of 55.04° N is applied, which excludes the journey segment between Sassnitz and Rønne.

In its initial state, the dataset contains 7453 data points which described the journey

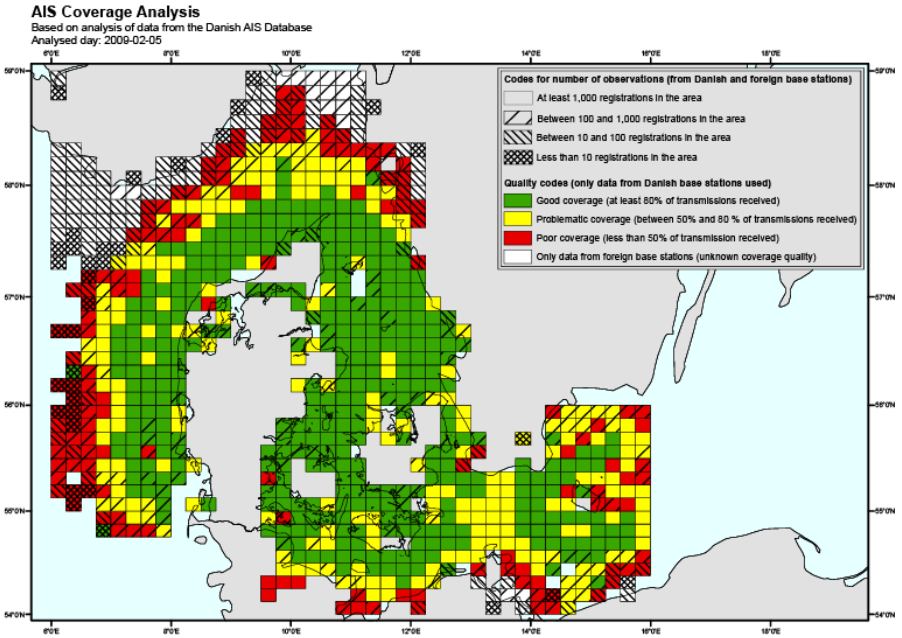

Figure 3.7: Shore based AIS Coverage based on data from AIS database Danish Maritime Authority (2023)

of the ship in one year. The initial data points represented all navigational statuses of the ship, which include "mooring", "anchoring" and "underway using engine". This is observed in the histogram for the SOG Figure 3.8.

To ensure the dataset accurately reflects the ship's operational status under steady conditions, a threshold for SOG is implemented. While variations in SOG can arise from changing sea conditions, they can also result from deliberate speed reduction during port departures or arrivals. Thus, any data points with SOG below 5 knots, which is indicative of manoeuvring, are removed **(Abebe et al. 2020; Yan, Wang, and Du 2020)**. Following this filtering step, the dataset size notably decreases from 7453 data points to 3828 data points. This reduction shows that approximately half of the original data points correspond to the ship's stationary activities.

Preliminary analysis reveals a potential source of error in data points representing current speed. Within the range of current speeds between 0.01 and 0.03 m/s, a distinct peak in data points is evident, as depicted in Figure Figure 3.8. This peak can be attributed to incomplete information regarding northward and eastward current speed in certain data points within the provided dataset. Consequently, a single random error value for the current speed is generated, leading to the observed peak in the histogram.

To address the presence of missing values, the `KNNImputer` feature from `Scikit-Learn` is utilized to impute the missing values for eastward current and northward current. This step is essential as the modelling package provided by `Scikit-Learn` cannot handle instances with missing values. During the imputation process, the missing values for each sample are filled in using the mean value of the nearest neighbours

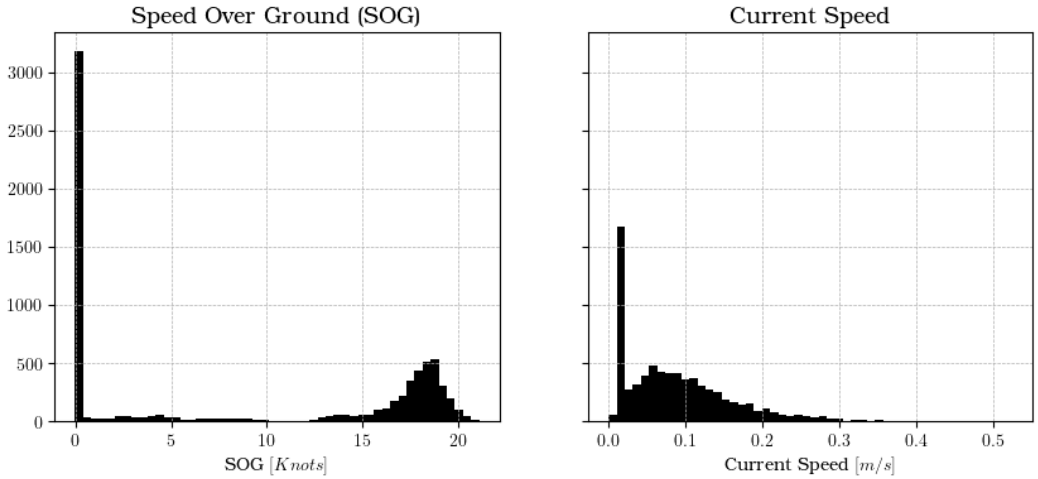

Figure 3.8: Histogram plot of pre-filtered SOG and current speed

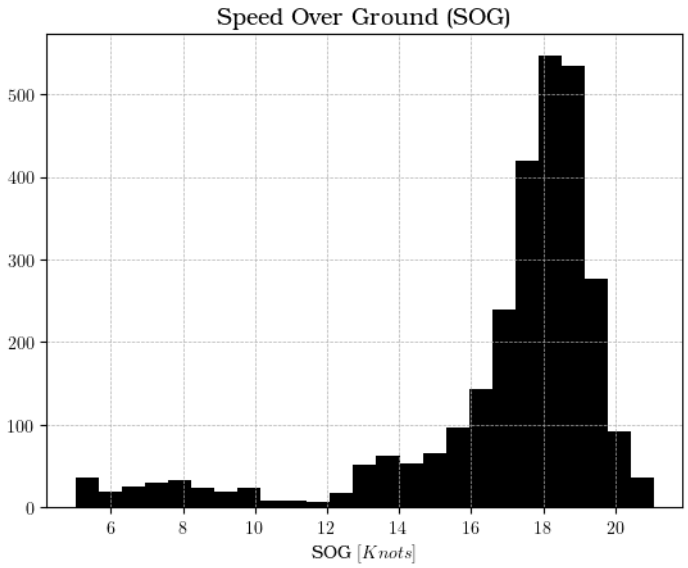

Figure 3.9: Histogram plot of SOG after threshold

found within the training dataset **(Pedregosa et al. 2011)**. The choice of using a k-nearest neighbour imputation strategy is appropriate, as it aims to capture the weather conditions within the vicinity of the missing values. Once the missing values for the northward and southward currents have been imputed, the current speed for these instances will be re-calculated. This k-nearest neighbour imputation approach is also extended to other weather features that contain missing values, specifically the `NaN` values.

3.2.2 Feature Selection

In order to choose suitable features for the model, the correlation among different features is studied. Feature selection is essential to simplify the model and consequently reduce computational expenses during the training phase. The process of

Training Label	
SOG [Knots]	`sog`
Training Features	
COG [°]	`cog`
Heading [°]	`heading`
Draught [m]	`draught`
Wind Speed [m/s]	`windspeed`
Air Temperature Above Oceans [K]	`oceantemperature`
Wave Period [s]	`waveperiod`
Sea Surface Temperature [K]	`surftemp`
Combined Wind Wave Swell Height [m]	`windwaveswellheight`
Current Speed [m/s]	`curspeed`
True Wind Direction [°]	`truewinddir`
True Current Direction [°]	`truecurrentdir`
True Wave Direction [°]	`truewavedir`

Table 3.2: Structure of training dataset

feature selection follows a statistical technique known as the High Correlation Filter, as proposed by **Abebe et al. (2020)**. This method treats pairs of features with correlation coefficients exceeding 0.7 as a single entity. Nonetheless, the process of selecting highly correlated features must align with established physical principles. Hence, feature selection in this study is predominantly guided by physical reasoning, and this principle takes precedence over purely statistical considerations.

From AIS data, the information on *time, latitude, longitude, width and length* are excluded, considering that time, latitude and longitude only describe the location of the ship at a particular position and the width and length of the ship are constant dimensions. As elaborated in Section 2.4.1, certain features like *combined wind wave swell height, swell height, maximum wave height,* and *wind wave height* are interconnected by physical relationships. The combined wind wave swell height corresponds to the significant wave height $H_{1/3}$, which is mathematically described by Equation (2.4.4). Furthermore, Equation (2.4.6) illustrates how the significant wave height serves to identify whether the sea is dominated by swell or wind-generated waves.

Hence, it is evident that retaining the significant wave height is essential for the model, given that various wave properties can be deduced from it. Features like swell height, wind wave height, and maximum wave height will be excluded, as they can be determined from the significant wave height $H_{1/3}$. This choice is also substantiated by statistical analysis using the high correlation filter method. As depicted in Figure 3.10, high correlations exist among $H_{1/3}$, swell height, wind wave height, and maximum wave height.

Figure 3.10 illustrates a significant correlation between wave period, swell period,

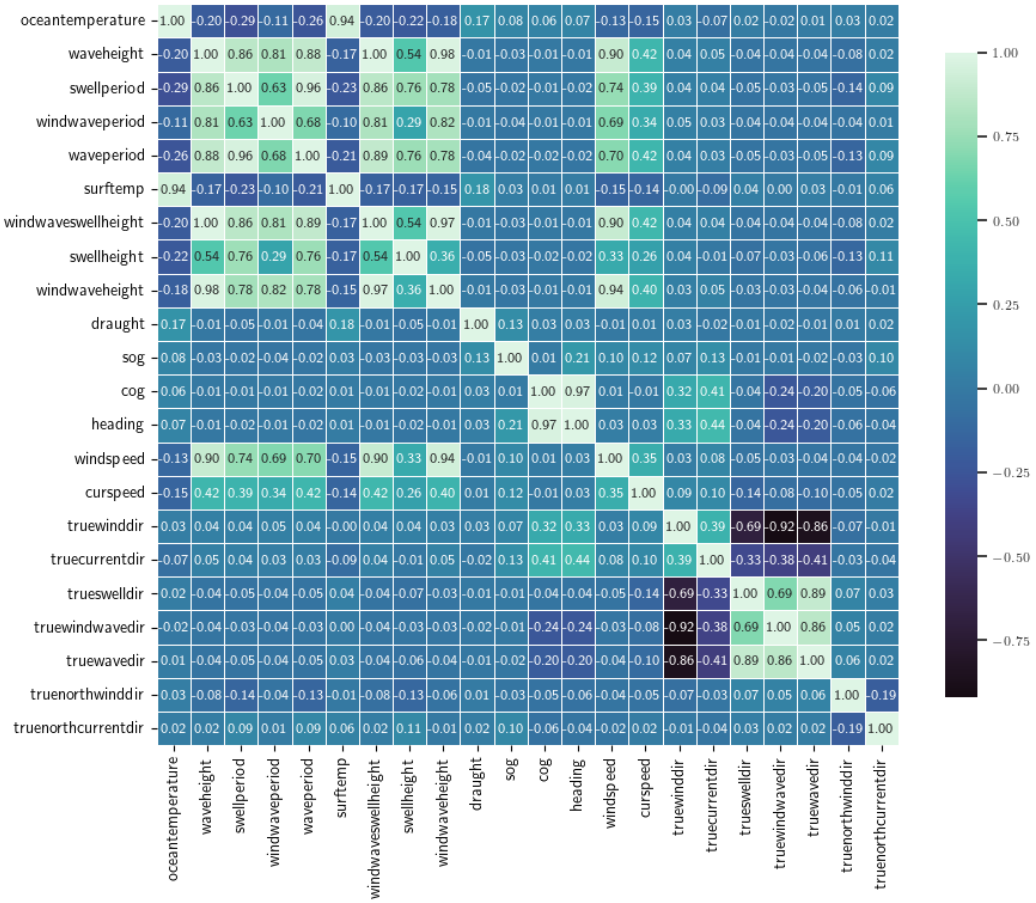

Figure 3.10: Correlation Heat Map

and wind wave period. As outlined in Section 2.4.1, the sea state's description involves the significant height $H_{1/3}$ and spectral peak T_p through the Torsethaugen peak method **(Torsethaugen and Haver 2004)**. Consequently, the features swell period and wind wave period will be omitted, as they primarily differentiate whether the sea is influenced by swell or wind. However, the wave period feature will be retained. Therefore, the attributes "true wind wave direction" and "true swell direction" will also be disregarded since the features accounting for their magnitude have been removed.

In a statistical sense, heading and COG are significantly correlated, yet both features are retained due to their representation of distinct ship parameters. Course Over Ground (COG) signifies the ship's course heading while heading signifies the ship's actual heading at a specific time point. A similar rationale applies to the relationship between air temperature above the ocean and sea surface temperature. Air temperature above oceans represents wind temperature, whereas sea surface temperature reflects the temperature of the water surface. Following this principle, 5 features from the AIS data and 11 features from the weather data are omitted through feature selection. For predicting ship speed, SOG is selected as the target label for model training. The remaining attributes are chosen as training features. This is summarised in Table 3.2.

Features	Count	Mean	Std.	Min	25%	50%	75%	Max
sog	2871	16.91	3.18	5.03	16.56	17.94	18.72	21.07
cog	2871	196.47	85.93	69.77	102.58	188.01	282.26	355.07
heading	2871	187.88	88.47	67.90	100.86	124.65	279.19	355.07
draught	2871	5.22	0.18	4.74	5.11	5.28	5.38	5.67
windspeed	2871	6.42	2.97	0.25	4.13	6.15	8.36	16.01
oceantemperature[3]	2871	282.71	6.49	264.08	277.13	282.64	288.82	296.83
waveperiod	2871	3.66	0.82	1.86	3.07	3.57	4.14	7.05
surftemp[4]	2871	283.40	5.73	273.05	278.13	282.83	288.86	294.75
windwaveswellheight[5]	2871	0.75	0.51	0.07	0.38	0.64	0.95	3.70
curspeed	2871	0.10	0.07	0.00	0.05	0.08	0.13	0.53
truewinddir	2871	87.14	55.96	0.00	34.19	84.79	140.58	179.77
truecurrentdir	2871	89.15	57.53	0.25	31.01	86.78	143.32	179.99
truewavedir	2871	91.74	55.53	0.13	39.12	92.28	143.33	179.92

Table 3.3: Descriptive statistics of preprocessed dataset

3.3 Black Box Modelling

In this section, the modelling of ship speed based on SOG using the selected features will be conducted employing a tree-based regressor model. The considered tree-based regressor models comprise the decision tree regressor (DTR), random forest regressor (RFR), and extra-tree regressor (ETR). Additionally, to establish a benchmark, the tree-based models are compared against multiple linear regressors (MLR). The dataset is divided into training and test datasets with a ratio of 75:25 for training and testing, respectively. This results in 2871 data points for training and 957 data points for testing. To facilitate robust evaluation, the training dataset is subjected to 10-fold splitting. The hyperparameters of the tree-based regressors will be systematically fine-tuned through iterative processes until no further improvement to the model's performance can be achieved.

3.3.1 Performance Metrics for Validation

To obtain a meaningful assessment of the model's performance and its precision, a cross-validation technique known as k-folding will be employed. K-fold cross-validation involves dividing the training set into k subsets, referred to as folds. Subsequently, the model will undergo k training iterations, using k-1 subsets for training and the remaining subset for validation. This process is visually depicted in Figure 3.12. During each iteration, the model's performance will be evaluated using

[3] Air temperature above oceans
[4] Sea Surface Temperature
[5] Significant wave height

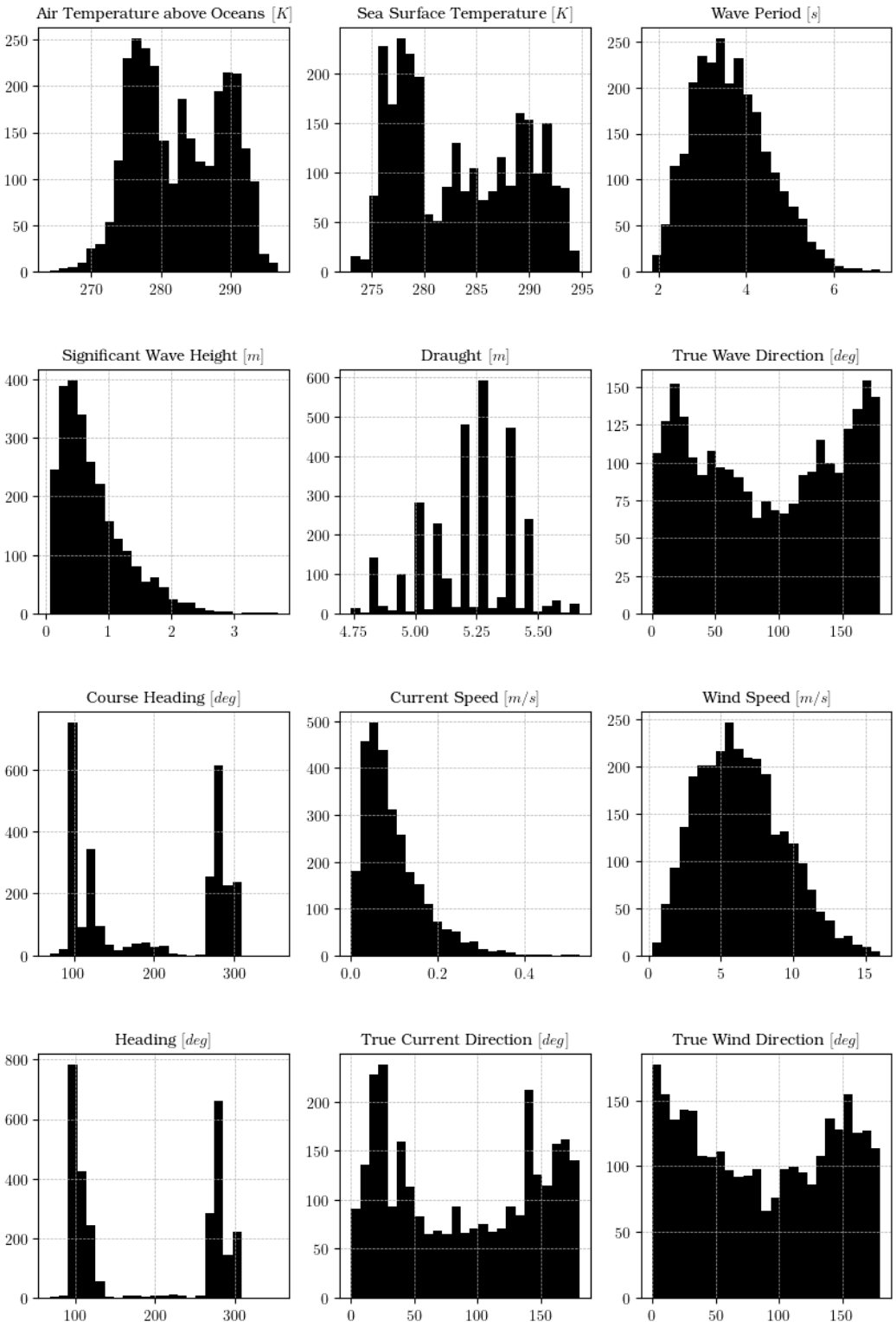

Figure 3.11: Histogram of training features

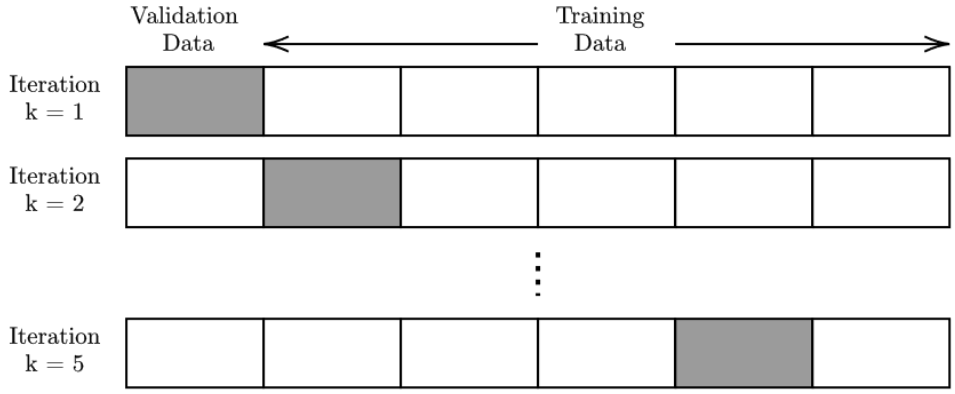

Figure 3.12: Visual illustration of k-folding, Grey shaded box represents the validation data while white box represents the training data

various metrics, including the **Coefficient of Determination (R^2), Explained Variance (EV), Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Median Absolute Deviation (MAD), and Mean Absolute Percentage Error (MAPE)**. The outcomes from each iteration will be averaged, enabling an assessment of the model's precision, which can be further understood through the standard deviation. The application of k-fold cross-validation aids in evaluating the model's robustness across different datasets. The characteristics of each performance metric will be discussed in the subsequent sections.

Coefficient of Determination (R^2)

The coefficient of determination R^2 gives a measure on prediction quality, R^2 quantifies the ability of the regression model to approximate the actual values. R^2 is defined by Equation (3.3.1), where y represents true target output, $\hat{y}$ represents the predictor output and $\overline{y}$ represents the mean. R^2 score range between 0 and 1, higher values i.e. $R^2 \to 1$ indicate better model fit and a score of 1 indicates perfect prediction.

$$R^2(y,\hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_\mathrm{i} - \hat{y}_\mathrm{i})^2}{\sum_{i=1}^{n}(y_\mathrm{i} - \overline{y}_\mathrm{i})^2} \quad \textbf{where} \quad \overline{y} = \frac{1}{n}\sum_{1}^{n} y_\mathrm{i} \tag{3.3.1}$$

Explained Variance (EV)

Explained variance indicates how well a model can capture variance from a dataset. It is defined by Equation (3.3.2), where σ_x represents the standard deviation of parameter x. EV score range between 0 and 1, where the best score of $EV = 1$ can be obtained if $\sigma^2_{(y-\hat{y})} \to 0$.

$$EV(y,\hat{y}) = 1 - \frac{\sigma^2_{(y-\hat{y})}}{\sigma^2_y} \tag{3.3.2}$$

Mean Absolute Error (MAE)

MAE indicated the expected value of absolute (L^1 norm) error, and it can be calculated by:

$$MAE(y,\hat{y}) = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3.3.3}$$

Root Mean Square Error (RMSE)

The RMSE describe the expected value of quadratic error. RMSE place a large penalty on large deviations between true and estimated values and for this reason, it can be used as a metric to indicate model performance against outliers. The ideal score is observed when RMSE $\rightarrow 0$. RMSE can be considered an absolute measure of model fitness. Omitting the root term, RMSE becomes MSE, which is the loss function of Equation (2.2.2) that is used to determine the most optimal split in a regression decision tree.

$$RMSE(y,\hat{y}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3.3.4}$$

Median Absolute Deviation (MAD)

MAD is a performance metric that considers the median of the absolute errors. It is robust to outlier as it only considers median performance

$$MAD(y,\hat{y}) = \text{median}(|y_1 - \hat{y}_1|,\ldots,|y_i - \hat{y}_i|) \tag{3.3.5}$$

Mean Absolute Percentage Error (MAPE)

Is an alternative to MAE, which provide easier interpretation, the result of MAPE can be interpreted according to Equation (3.3.6) **(Montaño Moreno et al. 2013)**. The usage of MAPE in model evaluation is to get an initial estimate, as MAPE comes with some drawbacks such as instability when $y_i = 0$ and it may lead to biased forecast **(Gkerekos, Lazakis, and Theotokatos 2019)**. As such, the evaluation of the model performance will be mainly based on MAE and RMSE.

$$MAPE(y,\hat{y}) = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \cdot 100\% \quad \textbf{with}$$

MAPE	Interpretation
< 10	Highly accurate forecasting
$10 - 20$	Good Forecasting
$20 - 50$	Reasonable forecasting
> 50	Inaccurate forecasting

(3.3.6)

3.3.2 Model Hyperparameter Optimisation

The subject of parameter tuning was briefly discussed in Section 2.2.1. In Section 2.2.1 parameter tuning was applied to the decision tree regressor to avoid overfitting by changing the minimum amount of samples a leaf node has. This example implies that altering the model hyperparameter will affect the model performance. However, the optimisation of the hyperparameter cannot be performed *a priori* and as such iterative process will be performed until the best hyperparameter value is found.

Model	Decision Tree	Random Forest	Extra-Trees
Number of trees	1	Many	Many
Features considered for split at each node	All features	Random subset of features	Random subset of features
Bootstrapping	Not applied	Yes	No
Split Rule	Best split	Best split	Random split

Table 3.4: Comparison of tree-based model from Section 2.2

`Scikit-Learn` offers `GridSearchCV` and `RandomizedSearchCV` to help search for the most optimal hyperparameter. Both solutions operate with similar principle: The selected hyperparameters to be tuned with their value range are evaluated using cross-validation to evaluate the best possible combination between the selected hyperparameters. The difference between `GridSearchCV` and `RandomizedSearchCV` lies in how it searches for the best value for the selected hyperparameters: `GridSearchCV` involves the construction of grids containing all possible combinations of hyperparameter value in a specified range.`RandomizedSearchCV` randomly samples hyperparameter values. The exhaustive nature of `GridSearchCV` means that it is computationally costly to perform, especially when there are multiple hyperparameters to be considered and the value search space is large. `RandomizedSearchCV` gives more control to computing budget by setting the number of iterations and usually produces more accurate results than `GridSearchCV` approach. **(Géron 2019; Bergstra and Bengio 2012)**.

To address this issue, the `RandomizedSearchCV` technique will be applied to identify the optimal hyperparameters. However, it's important to note that a lack of *a priori* knowledge regarding hyperparameter values remains a challenge. Despite the ability of `RandomizedSearchCV` to manage computational resources, obtaining the best

hyperparameter values can still demand a significant amount of time. This could potentially lead to spending computational resources on searches within unpromising search spaces. Therefore, an initial investigation into the impact of each hyperparameter on model performance will be conducted. This preliminary exploration aims to provide a clearer understanding of which search spaces should be considered during the subsequent hyperparameter optimization process. In the forthcoming subsections, the influence of tunable hyperparameters within the tree-based models from `Scikit-Learn` will be examined. This exploration seeks to establish baseline values for the search space. Throughout this analysis, MAE will serve as the chosen performance metric. The hyperparameter optimization pursued within this thesis is aimed towards minimising prediction errors.

Number of features

Defined with default value as `max_features=None` in `Scikit-Learn`. This hyperparameter controls the number of features to be considered when looking for the best split, the default `None` option means it will consider all features. This parameter tuning is available for Decision Tree Regressor, Random Forest Regressor and Extra-Tree Regressor. Initial exploration indicated Random Forest Regressor and Extra Tree Regressor benefit from considering more features. Decision Tree Regressor also benefits from considering all features for determining the split.

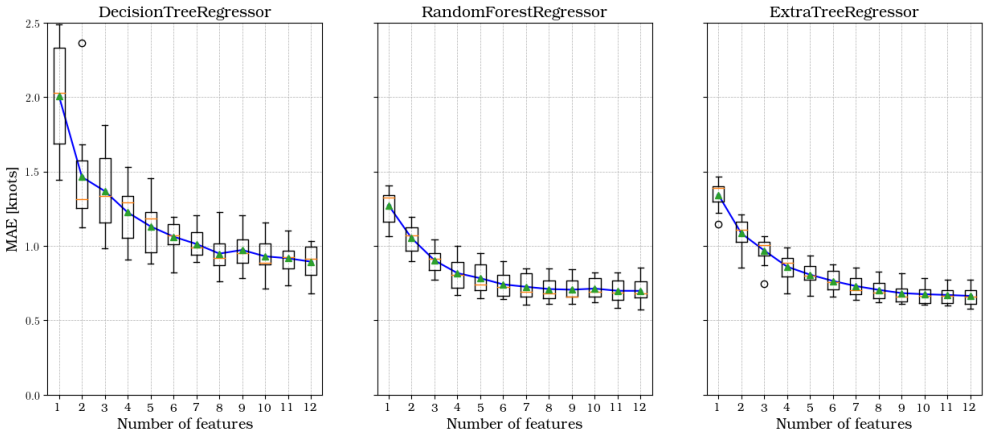

Figure 3.13: Hyperparameter tuning of `max_features`

Minimum samples to split a node

Defined with default value as `min_samples_split=2` in `Scikit-Learn`. This hyperparameter controls the minimum number of samples i.e. data points required to split a node. The default value of 2 is the least number of samples required to split a node i.e. 1 sample is split to the left and right branches respectively. The plot at Figure 3.15 indicates that tuning this hyperparameter will not have any major impact on any model's performance.

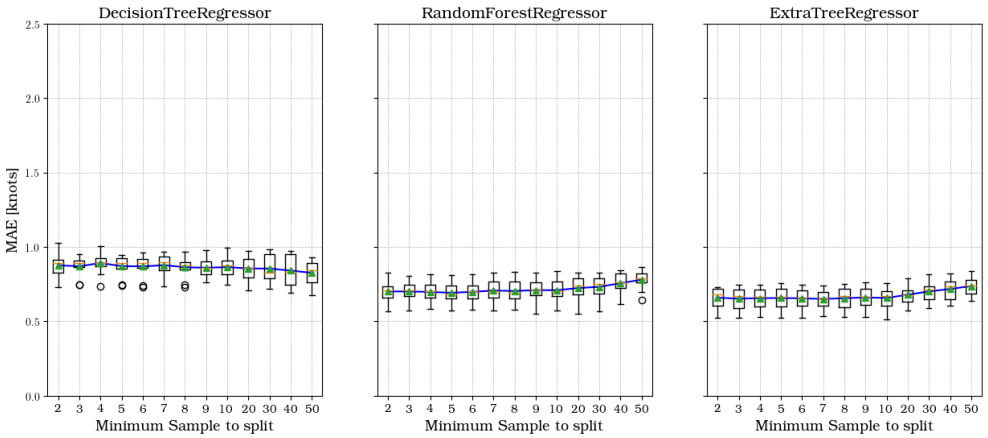

Figure 3.14: Hyperparameter tuning of `min_samples_split`

Number of sample in a leaf node

Defined with default value as `min_samples_leaf=1` in `Scikit-Learn`. This parameter controls the number of samples required to be at the leaf node, where the split point will be considered if the leaf contains at least `min_samples_leaf=n` training samples in each left and right branch. As shown in Figure 2.4, tuning this hyperparameter to higher values helps to smoothen the model and avoid overfitting. However, this may lead to underfitting as the model is unable to capture the trend within the data. This is supported by the findings shown in Figure 3.15, the DTR benefits from regularisation at a certain breakeven point. But, after this breakeven point, the model's performance degrades. It is also observed that tuning this parameter will negatively impact RFR and ETR model's performance.

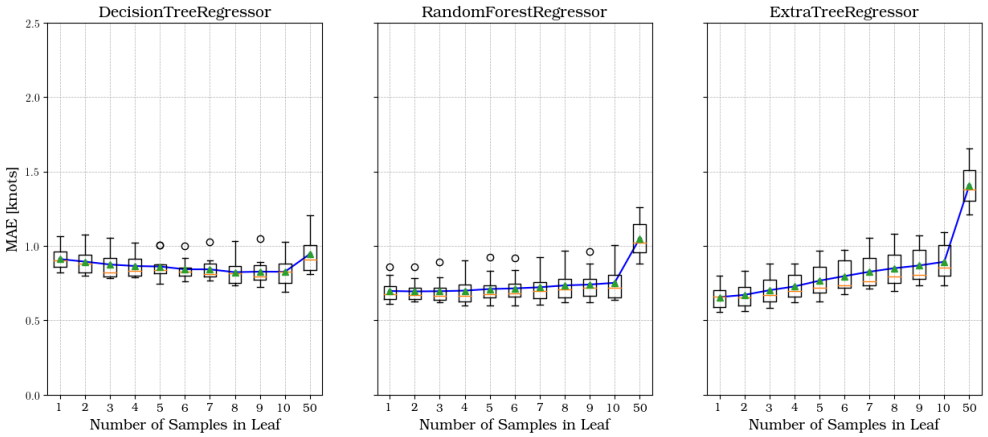

Figure 3.15: Hyperparameter tuning of `min_samples_leaf`

Depth of Tree

Defined with default value as `max_depth=None` in `Scikit-Learn`. This hyperparameter is defined as the count of nodes along a path from the root node to its parent node. Leaving it at `max_depth=None` means the tree will grow until all leaves are pure i.e. until minimum MSE is obtained or when the number of samples is less

than the minimum number of samples required to split an internal node. Similar to `min_samples_leaf`, DTR shows improvement until a certain breakeven point. RFR performance seems to stabilise at a certain depth while ETR benefits from allowing full growth of the tree. It can also be observed that the models' performance are identical for `max_depth=1`, which is evident as shown in Figure 3.16

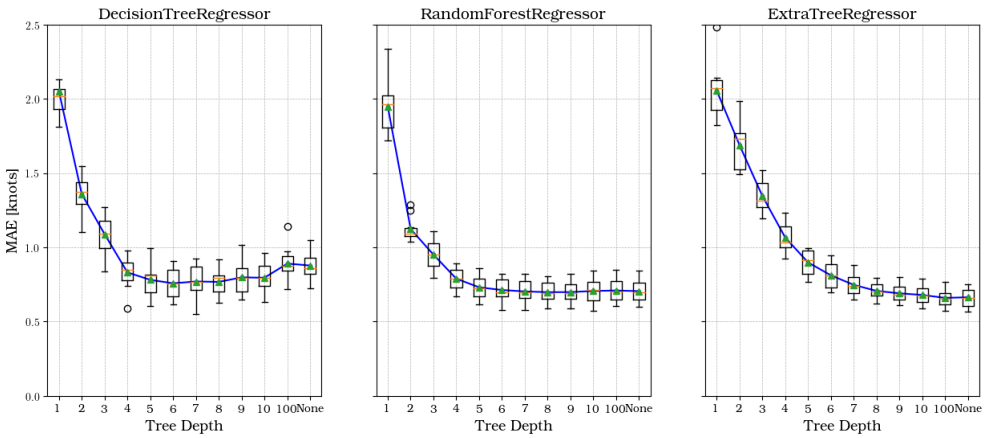

Figure 3.16: Hyperparameter tuning of `max_depth`

Number of Trees

Defined with default value as `n_estimators=100`. This hyperparameter controls the amount of trees i.e. predictors in a forest. Tuning the number of trees will affect the training time, and it is only available to RFR and ETR. The default value seems to yield a satisfactory result, as the performance for both RFR and ETR stabilise after in this case stabilise after 100 trees, as seen in Figure 3.13.

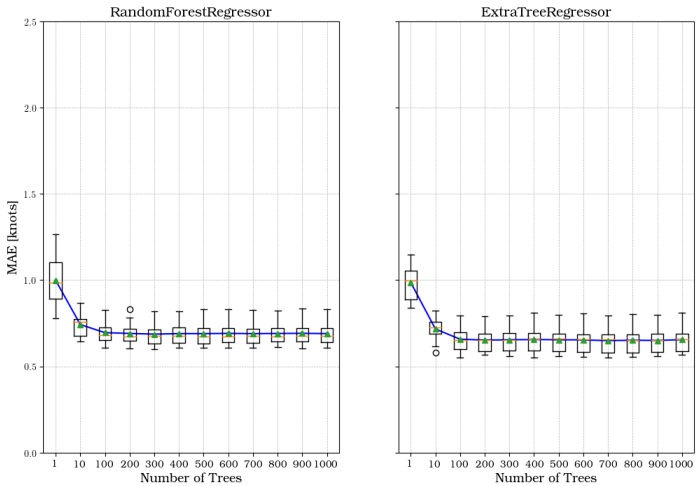

Figure 3.17: Hyperparameter tuning of `n_estimators`

3.4 White Box Modelling

This section involves the integration of the predicted SOG from the BBM into the WBM for the estimation of bunker fuel consumption (FOC). The Holtrop-Mennen approximation method will be employed to estimate the ship's encountered resistance, which will enable the determination of the total resistance crucial for calculating the power required for propulsion. However, it's worth noting that in the process of resistance calculation, certain form coefficients and ship parameters may not be readily available and could necessitate assumptions based on existing literature. Such assumptions will be explicitly indicated throughout this section. Subsequently, the resulting brake power P_B will be plotted against the STW v_S to generate a power-speed curve model. This model offers an alternative approach for estimating the FOC, representing the energy expended during a specific voyage.

3.4.1 Calculation of Total Resistance

The formula used to calculate total resistance R_{TOTAL} in this section is presented in Section 2.5.2 with the principle dimension of the ship which was given in Figure 3.4. Some assumptions are made for the sea state and the values of some form coefficients are calculated based on empirical formulas presented in Section 2.5.1. In this case study, Holtrop-Mennen method can be used as it fulfils the condition set in Equation (2.5.10):

$$\begin{aligned} Fr &= 0.2417 \leqslant 0.45 \\ 0.55 \leqslant C_P &= 0.6707 \leqslant 0.85 \\ 3.9 \leqslant \frac{L}{B} &= 5.91 \leqslant 9.5 \end{aligned} \tag{3.4.1}$$

The calculations of resistance are dynamics as the Froude number Fr is based on v_S, and the design Froude number Fr_{DESIGN} is used to check use cases for some equations.

Calculation of frictional resistance R_F

For the calculation of the surface area of bare hull S, it is assumed that the aft has a U-shaped section. Then the appropriate C_{stern} can be calculated to obtain the constant c_{14} Equation (2.5.16). For the calculation of length of run L_R, approximation for ℓ_{CB} are made based on Equation (2.5.18). The constants c_{14} and L_R are used to calculate the form factor $1 + k_1$ which will be used to calculate R_F.

Calculation of appendage resistance R_{APP}

From known ship information and ship schematics shown in Figure 3.6, it can be deducted that the ship consists of the following appendages:

- Two high-lift flap rudders
- Single centre skeg

- Twin shafts supported by two brackets

The assumptions for the appendage area are made by scaling the schematics to known measurements (e.g. the L_{WL}). From here, the appropriate k_{2_i} constants for individual appendages will be selected from Table 2.3 to obtain the $(1 + k_{2_i})_{eq}$ from Equation (2.5.20).

Appendage type	Value	$(1 + k_{2_i})$
Two high-lift flap rudders		3
h_{RUDDER}	4.06 m	
B_{RUDDER}	1.99 m	
S_{RUDDER}	16.16 m^2	
Single centre skeg		1.5
h_{SKEG}	4.41 m	
B_{SKEG}	26.23 m	
S_{SKEG}	115.67 m^2	
Twin shafts supported by two brackets		3
D_{SHAFT}	0.55 m	
L_{SHAFT}	13.54 m	
S_{SHAFT}	46.79 m^2	
$S_{APP_{tot}}$	**178.62** m^2	
$(1 + k_{2_i})_{eq}$		**2.03**

Table 3.5: Assumed appendage values

Additionally, there are two bow thrusters installed with approximated diameter of $d_{TH} = 2.15m$, from here, the constant C_{D_TH} can be approximated using Equation (2.5.21). Hence, the appendage resistance R_{APP} can be calculated using Equation (2.5.22).

Calculation of wave resistance R_W

The calculation of wave resistance is based on the case for $Fr \leq 0.4$ using equation Equation (2.5.24). The estimation is done by adding the constants presented between Equation (2.5.27) and Equation (2.5.38). There are some use cases for some equations, which is summarised in Table 3.6.

Constant	Use Case	Equation
c_7	$0.11 < \frac{B}{L_{WL}} \leq 0.25$	Equation (2.5.27)
c_{15}	$\frac{L_{WL}^2}{V} \leq 512$	Equation (2.5.33)
c_{16}	$C_P \leq 0.8$	Equation (2.5.34)
λ	$L_{WL} \leq 12$	Equation (2.5.36)

Table 3.6: Use case of constants for R_W

Calculation of bulbous bow resistance R_B

The area A_{BT} used to calculate is approximated based on **Kracht (1978)** with:

$$A_{BT} = 0.085A_M \tag{3.4.2}$$

For the height h_B, the upper limit of $h_B = 0.6T_{DESIGN}$ is selected, and it is assumed that $T_F = T_{DESIGN}$.

Calculation of (immersed) transom resistance R_{TR}

Since the immersed transom area is unknown, **Rakke (2016)** approximated the immersed transom area based on the correlation of ship dimension from the literature review of **Holtrop and Mennen (1982)**:

$$A_{TR} = 0.051A_M \tag{3.4.3}$$

This approximation must be used with caution as it is only based on case study of **Holtrop and Mennen (1982)**. However, to author's best knowledge, there are no other literature that provide empirical estimation of A_{TR}. Therefore, this estimation will be selected in this case study. The selection for the value of constant c_6 is dependent on the value of Fr_T, which is a function of v_S. From there, the transom resistance can be calculated using Equation (2.5.44)

Calculation of correlation allowance resistance R_A

The selection of constant c_4 in equation Equation (2.5.45) is based on the T_F, then the correlation resistance R_A can be calculated using Equation (2.5.47).

Calculation of added resistance due to wind R_{AA}

Two assumptions are made during the calculation of R_{AA}, since the information of lateral area A_L and A_F are not readily available, these values are assumed based on the dimension of similar ferry in the case study of **Blendermann (1994)**. It is assumed that the ferry has an A_L of **2125.80** m^2 and A_F of **325.30** m^2. From Table 2.4, the case for ferry ship is taken to get the necessary constants for the calculation of R_{AA}.

Calculation of added resistance due to wave R_{AW}

This part of the equation is relatively straightforward, L_{BWL} will be approximated to about **43.75** m. The calculation of R_{AWL} will be based on the data of significant wave height $H_{1/3}$ from the dataset.

3.4.2 Calculation of total efficiency η_{TOT}

Calculation of open water efficiency η_O

This value is approximated based on the line of the Wageningen series in Figure 2.15 **(Breslin and Andersen 1994)**. The case will be for "Passenger ships and high-speed naval vessels". Since the value of C_{Th} is not available, the value of η_O is approximated as **0.7**.

Calculation of hull efficiency η_H

For the calculation of η_H, the value of the propeller diameter D is approximated as **4 m**, which is based on the schematics of the ship shown in Figure 3.6.

Calculation of relative rotative efficiency η_R

The missing value required to compute Equation (2.5.59) is the pitch-diameter propeller ratio. This value will be estimated as P/D = **1.135**, which is obtained from the work of **Bertram (2000)**.

	Tanker	*Series 60*	*Container*	*Ferry*
Scale	1:35	1:26	1:34	1:16
L_{pp}	8.286 m	7.034 m	8.029 m	8.725 m
B	1.357 m	1.005 m	0.947 m	1.048 m
T_{fp}	0.463 m	0.402 m	0.359 m	0.369 m
T_m	0.459 m	0.402 m	0.359 m	0.369 m
T_{ap}	0.456 m	0.402 m	0.359 m	0.369 m
C_B	0.805	0.700	0.604	0.644
Coord. origin aft of FP	4.143 m	3.517 m	4.014 m	4.362
LCG	−0.270 m	0.035 m	−0.160 m	−0.149 m
Radius of gyration i_z	1.900 m	1.580 m	1.820 m	1.89 m
No. of propellers	1	1	2	2
Propeller turning	right	right	outward	outward
Propeller diameter	0.226 m	0.279 m	0.181 m	0.215
Propeller P/D	0.745	1.012	1.200	1.135
Propeller A_E/A_0	0.60	0.50	0.86	0.52
No. of blades	5	4	5	4

Figure 3.18: Estimated value of propeller dimensions **(Bertram 2000)**

Calculation of shaft efficiency η_S

The value of shaft efficiency is estimated as η_S = **0.99** based on **MAN (2011)** and **Holtrop and Mennen (1982)**.

3.4.3 Calculation of FOC

Once R_{TOTAL} and η_{TOTAL} are determined, the brake power of the ship, denoted as P_B, can be computed using Equation 2.5.3. The resultant P_B values will then be graphed against the ship's STW (v_S), producing a power-speed curve. A regression

Parameter	Value	Remarks
g	9.805 kg/ms^2	
ρ_{sea}	1025 kg/m^3	
ν_{sea}	0.00000118 m^2/s	
ρ_{air}	1.25 kg/m^3	
1 m/s	1.9438 knots	
Required Parameters for Holtrop-Mennen		
L_{WL}	144.80 m	From Figure 3.4
B	24.50 m	From Figure 3.4
T	5.85 m	Assume $T_A = T_F = T$ for initial phase, otherwise use T from dataset, also assume maximum draught
V	13592.1413 m^3	$V = C_B \cdot L_{WL} \cdot T_{MAX}$
Fr_N	0.2417	From Equation (2.5.6)
C_B	0.6549	From Equation (2.5.5)
C_M	0.9764	From Equation (2.5.7)
C_P	0.6707	From Equation (2.5.8)
C_{WP}	0.7700	From Equation (2.5.9)
ℓ_{CB}	-0.0123	Equation (2.5.18)
A_{TR}	7.3581 m^2	Equation (3.4.3)
A_{BT}	12.2634 m^2	Equation (3.4.2)
h_B	3.5100 m	Assume upper limit $h_B = 0.6T_F$
D	4 m	Approximated from schematics Figure 3.6
A_E/A_0	1.135	Value assumed from Figure 3.18
C_{stern}	10	Assume u-shaped section Equation (2.5.16)
Optional Parameters for Holtrop-Mennen		
S	3881.0231 m^2	approximated from Equation (2.5.14)
S_{APP}	178.62m^2	approximated from schematics Figure 3.6
i_E	21.6014°	Equation (2.5.29)
d_{TH}	2.15 m	Approximated from schematics Figure 3.6

Table 3.7: Assumed values for power estimation

line will be fitted to the data points on this curve. Since each BBM model generates distinct SOG predictions, the regression equation for the line signifies the characteristics of different BBM models. To assess the performance of each BBM, the regression models produced by each BBM will be compared against the regression model derived from actual data. This evaluation will gauge the predictive capabilities of each model. Subsequently, the FOC can be calculated using Equation 2.5.4. The specific fuel oil consumption (SFOC) information can be extracted from Table 3.4. Without multiplication by the operational time τ_{OP}, the fuel consumption in metric tons per hour (T/h) can be computed by dividing the value by 1×10^6.

Chapter 4

Result and Discussion

To assess the performance of the GBM, a case study will be conducted using the test dataset, which contains journey data for the entire year of 2021. The evaluation process consists of two main parts. The first part involves assessing the performance of the BBM, where the trained model will predict the ship's SOG. The output of BBM, which is the SOG, will be fed to the WBM to estimate the power and subsequently the bunker consumption. For further clarity regarding the methodology, the following steps are taken which are based on the proposed methodology shown in Figure 3.1 and Figure 3.2. For generation of the BBM, the steps taken are:

1. Dataset is loaded.
2. Identify and remove any anomalies.
3. Remove static and unneeded features.
4. Apply speed threshold of 5 knots.
5. Highly correlated features are combined/removed based on physical and statistical reasoning.
6. Impute missing values using `KNNImputer`.
7. Split the dataset into training and testing.
8. Train the model using the whole dataset with default hyperparameter.
9. Evaluate model performance using k-fold cross-validation.
10. Tune the model until the best model is obtained.
11. For the case study, the best models will be used to predict the SOG using the test dataset.

Subsequently, for FOC calculation, the following steps are taken:

1. The test dataset is split into seasonal data. Summer-Fall season and Winter-Spring season corresponding to data for 6 months respectively.
2. Impute missing values using `KNNImputer`.
3. SOG is converted to STW.
4. Calculate calm water resistance R_{CALM}.

5. Calculate added resistance due to wave R_{AW}.
6. Calculate added resistance due to wind R_{AA}.
7. Calculate total effective power P_E using total resistance R_{TOTAL}.
8. Calculate brake power P_B from total efficiencies.
9. Plot resulting regression line for Power-Speed curve from all models and actual case.
10. Calculate the FOC by considering the engine SFOC and operation time.
11. Plot resulting regression line for FOC-Speed curve from all models and actual case.
12. Evaluate the performance of the model generated from the regression lines.

4.1 Evaluation of BBM

4.1.1 Model Training and Selection of Optimal Parameter

As noted in Section 3.3, the training dataset encompasses 2871 data points. To refine the range of hyperparameter search for the tree-based model, MAE plots were generated against varying hyperparameter values, a technique outlined in Section 3.3.2. The hyperparameters are iteratively tuned until the optimal model is achieved. The outcome of this hyperparameter optimization process is presented in Table 4.1. The model training is conducted using an **AMD Ryzen 7 2700X, Eight-Core Processor** operating at 3.7 GHz, with 16384 MB of installed RAM.

With the default hyperparameters, the RFR exhibits the lengthiest training duration, followed by the ETR and the DTR. This aligns with expectations, as RFR employs a greedy algorithm, seeking the optimal feature for node splitting, leading to increased training time. ETR, on the other hand, trains faster due to its random feature selection during node splitting. DTR displayed the shortest training time as it constructs only a single tree. However, in the case of the optimised model, ETR requires a lengthier training period compared to RFR. This discrepancy arises from the number of trees in the optimized model, controlled by the `n_estimator` parameter. The optimized ETR model has 800 trees, while RFR comprises 100 trees. Notably, the training time of the optimized DTR model is halved, as pruning the tree results in a simpler model requiring less time for training.

To further investigate the effect of hyperparameter optimisation, the learning curve of each tree-based model is plotted. For DTR, generated model with default parameter will result in a model that heavily overfits the training data, which is evident from the large gap between the training error and validation error which indicated a high variance as shown in Figure 4.1. Regularisation i.e. parameter tuning of the DTR model helps balance between bias and variance by trading off bias for variance. This is observed from the substantial reduction in the gap between the training and validation error from Figure 4.1. Additionally, the learning curve indicates that the

Model	Training time [s]	Optimal Hyperparameter	Search Range
DTR	0.044	None	
DTR_{OPT}	0.021	`min_samples_split = 7`	[2,10]
		`min_samples_leaf = 10`	[1,10]
		`max_features = 12`	[1,12]
		`max_depth = 8`	[1,10] and [None]
RFR	4.112	None	
RFR_{OPT}	3.431	`min_samples_split = 2`	[2,10]
		`min_samples_leaf = 1`	[1,10]
		`max_features = 10`	[6,12]
		`max_depth = 120`	[10,200] and [None]
		`n_estimators = 100`	[100,1000]
ETR	0.944	None	
ETR_{OPT}	4.390	`min_samples_split = 9`	[1,10]
		`min_samples_leaf = 1`	[1,10]
		`max_features = 12` [1,12]	
		`max_depth = 120`	[10,200] and [None]
		`n_estimators = 800`	[100,1000]
MLR	0.004	None	

Table 4.1: Optimal hyperparameter with training time of each model

model performance can be improved by increasing the amount of data points as the MAE continue to decrease with increasing amount of data points.

The process of hyperparameter tuning for the Random Forest Regressor (RFR) model did not show any significant improvement in model performance. This outcome aligns with the findings of **Kuhn and Johnson (2013)** and **Hastie, Tibshirani, and Friedman (2009)** which was discussed in Section 2.2.2. The model starts to plateau at approximately 1000 data points. Furthermore, there is a noticeable variance in the RFR model, which indicates that the model will have a slight tendency to overfit. On the other hand, hyperparameter tuning contributes to variance reduction in the ETR model. However, its impact on overall model performance remains limited. The ETR model reaches a performance plateau beyond 1000 data points, indicating that augmenting the dataset with additional points is unlikely to yield substantial improvements in model effectiveness.

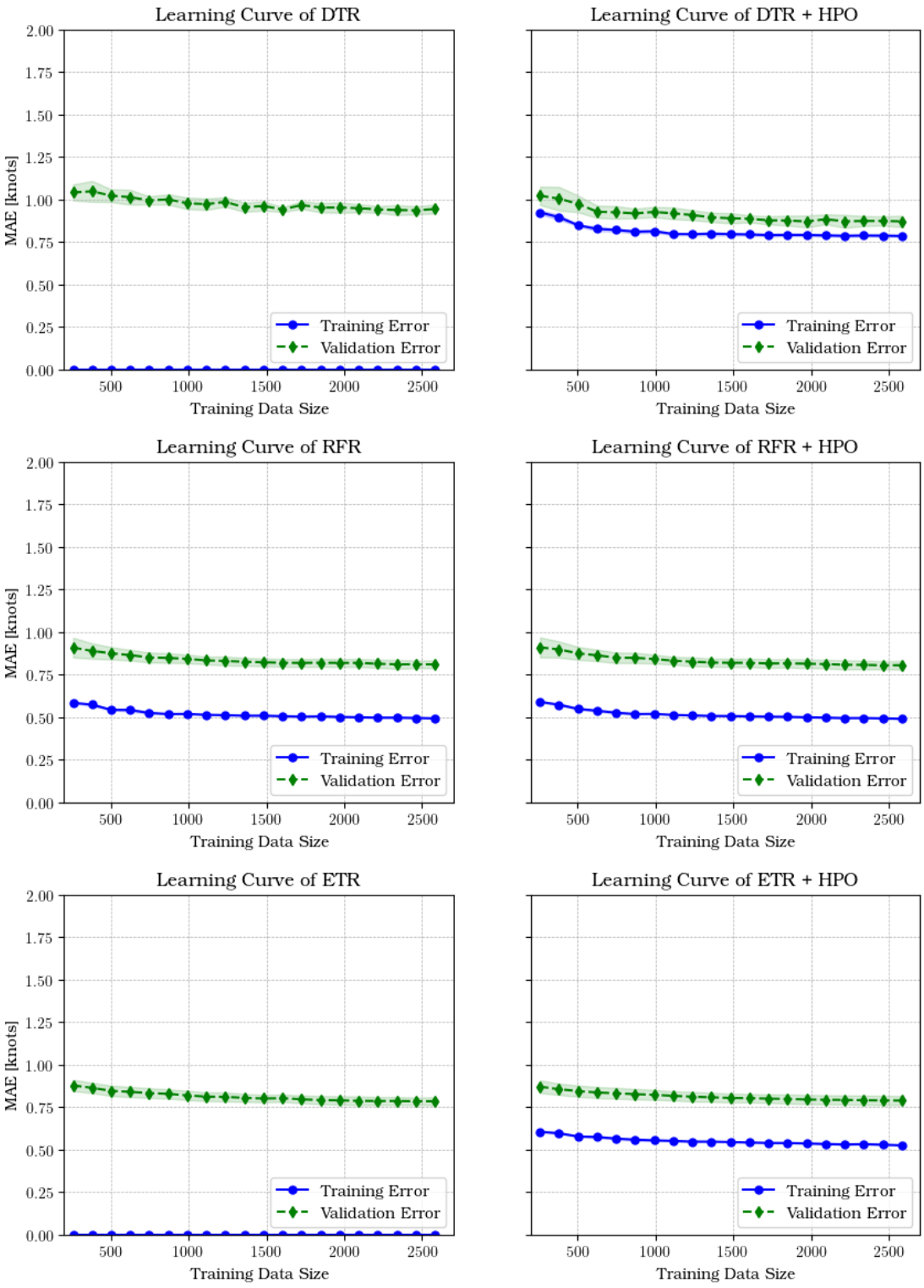

Figure 4.1: Learning curve of various tree-based models

4.1.2 Analysis of trained model

4.1.2.1 Feature Importance

As discussed in Section 2.2.2, tree-based models inherently possess the ability to quantify the influence of each feature during the splitting process. This is performed using `feature_importances_` feature in `Scikit-Learn` **(Kuhn and Johnson 2013)**. According to documentation by **Pedregosa et al. (2011)**, this attribute is computed as the mean and standard deviation of the accumulated reduction in impurity within each tree, which is essentially the total reduction in the criterion achieved by a specific feature. Alternatively, it can be interpreted as a measure of how extensively a feature is employed in each tree.

DTR_{OPT}		RFR_{OPT}		ETR_{OPT}	
Feature	Importance	Feature	Importance	Feature	Importance
heading	0.6563	heading	0.4927	cog	0.6410
cog	0.3183	cog	0.4183	heading	0.2707
draught	0.0105	draught	0.0210	truecurrentdir	0.0200
truewinddir	0.0047	curspeed	0.0104	draught	0.0144
oceantemperature	0.0029	waveperiod	0.0093	windwaveswellheight	0.0112
surftemp	0.0025	truecurrentdir	0.0092	curspeed	0.0110
waveperiod	0.0019	windwaveswellheight	0.0084	waveperiod	0.0095
truecurrentdir	0.0010	surftemp	0.0075	windspeed	0.0053
windwaveswellheight	0.0008	truewinddir	0.0075	surftemp	0.0046
curspeed	0.0004	truewavedir	0.0058	truewavedir	0.0045
windspeed	0.0004	oceantemperature	0.0057	oceantemperature	0.0044
truwavedir	0.0001	windspeed	0.0056	truewinddir	0.0033

Table 4.2: Feature importance of different models

The feature importances for all tree-based models shown in Table 4.2 indicated that the structure of the model is significantly influenced by the features `heading` and `cog`. This observation suggests that the models rely considerably on ship movement direction, represented by heading and COG, to predict the SOG at a given location. However, from a physical perspective, it will be more insightful to consider the ship state and weather conditions that affect the prediction of the SOG.

Excluding ship heading and COG. The ship draught T emerges as a significant factor influencing the prediction of SOG. This aligns with the theory of frictional resistance R_F encountered by the ship, which is discussed in Section 2.5.2.1. Equation (2.5.12) is a function of wetted surface area of bare hull S. Deeper draught T will result in more submerged area of the hull and this will consequently increase the frictional force R_F of the ship. Given a constant supply of power to the ship propulsion system, the speed of the ship will decrease which is shown in Equation (2.5.2).

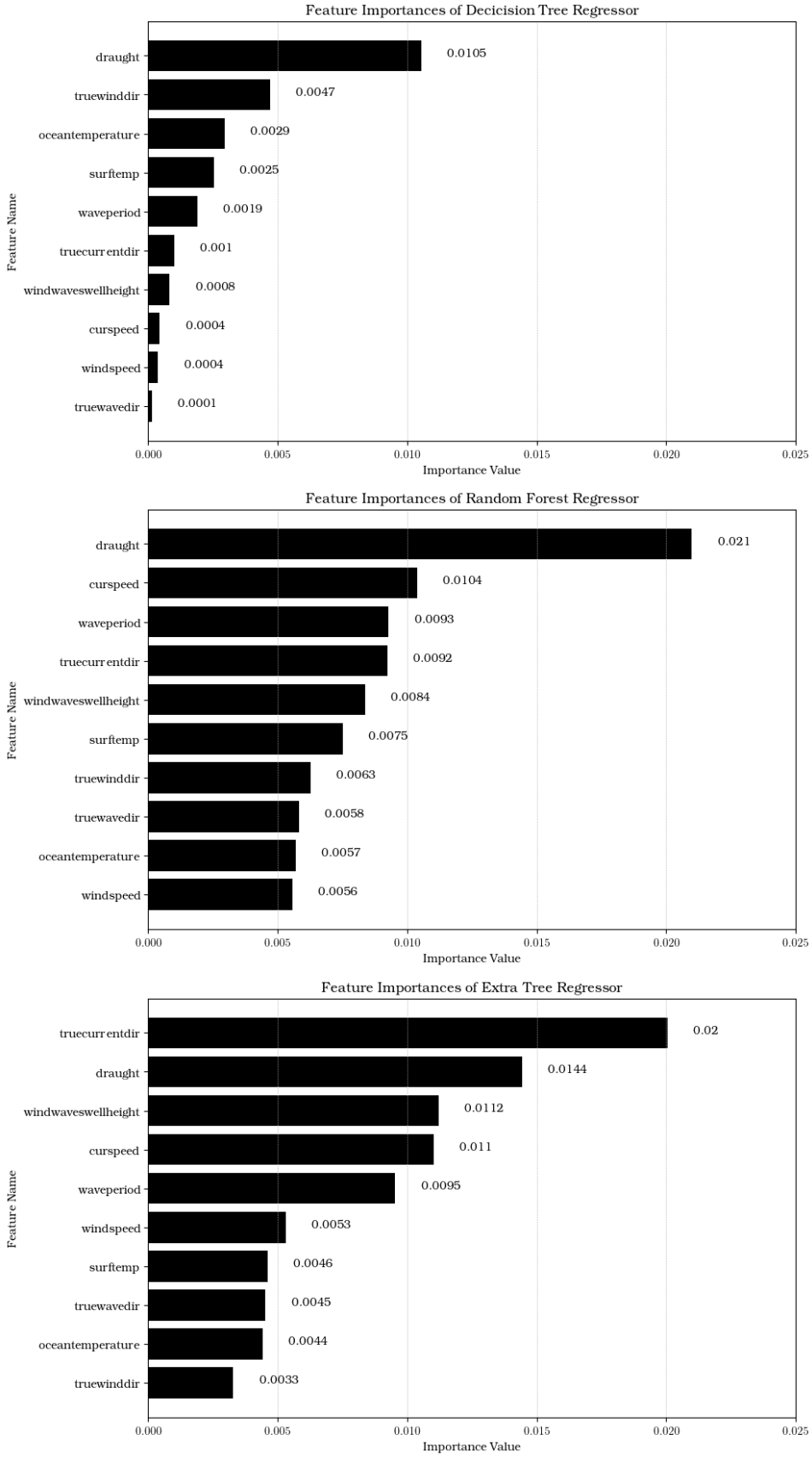

Figure 4.2: Feature importance of different tree-based models

Concerning weather states, both the RFR and ETR models identify current-based information, such as current speed and true current direction, as the most influential factors affecting SOG prediction. This observation concurs with the suggested methodology for current correction presented in Section Section 2.3.2, which outlines that the process of converting SOG to STW necessitates both the magnitude and direction of the current. The subsequent influential features, according to the rankings of the RFR and ETR models, are wave-related attributes: significant wave height ($H_{1/3}$), true wave direction, and wave period (`waveperiod`). This alignment reflects the added resistance caused by waves (R_{AW}) in the computation of the total resistance (R_{TOTAL}) experienced by the ship. On the other hand, wind-related features, encompassing wind speed and its true direction, are associated with the added resistance due to wind force (R_{AA}). However, these factors are found to be the least impactful in the prediction of SOG according to the RFR and ETR models.

Based on the behaviour of the Random Forest Regressor (RFR) and Extra Trees Regressor (ETR) models, it can be inferred that waves have a more significant impact on the Speed Over Ground (SOG) compared to the influence of wind during the ship's journey. However, the Decision Tree Regressor (DTR) model demonstrates that temperature-related features, such as Sea Surface Temperature (SST) and air temperature above the ocean, have a more significant effect on SOG predictions than most other features. While the importance of temperature is not as pronounced as in RFR or ETR models, this finding suggests that the ship's SOG is implicitly influenced by the time of the travel or the season in which the journey takes place.

Structure of generated tree-based model

To comprehend the impact of hyperparameter optimisation and feature importance, an analysis of the structure of the generated tree-based models will be conducted. The shading within the nodes conveys the probability of the decision, with darker shading indicating a higher likelihood. Each node provides information about the splitting feature, along with its threshold, SSR (Sum of Squared Residuals) value, sample count, and the predicted SOG value. Despite pruning efforts, the tree structure can potentially become quite large. To enhance clarity, the visualization of the trees will be restricted to a maximum depth of `max_depth = 3`. Additionally, for the RFR and ETR models, only the illustration of a specific tree within the forest will be presented.

The structure of the optimized decision tree, as depicted in Figure 4.3, illustrates the impact of regularisation at the leaf nodes. Notably, the leaf nodes that partition the feature ocean temperature do not achieve a complete minimization of the SSR. This outcome arises from the hyperparameter tuning of the minimum samples at the leaf nodes, set at `min_samples_leaf = 10`. Further division of these nodes would result in subsequent leaf nodes with fewer than 10 samples. Within this visualisation, the significance of features such as COG and ship heading becomes evident, as they are employed to partition numerous internal nodes in the tree.

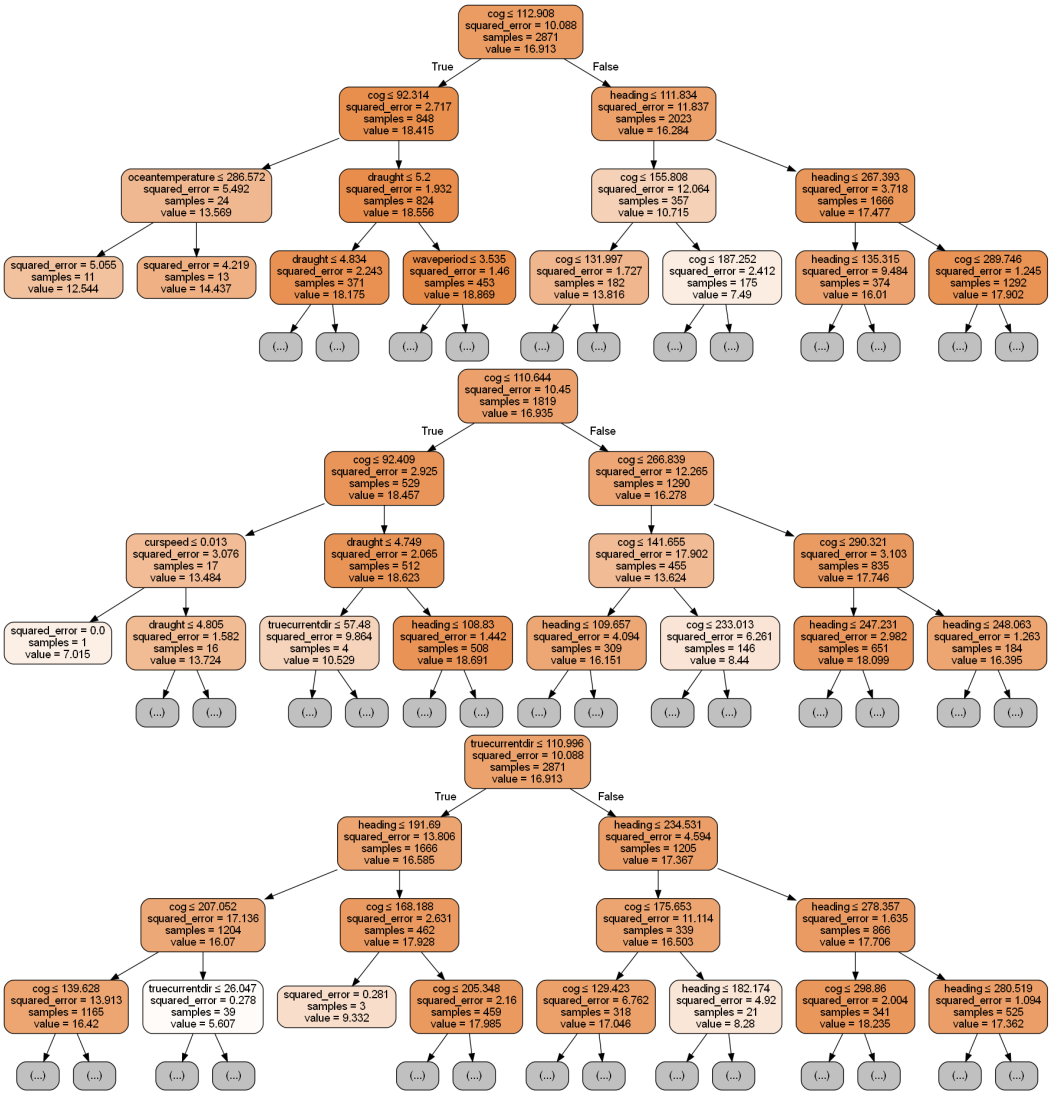

Figure 4.3: Partial structure of DTR, RFR, and ETR

The illustration of the first RFR tree is shown in Figure 4.3. Similar to DTR, both COG and ship heading are regarded as the best features to split the internal node. In this tree of the forest, the effect of allowing full tree growth can be observed in the leaf node when splitting the feature current speed. This tree is able to minimise the SSR to its possible minimum value, and the leaf node cannot be further split as there are no more available samples. The effect of bagging for the dataset and feature selection in RFR can also be observed in this tree as the structure of this tree is completely different to DTR tree shown in Figure 4.3.

The influence of random feature selection in ETR is demonstrated by the structure of its first tree, as presented in Figure 4.3. Unlike the DTR and RFR, which employ a greedy algorithm to identify splits minimizing the cost function, the ETR exhibits a noticeable randomness in feature selection. In this specific illustration, the ETR model designates the true current direction as the parent node. Furthermore, due to

the regularisation applied to the ETR, the leaf node formed during the COG split does not entirely minimize the SSR. It is worth noting that this particular split does not occur due to the constraint imposed by the tuning parameter `min_samples_split = 9`.

4.1.2.2 Evaluation of k-fold cross-validation

The results of the 10-fold validation process are shown in Figure 4.4. The inner (orange) line corresponds to the median, representing the central 50% of scores in the k-folding procedure. The upper and lower boundaries of the box signify the first (25%) and third (75%) quartiles, respectively. The whiskers denote the lowest data point within a 1.5 Interquartile Range (IQR) of the lower quartile and the highest data point within a 1.5 IQR of the upper quartile. The mean value is indicated by the (green) triangle, while data points beyond the whisker range are depicted as hollow circles.

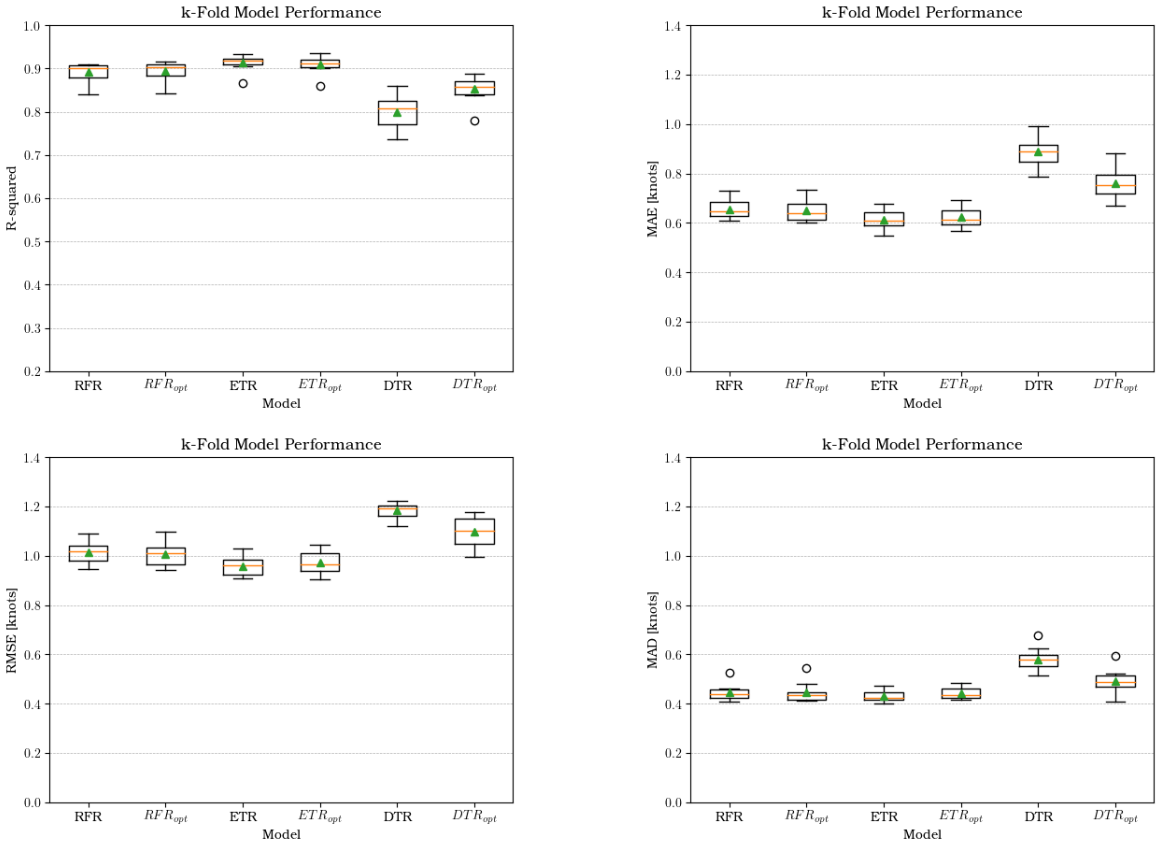

Figure 4.4: Evaluation of k-fold cross-validation for different performance indices

The box plots reveal that ETR outperformed other tree-based models, achieving a R^2 score of approximately 91% and an MAE of around 0.6 knots. This model also demonstrated notable stability, evident from the narrow length between the first and third quartile. RFR showcased a similar level of performance, attaining an R^2 score of roughly 89% and an MAE of about 0.65 knots. As demonstrated previously in Figure 4.1, hyperparameter optimization did not lead to any substantial enhancement

in model performance. DTR significantly benefited from the process of regularisation, resulting in an increase of around 5% in the R^2 score and a reduction in MAE from about 0.89 knots to 0.76 knots. Similar improvements can be observed in both RMSE and MAD. In summary, all tree-based models demonstrated a satisfactory fit, with mean/median R^2 scores exceeding 80%. However, the RMSE values are relatively significant, ranging from 1.00 to 1.20 knots across the models. To provide context, the mean SOG of the training data is 16.91 knots, as indicated in Table 3.3.

4.1.3 Performance evaluation of BBM

4.1.3.1 Analysing the testing dataset

Once the best-optimised model is identified, the performance of the model will be further evaluated using the testing dataset. This testing dataset comprises 957 data points from the year 2021, encompassing the entirety of the year. The dataset for the full year is denoted as DS_{year}. In order to explore the influence of data points on model performance, the dataset is divided into two distinct seasons: DS_{summer}, covering the period from May 2021 to October 2021, consisting of 454 data points; and DS_{winter}, encompassing data from January 2021 to April 2021, as well as November 2021 to December 2021, with a total of 503 data points. Any missing values present within the testing dataset will be addressed using the `KNNImputer` method.

Features	Count	Mean	Std.	Min	25%	50%	75%	Max
sog	957.00	16.99	3.10	5.10	16.68	18.05	18.72	21.00
cog	957.00	196.73	86.72	56.02	102.32	185.22	282.18	319.85
heading	957.00	188.30	89.17	63.49	100.86	124.24	279.38	308.04
draught	957.00	5.23	0.19	4.74	5.11	5.29	5.38	5.66
windspeed	957.00	6.45	3.04	0.40	4.11	6.13	8.21	15.85
oceantemperature	957.00	282.28	6.48	267.25	276.80	281.91	288.42	295.70
waveperiod	957.00	3.69	0.88	1.67	3.06	3.55	4.22	7.01
surftemp	957.00	283.20	5.72	273.15	277.98	282.65	288.82	294.93
windwaveswellheight	957.00	0.77	0.54	0.08	0.37	0.63	0.95	3.24
curspeed	957.00	0.09	0.07	0.00	0.05	0.07	0.13	0.50
truewinddir	957.00	91.39	56.23	0.03	38.80	95.25	142.83	179.86
truecurrentdir	957.00	90.75	57.76	0.26	31.52	90.44	144.65	179.95
truewavedir	957.00	86.79	55.76	0.06	35.81	82.32	138.93	179.81

Table 4.3: Descriptive statistics of DS_{year}

Features	Count	Mean	std	Min	25%	50%	75%	Max
sog	454.00	17.26	2.91	5.22	16.74	18.17	18.95	21.01
cog	454.00	196.06	87.55	56.02	102.80	182.79	282.03	319.85
heading	454.00	188.08	89.02	63.49	100.75	124.68	278.07	303.30
draught	454.00	5.30	0.17	4.74	5.20	5.29	5.38	5.66
windspeed	454.00	6.64	3.33	0.40	4.08	6.30	8.71	15.85
oceantemperature	454.00	285.59	5.90	269.27	282.90	286.70	290.04	295.70
waveperiod	454.00	3.73	0.99	2.02	2.95	3.57	4.36	7.01
surftemp	454.00	286.40	5.09	274.75	283.17	287.81	290.18	294.93
windwaveswellheight	454.00	0.82	0.63	0.08	0.36	0.65	1.02	3.24
curspeed	454.00	0.10	0.07	0.00	0.04	0.07	0.13	0.50
truewinddir	454.00	90.94	58.05	0.60	38.40	89.86	145.86	179.58
truecurrentdir	454.00	83.65	59.53	0.26	26.68	70.51	143.73	179.33
truewavedir	454.00	87.79	59.58	0.09	32.49	82.34	145.08	179.81

Table 4.4: Descriptive statistics of DS_{summer}

Features	Count	Mean	std	Min	25%	50%	75%	Max
sog	503.00	16.75	3.24	5.10	16.59	17.98	18.61	20.70
cog	503.00	197.33	86.06	80.81	102.25	187.56	282.63	307.92
heading	503.00	188.50	89.39	89.22	100.87	123.92	280.05	308.04
draught	503.00	5.16	0.18	4.76	5.02	5.20	5.29	5.65
windspeed	503.00	6.28	2.76	0.43	4.12	6.05	8.01	14.35
oceantemperature	503.00	279.29	5.44	267.25	275.74	278.22	281.25	292.72
waveperiod	503.00	3.67	0.76	1.67	3.16	3.62	4.13	5.98
surftemp	503.00	280.30	4.67	273.15	277.23	278.67	282.31	292.85
windwaveswellheight	503.00	0.73	0.44	0.08	0.40	0.66	0.89	2.43
curspeed	503.00	0.09	0.07	0.00	0.05	0.08	0.12	0.42
truewinddir	503.00	91.81	54.61	0.03	39.66	97.92	140.20	179.86
truecurrentdir	503.00	97.16	55.40	1.44	41.92	102.12	145.34	179.95
truewavedir	503.00	86.29	51.48	0.06	40.29	82.36	131.52	178.30

Table 4.5: Descriptive statistics of DS_{winter}

4.1.3.2 Result and Discussion of BBM

The results of SOG prediction of optimised tree-based models are summarised in Table 4.6. Each model is tested against 3 different testing datasets, the yearly dataset DS_{year}, summer dataset, DS_{summer} and winter dataset DS_{winter}. The performance of the tree models is compared against Multiple Linear Regressor (MLR) model.

Model	Dataset	R^2 [%]	expVar [%]	MAE $[kn]$	RMSE $[kn]$	MAD $[kn]$	MAPE [%]
DTR_{OPT}	DS_{year}	86.73	86.75	0.714	1.128	0.480	4.96
	DS_{winter}	88.52	88.62	0.690	1.098	0.441	4.93
	DS_{summer}	84.10	84.10	0.738	1.159	0.516	4.92
RFR_{OPT}	DS_{year}	90.10	90.13	0.619	0.974	0.417	4.29
	DS_{winter}	93.41	93.53	0.548	0.832	0.372	3.94
	DS_{summer}	85.48	85.48	0.693	1.108	0.452	4.63
ETR_{OPT}	DS_{year}	91.88	91.89	0.582	0.883	0.398	3.96
	DS_{winter}	**94.56**	**94.63**	**0.532**	**0.756**	**0.394**	**3.71**
	DS_{summer}	88.14	88.15	0.635	1.001	0.409	4.20
MLR	DS_{year}	69.57	69.62	1.147	1.709	0.917	7.75
	DS_{winter}	67.82	67.83	1.133	1.838	0.875	8.05
	DS_{summer}	71.24	71.63	1.159	1.559	0.952	7.38

Table 4.6: Performance indices for SOG predictions

The results presented in Table 4.6 shows that all tree-based models demonstrated good predictive capabilities across various testing datasets. Generally, all tree-based models obtained R^2 score above 80% and perform better when using the DS_{winter} datasets. All tree-based models offer a substantial improvement from MLR. Among the tree-based model, ETR presented the best SOG prediction across different datasets, closely followed by RFR and followed by DTR.

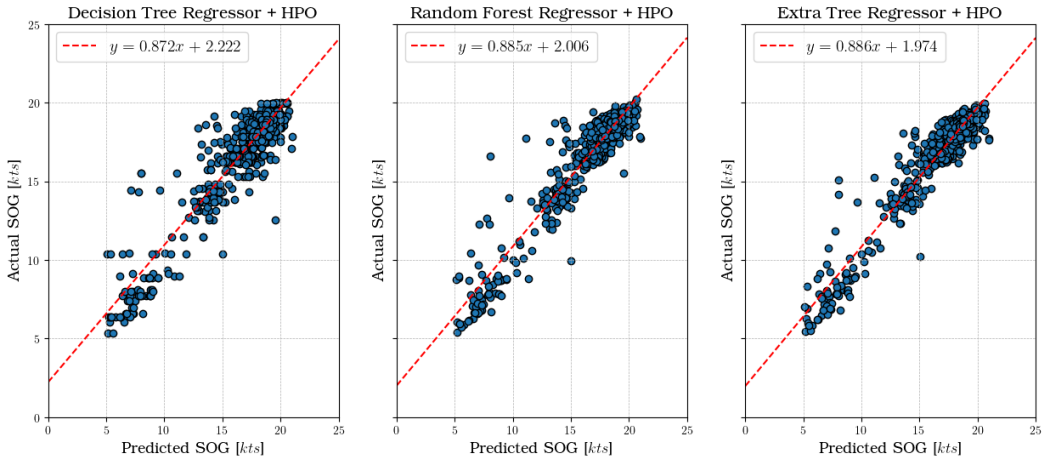

Figure 4.5: Actual vs Predicted SOG for DTR, RFR and ETR

Model	Dataset	Count	Mean	std	Min	25%	50%	75%	Max
Actual	DS_{year}	957.00	16.99	3.10	5.10	16.68	18.05	18.72	21.01
	DS_{winter}	503.00	16.75	3.24	5.10	16.59	17.98	18.61	20.70
	DS_{summer}	454.00	17.26	2.91	5.22	16.74	18.17	18.95	21.01
DTR_{OPT}	DS_{year}	957.00	17.04	2.90	5.36	16.58	18.11	18.66	20.00
	DS_{winter}	503.00	16.85	3.06	5.36	16.58	17.86	18.58	19.98
	DS_{summer}	454.00	17.25	2.72	5.98	16.58	18.15	18.91	20.00
RFR_{OPT}	DS_{year}	957.00	17.04	2.90	5.36	16.59	18.11	18.66	20.22
	DS_{winter}	503.00	16.86	3.04	5.36	16.70	18.02	18.60	19.74
	DS_{summer}	454.00	17.25	2.72	5.66	16.74	18.20	18.77	20.22
ETR_{OPT}	DS_{year}	957.00	17.03	2.87	5.39	16.67	18.05	18.65	19.96
	DS_{winter}	503.00	16.84	3.04	5.38	16.65	17.95	18.56	19.87
	DS_{summer}	454.00	17.23	2.65	5.90	16.71	18.18	18.75	19.96

Table 4.7: Descriptive statistics of SOG Prediction

For a deeper understanding of predictive performance, descriptive statistics of the SOG prediction are presented in Table 4.7. The results suggest that the model struggles to capture information at lower ship speeds across the various datasets. This limitation could potentially be caused by the scarcity of data points representing the lower end of SOG. Such sparsity is evident from the boxplots of diverse datasets depicted in Figure 4.6, as well as the actual versus predicted SOG plot illustrated in Figure 4.5, where the scarcity of information in lower SOG range data points becomes notably evident.

Both the descriptive statistics and boxplots indicate that the distribution of data points is skewed towards higher SOG values. The majority of data points cluster around approximately 16.5 knots for the first quartile and 19 knots for the third quartile, which is consistently observed across all datasets. The lowest data point within the $1.5 \cdot \text{IQR}_{LOW}$ range is situated around 13.6 knots, as evident from the lower whisker in the boxplot. Despite the appearance of potential outliers beyond the whisker in the boxplot, these points do not signify actual outliers; instead, they represent ship journeys at low speeds. The limited representation of low-speed data points contributes to the model's challenge in accurately predicting SOG values within the lower SOG range.

This observation could potentially explain the relatively high values of RMSE for all the models. As discussed earlier in Section 3.3.1, RMSE is more sensitive to outliers compared to both MAE and MAD, which makes it less suitable for this case study. In contrast, MAE and MAD provide more robust error evaluation metrics in situations like this. Additionally, the elevated R^2 scores and explained variance could be attributed to the models' effectiveness in predicting SOG values that fall within the interquartile ranges indicated by the boxplot whiskers. This suggests strong prediction performance within those specific ranges.

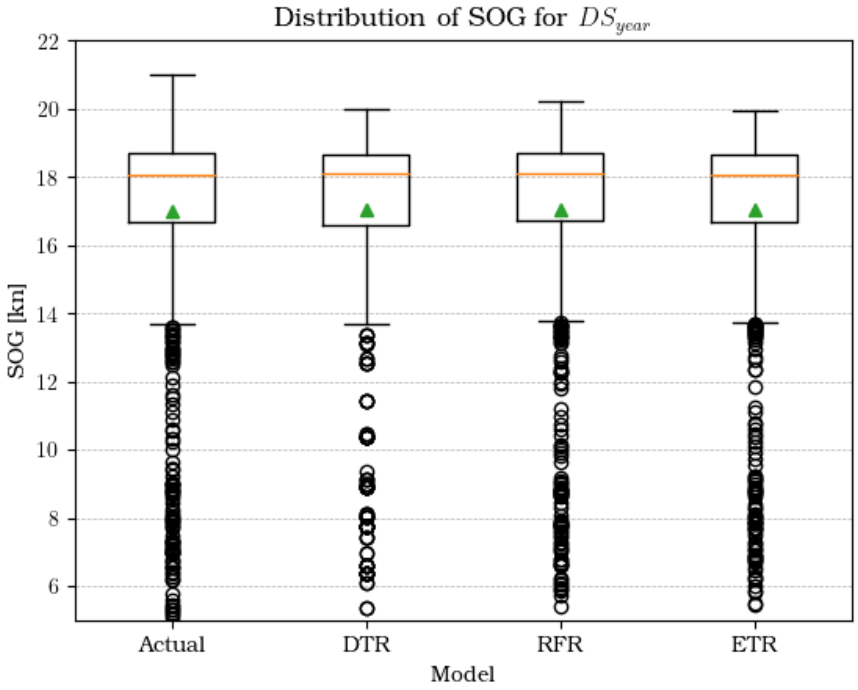

Figure 4.6: SOG distribution for DS_{year}

There are two possible solutions to address this problem:

Increasing the number of data points in the lower SOG Range: This is the most apparent solution to address this issue. By doing so, the interquartile range (IQR) of the boxplot can be expanded, potentially preventing the misclassification of lower SOG data points as outliers. Furthermore, this could lead to improvements in performance metrics such as R^2 and RMSE. However, it is important to note that in the context of this specific case study, this approach may not be feasible due to the limitations of T-AIS coverage, as illustrated in Figure 3.7. The ship's speed decrease as it approaches a port and it gradually increases as it departs from one. Unfortunately, T-AIS coverage around the ports of Køge and Rønne is either problematic, with a transmission and reception rate of approximately 50% to 80%, or poor, with a transmission rate of less than 50%. This presents a significant challenge, as increasing the number of data points around these port regions may not be achievable given the limitations of T-AIS transmission rates.

Reducing data Skewness: A strategy to address data skewness could involve applying outlier rejection techniques, as demonstrated by **Gkerekos, Lazakis, and Theotokatos (2019)**. The outlier rejection formula $\mu \pm 3\sigma$, where μ is the mean and σ is the standard deviation, could be employed. Assuming a normal distribution, this range should encompass approximately 99.7% of the normal data. Another approach to mitigate skewness is to set a higher threshold for the minimum SOG value. While this might lead to an overall reduction in errors and an improvement in model fit, it's important to note that such an approach could limit the model's ability to make accurate predictions when the ship is operating at lower speeds. Careful consideration

of the trade-offs between model performance and predictive capability is essential when determining the most suitable approach. The effect of reducing skewness on the testing datasets will be demonstrated in subsequent sections.

4.2 Evaluation of WBM

4.2.1 Analysis of WBM

The findings of power estimation using the Holtrop-Mennen method are presented in Table 4.8 and Figure 4.7. To gain insights into the overall model behaviour, the analysis is conducted using the actual data from the DS_{year} datasets. To facilitate a better interpretation of the scale of magnitude, the histograms for the resistance components utilise identical scaling.

For calm water resistance R_{CALM}, the largest portion is attributed to the frictional resistance R_F, which accounts for approximately 39% of R_{TOTAL} when considering the mean. Following this, the wave-making and breaking resistance R_W contribute around 21%. Subsequently, the bulbous bow resistance R_B accounts for approximately 16%, the correlation resistance R_A around 10%, and the appendage resistance R_{APP} at approximately 8%. The transom resistance R_{TR} has a relatively negligible impact on R_{CALM}, accounting for only about 1%.

Regarding the added resistance due to surrounding weather conditions, the contribution of added resistance due to wind R_{AA} are noticeable and adds a significant amount to the total resistance R_{TOTAL}. In contrast, the added resistance due to waves R_{AW} is comparatively negligible. This behaviour can be explained by referring to Table 4.3, the sea conditions along the ship's sailing path are relatively calm, with a mean significant wave height $H_{1/3}$ of 0.77 m and a mean wind speed of 6.45 m/s. By only considering the mean, the combined additional resistance to weather conditions only makes up about 3.5% of the total resistance.

These results further validate the BBM's ranking of ship draught T as the most significant physical factor that affects the ship speed. The WBM also further confirm the BBM suggestion that wave resistance might have a more significant impact on the speed compared to wind resistance. This is evident from the maximum achievable resistance of R_{AA} and R_{AWL} shown in Table 4.8. However, over the journey, the mean of R_{AWL} is significantly lower, primarily due to the condition in Equation (2.5.53), which only consider the wave correction force within ±45° off bow.

By considering the total resistance R_{TOTAL} and the total efficiencies η_{TOTAL}, the brake power P_B can be calculated. Using the brake power and the corresponding speeds, a power-to-speed curve can be plotted. This curve can then be transformed into a bunker-to-speed curve, which enables the estimation of the energy required for different operating speeds.

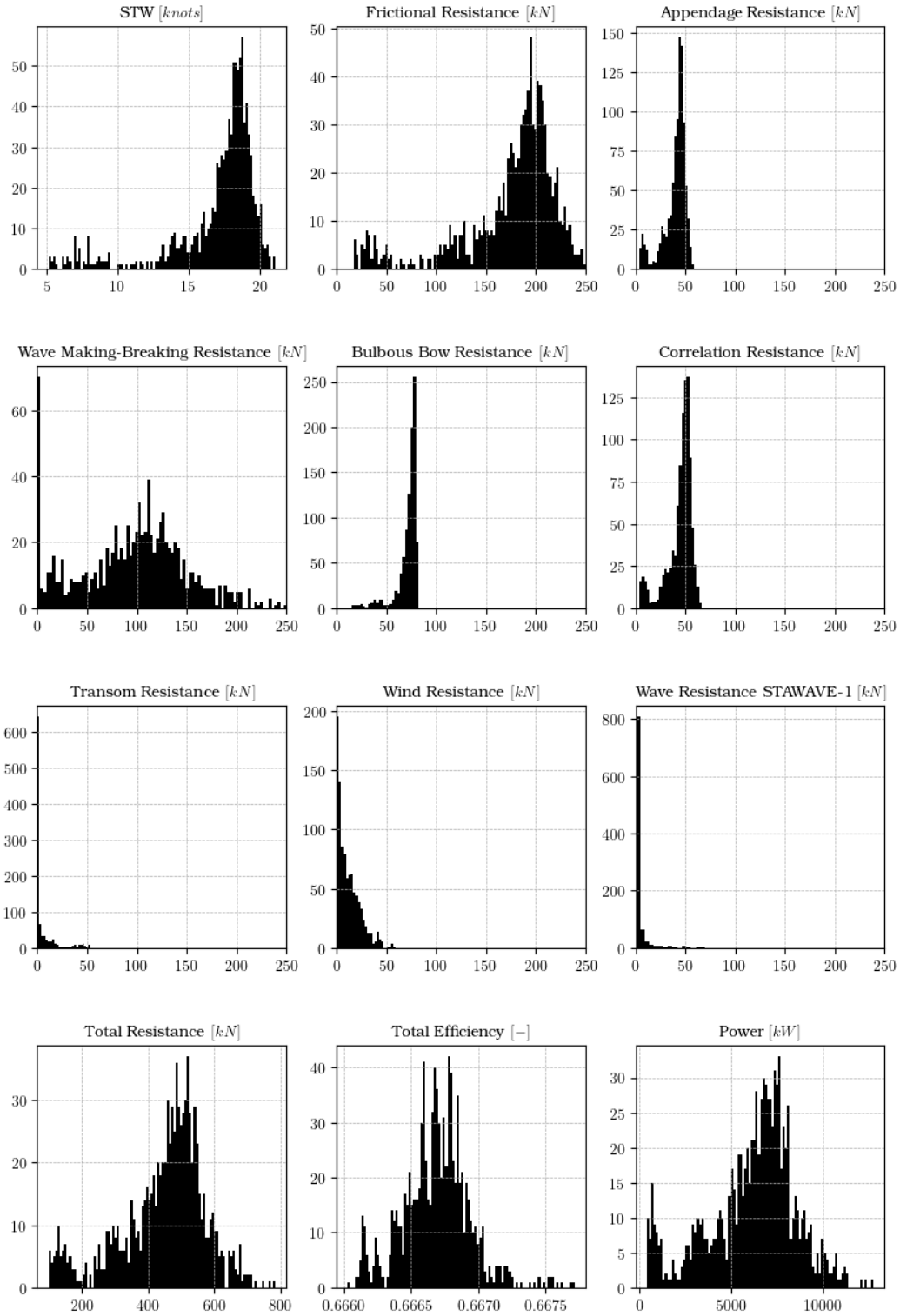

Figure 4.7: Histogram of encountered resistances for DS_{year}

To determine the most suitable regression fit, various polynomial regression lines with different orders are assessed. The selection of the optimal model is based on achieving the highest R^2 score while minimising the Mean Squared Error (MSE). To

Features		Count	Mean	std	Min	25%	50%	75%	Max
STW	$[kt]$	957.00	17.03	3.10	5.14	16.62	18.07	18.79	21.08
R_F	$[kN]$	957.00	174.65	49.25	17.17	162.08	189.16	205.18	262.25
R_{APP}	$[kN]$	957.00	39.52	11.29	3.64	36.53	43.03	46.46	58.25
R_W	$[kN]$	957.00	96.51	55.49	0.00	61.08	102.36	129.98	297.53
R_B	$[kN]$	957.00	71.23	11.30	15.59	69.79	74.93	77.64	82.20
R_{TR}	$[kN]$	957.00	5.58	11.60	0.00	0.00	0.00	4.70	53.56
R_A	$[kN]$	957.00	44.45	12.85	3.92	41.03	48.37	52.44	66.23
R_{AA}	$[kN]$	957.00	12.15	11.27	0.01	3.07	8.74	18.08	59.50
R_{AWL}	$[kN]$	957.00	3.49	11.23	0.00	0.00	0.00	1.17	116.18
R_{TOT}	$[kN]$	957.00	447.57	129.17	100.29	387.05	473.26	527.77	784.72
η_{TOT}	[%]	957.00	0.67	0.00	0.67	0.67	0.67	0.67	0.67
P_B	$[kW]$	957.00	6173.34	2361.10	397.02	4987.21	6607.35	7654.90	12755.90
FOC	$[T/h]$	957.00	1.04	0.40	0.06	0.84	1.11	1.29	2.09

Table 4.8: Descriptive statistics of power estimation method

ensure robust model selection, the data from each dataset is partitioned into training and validation sets, allowing thorough validation of model performance. The evaluation of regression fits reveals that a polynomial regression of order $n = 4$ is adequate to represent the relation between bunker consumption and STW. This model attains a R^2 score of approximately 99% and significantly reduces the MSE compared to the $n = 3$ model. Furthermore, as indicated in Table 4.9, there is no substantial performance enhancement when progressing from $n = 4$ to $n = 5$.

Dataset			Polynomial Order				
	Metric	Unit	1	2	3	4	5
DS_{year}	R^2	[%]	86.95	98.87	99.81	99.95	99.97
	MSE	$[(T/h)^2] \cdot 10^{-3}$	20.62	2.06	0.23	0.06	0.05
DS_{winter}	R^2	[%]	87.69	98.87	99.81	99.95	99.96
	MSE	$[(T/h)^2] \cdot 10^{-3}$	21.37	1.96	0.33	0.09	0.06
DS_{summer}	R^2	[%]	86.53	98.77	99.81	99.93	99.92
	MSE	$[(T/h)^2] \cdot 10^{-3}$	27.66	2.52	0.38	0.15	0.16

Table 4.9: Performance indices for FOC regression functions

The resulting bunker-to-speed curve plots are shown in figure Figure 4.8, The resulting functions align with the findings from **Psaraftis and Kontovas (2013)**. The cubic law dictates that the fuel consumption rate can be defined with a proportional factor and sailing speed raised to the power of $\alpha = 3$ **(Du et al. 2019)**. However, **Psaraftis and Kontovas (2013)** stated that the cubic law is not applicable for low speeds, and the factor α can be 4 or 5 or possibly even higher for some types of ships or when travelling at high speeds.

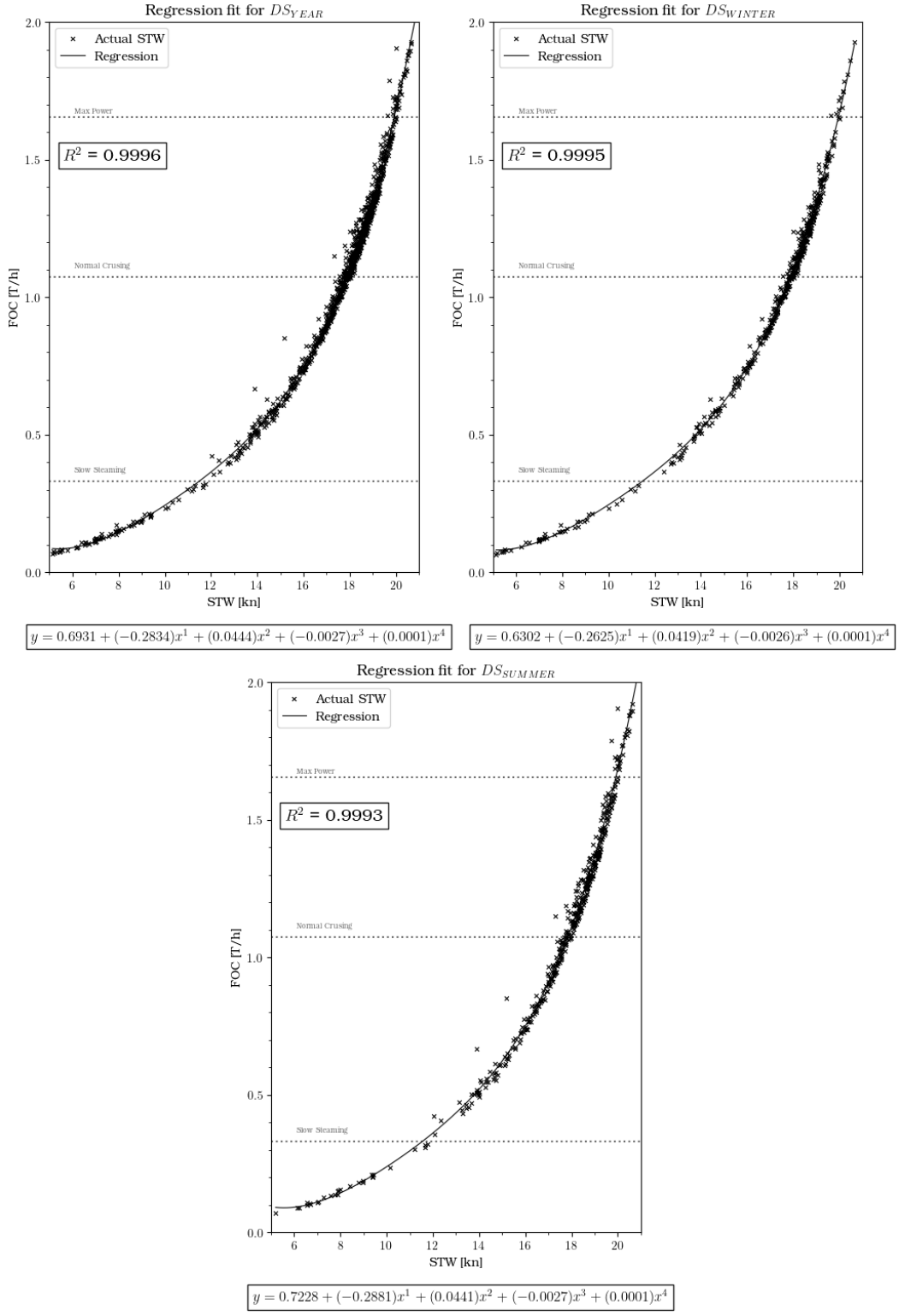

Figure 4.8: Bunker-to-speed curve for actual data

However, the observations outlined in Table 4.9 imply that the cubic law may still be applicable within the context of this case study. The slight improvement in function fit performance from $n = 3$ to $n = 4$ is not substantial. There is no notable reduction in the MSE, and the achieved R^2 score for $n = 3$ exhibits minor variation from the

best-fit model. Given that the model is primarily intended to characterize ship journeys at a steady sailing state, the limitations of the cubic law highlighted by Psaraftis and Kontovas (2013) may not be directly applicable.

The results also reveal instances where the model predicts ship speeds exceeding the maximum engine rating, which is physically impossible. This issue can be attributed to the conversion of SOG to STW. Analysing openly accessible data from the study conducted by **Petersen (2011)**[1], a noticeable distinction between SOG and STW can be observed, typically differing by a factor of approximately 0.85. In the context of this case study, even after accounting for the current correction, the difference in speeds remains marginal. This suggests that the conversion of SOG to STW should also include the influence of speed reduction due to wind and waves. **Molland, Turnock, and Hudson (2017)** presented illustrative speed loss curves based on varying Beaufort scales, as illustrated in Figure Figure 4.9. Furthermore, **Molland, Turnock, and Hudson (2017)** presented two direct speed loss formulae due to wind and wave effects by **Aertssen (1975)** and **Kwon (2008)**. However, these formulae possess a limited range of applicability and are not optimised for different vessel types. Therefore, these formulae are not implemented in this thesis.

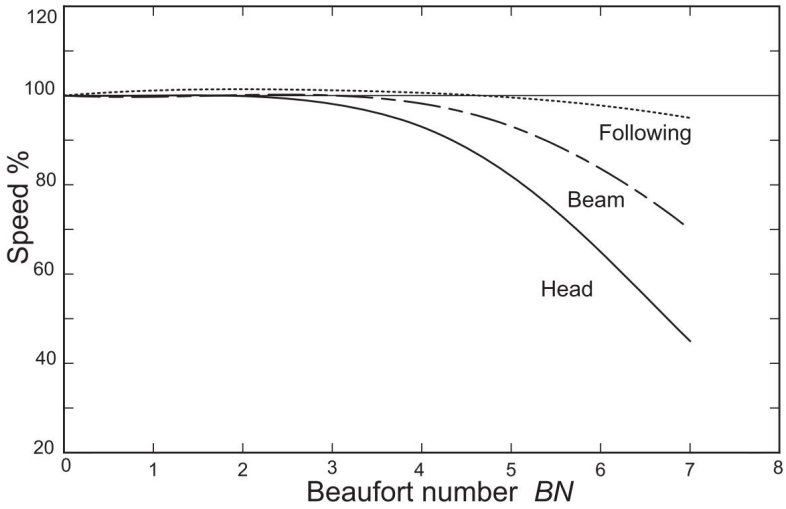

Figure 4.9: Speed loss with increase in Beaufort Number BN **(Molland, Turnock, and Hudson 2017)**

4.2.2 Result and Discussion of WBM on predicted SOG

Table 4.10 presents the results for FOC prediction across different modelling approaches and datasets. The ETR model demonstrated the best predictive performance, it is closely trailed by RFR and then followed by DTR. The similarity in results can be attributed to the sequential nature of the GBM approach. Notably, there is a decline in the R^2 score, explained variance, and MAPE across all models.

[1] http://cogsys.imm.dtu.dk/propulsionmodelling/

This decrease in performance could be attributed to the inherent nonlinearity of the WBM, which tends to amplify FOC prediction errors at higher ship speeds. This phenomenon is exemplified by the case study conducted by **Birk (2019)**, as depicted in Figure Figure 4.10. In this study, a discrepancy of 1 knot in ship speed resulted in a power difference of around 1200 kW at lower speeds. However, at higher speeds, the power difference escalated to about 2300 kW. Assuming an identical SFOC for the engine in both the **Birk (2019)** case study and this thesis, this discrepancy translates to FOC differences of approximately 0.2 T/h and 0.4 T/h, respectively.

Model	Dataset	R^2 [%]	expVar [%]	MAE $[T/h]$	RMSE $[T/h]$	MAD $[T/h]$	MAPE [%]
DTR_{OPT}	DS_{year}	76.58	76.63	0.133	0.193	0.087	15.78
	DS_{winter}	78.26	78.27	0.123	0.179	0.080	15.35
	DS_{summer}	74.42	74.72	0.144	0.208	0.097	16.17
RFR_{OPT}	DS_{year}	81.81	81.87	0.117	0.171	0.079	13.64
	DS_{winter}	86.57	86.58	0.099	0.141	0.068	12.06
	DS_{summer}	76.82	77.20	0.137	0.198	0.088	15.31
ETR_{OPT}	DS_{year}	83.83	84.00	0.111	0.161	0.076	12.45
	DS_{winter}	**87.58**	**87.58**	**0.097**	**0.135**	**0.067**	**11.35**
	DS_{summer}	79.86	80.49	0.127	0.185	0.082	13.59
MLR	DS_{year}	29.16	31.81	0.223	0.337	0.171	29.39
	DS_{winter}	10.39	11.37	0.212	0.363	0.161	33.19
	DS_{summer}	44.33	49.66	0.235	0.307	0.191	25.24

Table 4.10: Performance indices for FOC prediction

v_S [kn]	v_S [m/s]	Fr [−]	η_H [−]	η_O [−]	η_D [−]	n [1/s]	n [rpm]	P_D [kW]
15.0	7.717	0.2028	1.0976	0.5902	0.6440	1.883	112.990	4637.23
15.5	7.974	0.2095	1.0974	0.5884	0.6418	1.955	117.277	5208.37
16.0	8.231	0.2163	1.0971	0.5861	0.6392	2.028	121.700	5850.65
16.5	8.488	0.2230	1.0969	0.5835	0.6363	2.104	126.269	6573.49
17.0	8.746	0.2298	1.0967	0.5808	0.6331	2.182	130.949	7378.31
17.5	9.003	0.2365	1.0965	0.5780	0.6299	2.262	135.703	8263.98
18.0	9.260	0.2433	1.0963	0.5750	0.6265	2.342	140.542	9239.76
18.5	9.517	0.2501	1.0961	0.5717	0.6229	2.425	145.520	10328.13
19.0	9.774	0.2568	1.0959	0.5680	0.6187	2.512	150.747	11573.40

Figure 4.10: Case study for power estimation by **Birk (2019)**

This phenomenon could also account for the model's limitations in accurately predicting FOC at higher end of STW, as indicated by the plot in Figure Figure 4.11. The plots also demonstrated the underfitting behaviour of the models, Figure 4.11 shows that all models underpredict the FOC for STW higher than 20 knots. However, the model's prediction errors remain reasonable. For instance, considering the DS_{year} dataset, the MAE ranges between 0.111 T/h and 0.133 T/h, in the context of a mean FOC of 1.04 T/h.

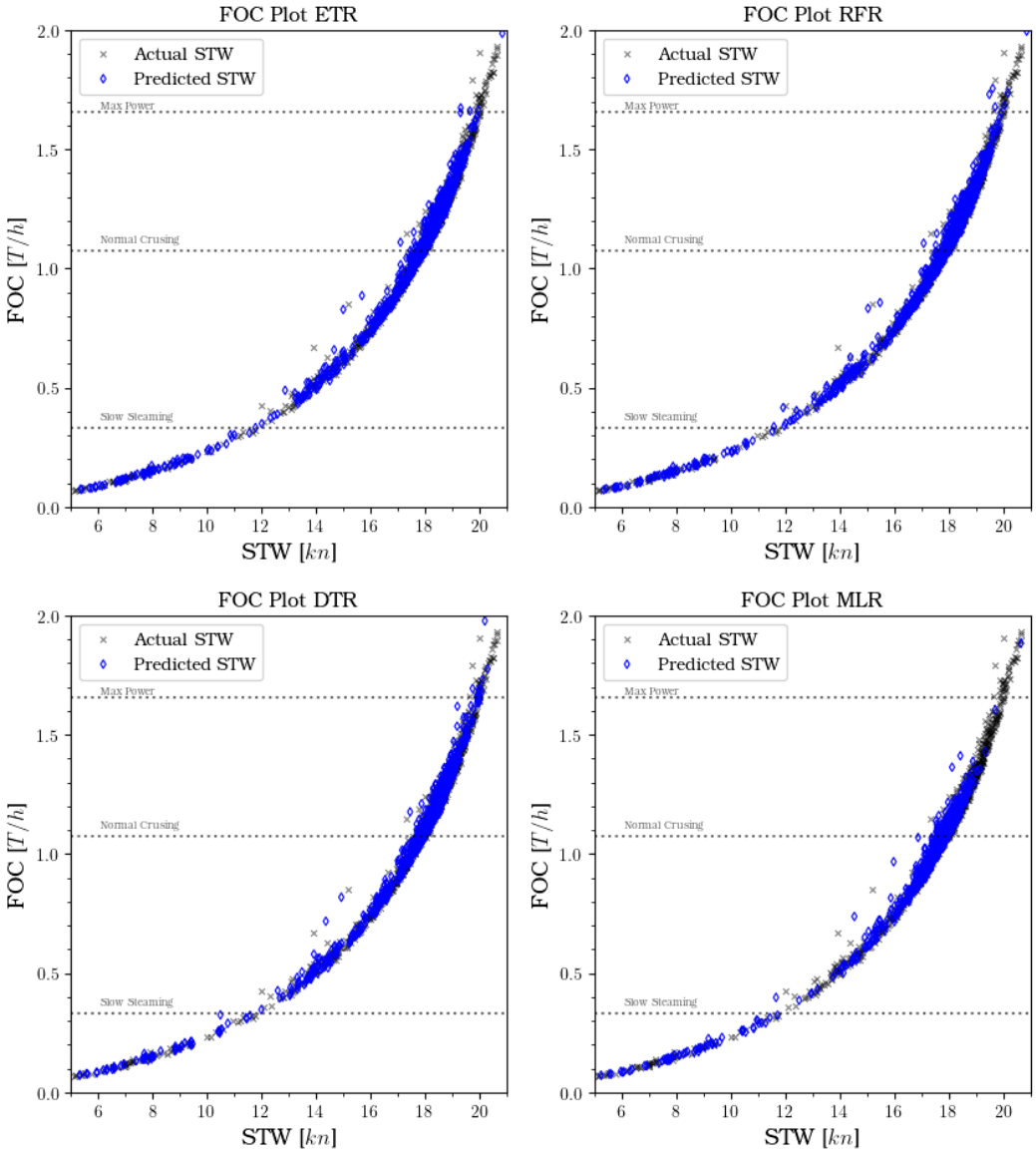

Figure 4.11: Predicted and Actual FOC for different models using DS_{year}

The resulting bunker-to-speed curves for different models are illustrated in Figure Figure 4.12 for ETR, Figure 4.13 for RFR, and Figure 4.14 for DTR. Overall, the plot for ETR exhibits the highest R^2 score. Nevertheless, the disparities are minimal, given that the functions are almost indistinguishable across different models.

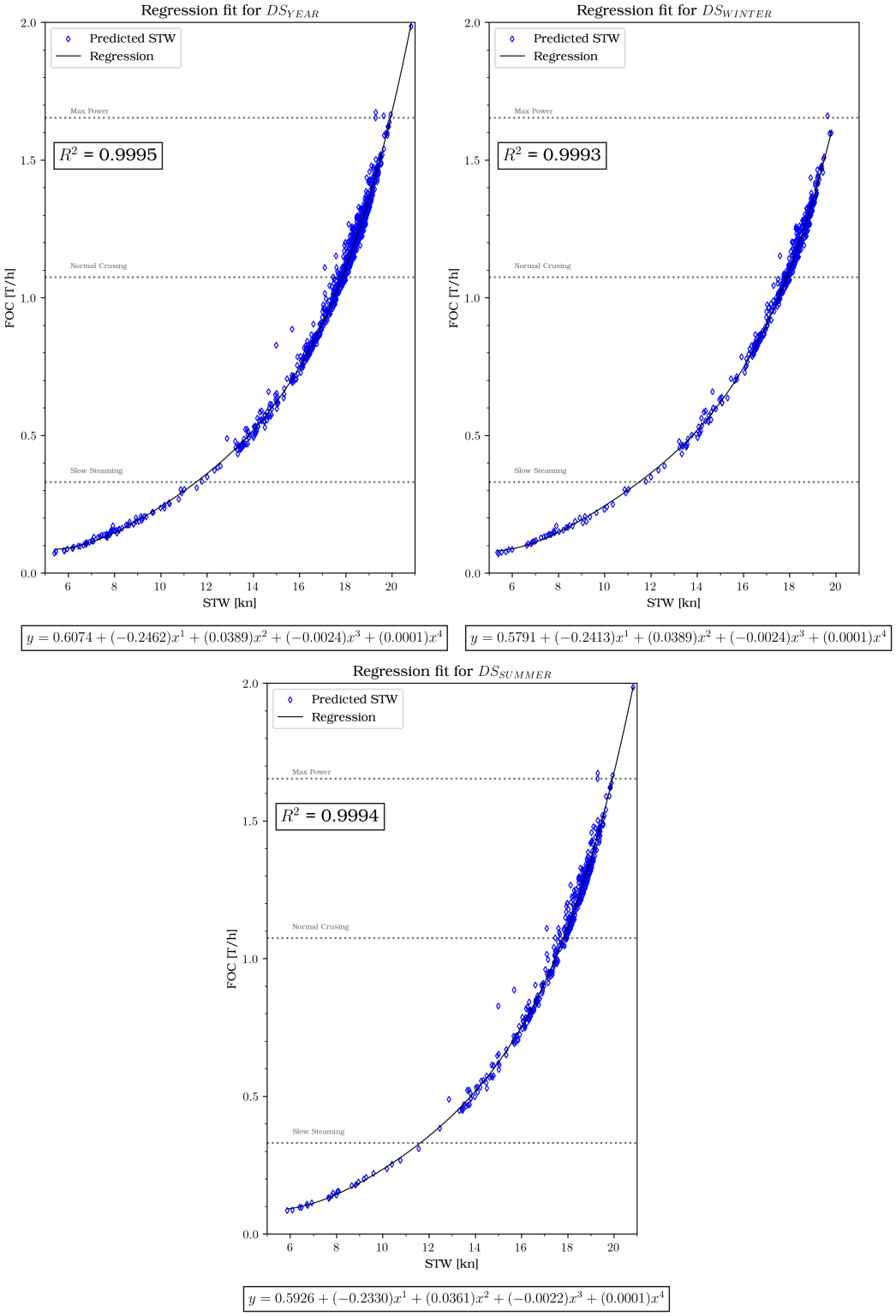

Figure 4.12: Bunker-to-speed curves generated from ETR

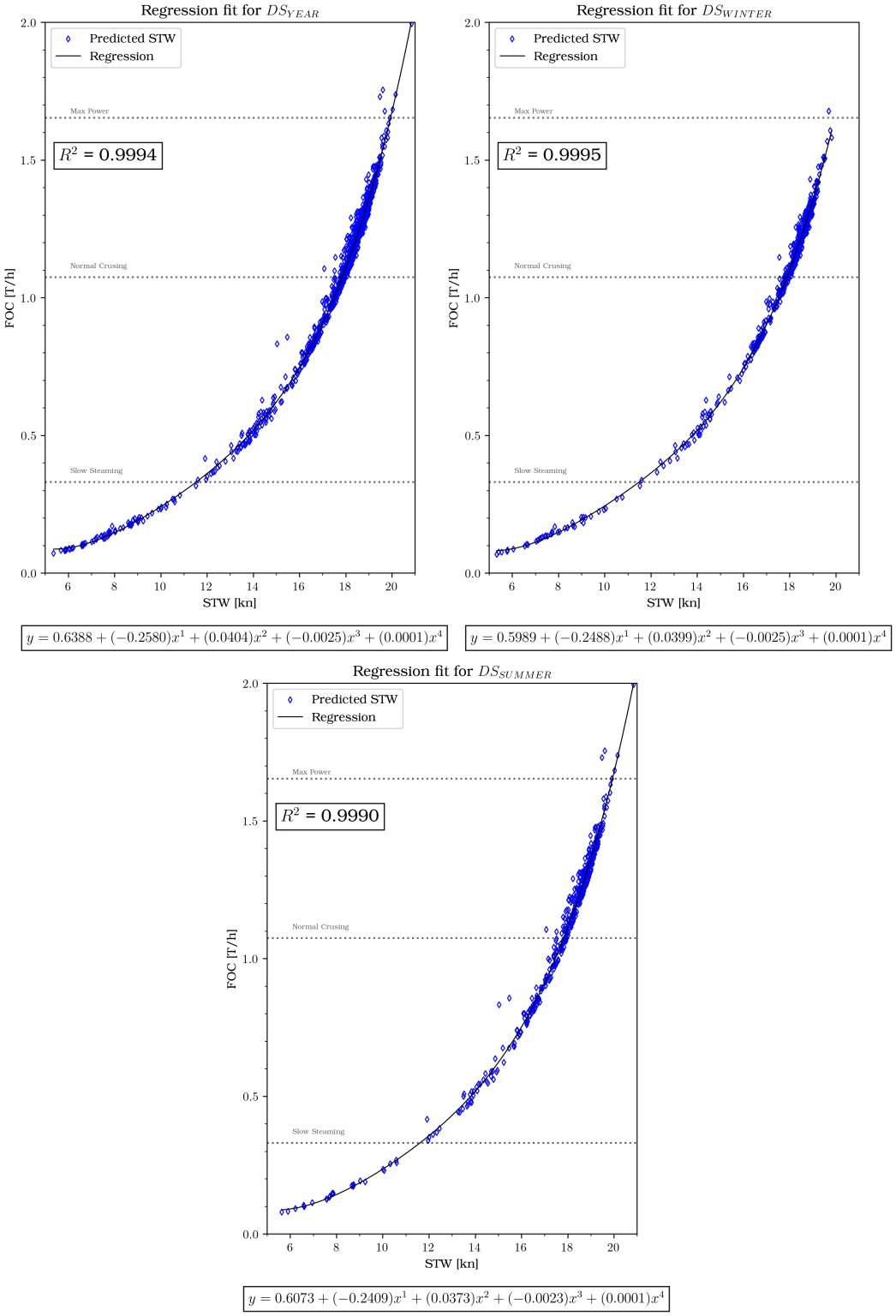

Figure 4.13: Bunker-to-speed curves generated from RFR

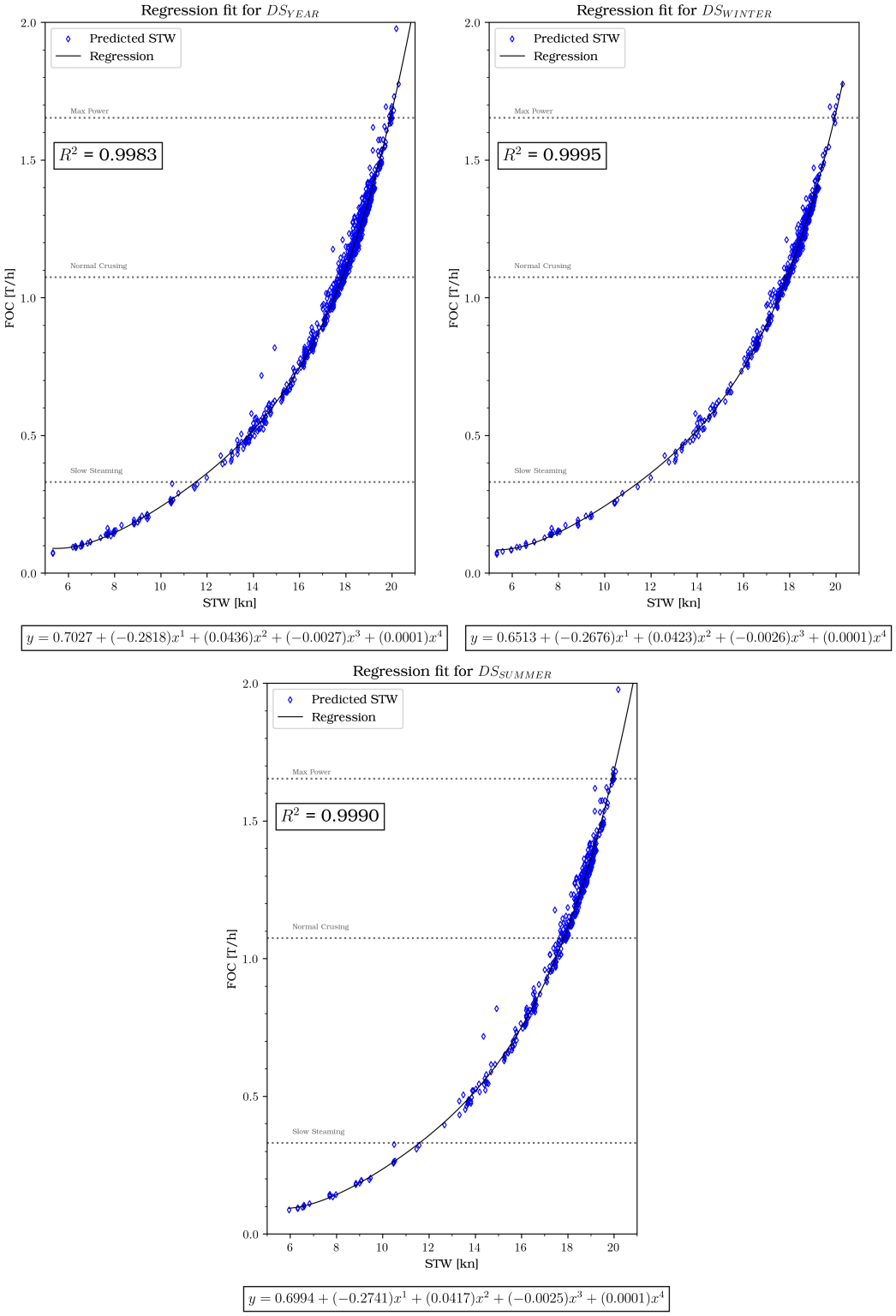

Figure 4.14: Bunker-to-speed curves generated from DTR

4.2.3 Key Findings

Performance of BBM

The best predictive performance for SOG from different datasets was obtained by ETR, which is able to achieve R^2 score of 95% and MAE of 0.53 knots. The performance gap between RFR and ETR is slight, RFR is able to obtain R^2 score of 93% and MAE of 0.55 knots. Despite being a relatively simpler model to both RFR and ETR, DTR performs reasonably well for SOG and FOC prediction, it obtained a R^2 score of 89% and MAE of 0.69 knots. A simple MLR model may be insufficient for implementation, as substantial performance gap is found between MLR and other tree-based models, with R^2 score of 72% and MAE of 1.16 knots.

The quality and quantity of data are closely interrelated factors that significantly impact model performance. There are noticeable differences in the performance between the DS_{winter} and DS_{summer} datasets, with an increase of approximately 6% in the R^2 score and a reduction of about 0.1 knots in MAE is achievable with an increase in about 50 data points. However, the quantity of data alone is insufficient to ensure an increase in model performance. This observation is evident from the DS_{year} datasets, which contain approximately twice as much data as the other datasets, yet the model's performance is not superior to that of the DS_{winter} dataset. This suggests that the quality of the DS_{summer} dataset may compromise the model's performance when using the DS_{year} dataset.

For BBM, hyperparameter optimisation substantially benefits the DTR model, while the impact of optimization on RFR and ETR models is minimal. Analysing feature importances and model structure visualisation, the RFR model recognizes draught as the most influential factor affecting SOG prediction. While the ETR and DTR models also identify this factor, but with a lesser degree of recognition.

Performance of WBM

Due to the sequential approach of GBM, the predictive performance of BBM is carried over to WBM during FOC prediction. Similar to BBM, ETR achieved a R^2 score of 88% with MAE of 0.097 T/h for the best dataset. This is then followed by RFR which achieved a R^2 score of 87% and MAE of 0.099 T/h, DTR obtained 78% for R^2 and 0.123 T/h for MAE. MLR sees a very significant decline in R^2 score, only achieving 44.28% and MAE of 0.34 T/h.

The nonlinear relation between speed and bunker means that significant deviation of SOG leads to an amplified magnitude of errors for FOC. This is evident in the case of MLR, the model already made relatively larger errors than other tree-based models during SOG prediction. With that, it is important to ensure that the SOG prediction of BBM is to be optimised as much as possible to ensure accurate FOC prediction. In the case study, the decline in performance is apparent when comparing the SOG prediction in Table 4.7 and Table 4.10.

The power estimation method by Holtrop-Mennen method resulted in a 4th order bunker-to-speed function. However, the resulting regression function of 3rd order shows comparable performance. Therefore, a relatively simpler model such as cubic law is a feasible replacement for the WBM part of the GBM. For calm water resistance R_{CALM}, all resistance components make noticeable contributions to calm water resistance R_{CALM} and the additional resistance due to wind and wave makes up about 3.5% of total resistance R_{TOT}.

Application of outlier rejection to test data

Two possible solutions to improve model performance was presented Section 4.1.3.2, since the addition of data points is not feasible, the application of outlier rejection is adopted in this section. The outlier in higher speed is not applicable, however, for the lower limit, it was found to be around 8 knots which are based on the mean and standard deviation of DS_{year} shown in Table 4.3. Due to the sparsity of data points at low SOG, the amount of data points is not compromised.

Model	Dataset	R^2 [%]	expVar [%]	MAE $[kn]$	RMSE $[kn]$	MAD $[kn]$	MAPE [%]
DTR_{OPT}	DS_{year}	79.54	79.55	0.658	0.971	0.462	3.99
	DS_{winter}	84.17	84.17	0.584	0.847	0.407	3.53
	DS_{summer}	74.26	74.36	0.742	1.096	0.524	4.51
RFR_{OPT}	DS_{year}	91.96	91.97	0.348	0.609	0.207	2.11
	DS_{winter}	95.33	95.33	0.304	0.460	0.201	1.83
	DS_{summer}	88.18	88.24	0.399	0.743	0.217	2.43
ETR_{OPT}	DS_{year}	92.60	92.62	0.363	0.584	0.234	2.19
	DS_{winter}	94.85	94.85	0.324	0.483	0.212	1.97
	DS_{summer}	90.01	90.12	0.409	0.683	0.253	2.46

Table 4.11: Performance indices for SOG prediction with outlier rejection

The results for SOG and FOC prediction are summarised in Table 4.11 and Table 4.12. In the case of both SOG and FOC prediction, there is an improvement observed in all aspects of the performance metrics for both RFR and ETR. While DTR displays a less accurate fit, it exhibits a decrease in prediction errors. Through the outlier rejection, it can be inferred that the model in this particular study is better suited for higher sailing speeds, given its training was performed on training datasets which are clustered at higher sailing speeds. Moreover, it is also observed that the RMSE notably reduces, which might be attributed to the removal of data points below the new threshold that were treated as outliers.

Model	Dataset	R^2 [%]	expVar [%]	MAE $[T/h]$	RMSE $[T/h]$	MAD $[T/h]$	MAPE [%]
DTR_{OPT}	DS_{year}	68.61	68.90	0.137	0.198	0.089	13.34
	DS_{winter}	72.18	72.24	0.118	0.170	0.078	11.76
	DS_{summer}	65.05	65.74	0.158	0.226	0.107	15.14
RFR_{OPT}	DS_{year}	88.41	88.60	0.072	0.120	0.042	7.05
	DS_{winter}	91.56	91.62	0.061	0.094	0.037	6.03
	DS_{summer}	85.58	85.97	0.086	0.145	0.044	8.23
ETR_{OPT}	DS_{year}	88.46	88.74	0.075	0.120	0.044	7.21
	DS_{winter}	90.64	90.73	0.065	0.099	0.041	6.45
	DS_{summer}	86.37	86.98	0.088	0.141	0.053	8.13

Table 4.12: Performance indices for FOC prediction with outlier rejection

Overall performance of GBM

Finally, it can be concluded that GBM is able to accurately predict SOG and FOC using a fusion of AIS data and weather data. This modelling approach benefits from the predictive power of BBM, which is utilised for SOG prediction. Subsequently, WBM facilitates the estimation of actual bunker consumption without neglecting fundamental vessel knowledge. However, due to the sequential model nature, it is crucial to ensure thorough data processing and model optimisation to minimise errors during initial SOG prediction. Given the nonlinear relationship between bunker consumption and speed, any initial prediction errors in BBM will be amplified.

Chapter 5

Summary and Outlook

5.1 Conclusion

This thesis introduces a comprehensive approach that combines data-driven techniques with empirical models to estimate FOC for a sailing vessel. The optimised machine learning model effectively forecasts FOC for vessels navigating at varying speeds, draughts, and weather conditions. The outcomes substantiate the viability of integrating AIS data and weather data for SOG prediction, which will then be used for FOC estimation. Technical details about the ship can be derived from AIS data. Along with suitable approximations from suitable and relevant literature, FOC can be forecasted using the empirical formulas proposed by Holtrop-Mennen. The results of predicted FOC can be used to generate bunker-to-fuel functions to estimate FOC for varying STW. The main findings and conclusion are presented in the following parts of this chapter.

Necessity of feature vorrelation in BBM

Machine learning-based FOC models rely heavily on feature engineering such as feature selection and feature importance identification. A prominent approach is the high correlation filter analysis, which involves the identification and elimination of highly correlated features. However, applying this filter necessitates careful consideration. Removing a feature should primarily be based on the understanding of physical and vessel-related knowledge.

Even though tree-based model inherently solves the problem of correlations between features and resist collinearity **(Yan, Wang, and Du 2020)**, feature selection might still be necessary to simplify the generated model and potentially enhance its performance. A more complex model with more features does not necessarily entails better performance and could be detrimental to computational cost. Additionally, it could be susceptible to endogeneity, defined as a correlation between the independent or observed variables in the model and the unobserved additive error term **(Danaf, Guevara, and Ben-Akiva 2023)**. Feature importance identification, which is an inherent benefit of tree-based model, serves as a valuable approach not only

for conducting post-training feature selection but also for verifying the model's alignment with the domain of physical and vessel-related knowledge thereafter.

Impact of data quantity, quality and resolution

As discussed in Section 4.2.3, the quality and volume of data are interrelated factors that greatly affect model performance. This should take precedence over hyperparameter optimisation. To put this into perspective, consider the instance of the STW distribution within the case study shown in Figure 4.7. Adding more data points in the higher speed range would not necessarily enhance the model's ability to predict FOC at lower speeds—an issue already evident in the case study. Addressing the challenge posed by AIS data, both in terms of its volume and quality, could potentially be tackled by exploring alternative data sources such as S-AIS. Unlike T-AIS, S-AIS circumvents the inherent range limitations and is particularly advantageous for scenarios involving ocean-going vessels.

Also, the hourly temporal resolution of AIS data, as opposed to noon data, presents a distinct advantage when predicting cases within narrower periods. The model trained in hourly resolution in this thesis proved to be effective in forecasting FOC within seasonal or yearly intervals. The influence of temporal resolution is also shown in the work of **Gkerekos, Lazakis, and Theotokatos (2019)**. For an equivalent volume of data, the noon data represents approximately 2.5 years' worth of information, whereas the sensor-based data, characterised by hourly resolution, corresponds to three months of information. The resulting assessments indicate that the model generated using sensor-based data exhibits fewer errors during predictions. For instance, considering the ETR model, the MAE for sensor-based data is calculated as 0.534 T/day, whereas for the noon data, the MAE is observed to be 1.434 T/day.

Power estimation method using Holtrop-Mennen method

The use Holtrop-Mennen method as power estimation method in this thesis has proven to be effective in estimating the energy required for operation. Missing input values that are not available can be approximated using formulas from different literature or estimated from similar case studies. However, the approximations are possible sources of errors and deviations for the estimation. If results from a towing tank resistance test are available, performing interpolation on the measurement values rather would be the preferable approach **(Lang 2020)**. Due nonlinearity of the power estimation method, it would be necessary to ensure the best possible accuracy and precision during the modelling of SOG or STW, especially if the vessel is sailing at high speed e.g. in the context of a merchant ship, the vessel sails at around 19 to 20 knots.

The minimisation of error terms for power estimation is crucial for scenarios such as Short Sea Shipping (SSS) which is defined as the maritime transport of goods and passengers by sea over enclosed seas **(van den Bos and Wiegmans 2018)**. Consider the following scenarios: an intra-regional journey of a feeder vessel that consumes 59.3 T of bunker across different legs in her journey **(Schøyen and Bråthen 2015)** and an ocean-going 8000 TEU vessel travelling from Yantian (YT) to Los Angeles (LA) which could consume up to 147 T/d of bunker **(Wang and Meng 2012)**. The effect of any error terms will be more significant for the SSS scenario. For identical total sailing distances, the prediction error from each journey in SSS accumulates until it reaches the same distance covered by an ocean-going vessel.

Strength and limitation of GBM approach for prediction of energy-efficient operation

The use of the Random Forest Regressor model as predictor for SOG of the Black Box Model has proven to be effective, slight performance improvement can be extracted when using Extra Trees Regressor. The approach requires minimal data preprocessing and minimal model configuration and the low variance in performance across different datasets showcased its robustness. The feature importance identification feature available to tree-based models provides implicit feature selection as well as an analysis tool to check whether the model obeys the physical domain knowledge of the vessel.

The White Box Model which incorporates the power estimation method by Holtrop-Mennen method further ensures that the energy estimation adheres to the physical principles and hydrodynamic laws of the vessel. The power estimation method can be used to plot bunker-to-speed functions to estimate the energy required for different operating speeds.

The combination with WBM diminishes the advantage of a Black Box Model which does not require any additional domain knowledge of the vessel. Additionally, the sequential approach will result in prediction errors that will be carried over during energy estimation.

5.2 Research outlook

Thus far, all the research questions outlined in the introduction have been addressed. Within the established research boundaries, measures have been undertaken to enhance model performance. The demonstrated efficiency of the model in predicting SOG and FOC demonstrated the viability of fusion between AIS and weather data. However, there remain prospects to further refine the proposed methodology.

Improvement to BBM

In this thesis, the bagging ensemble tree-based model is used for the BBM. However, there are other types of tree-based model which uses the boosting ensemble strategy, which trains decision trees in sequence and improves the performance of trees step by step using the information of fitting errors and negative gradient. Research by **Li et al. (2022)** demonstrates encouraging outcomes, suggesting that adopting this tree growth strategy could potentially enhance the model's performance.

Improvement of WBM

To the best of the author's knowledge, there is no existing research that offers a systematic conversion approach from SOG to STW. This conversion holds particular significance within this model, given that STW is a fundamental component for energy estimation. The methodology adopted in this thesis solely accounts for current as a factor for the SOG to STW conversion. However, in actuality, this conversion could be influenced by additional factors such as wind and wave effects, water depth, and potential hull fouling. Therefore, this is a potential research gap that may be pursued to further improve the energy estimation during vessel operation.

To further improve the accuracy of energy prediction, interpolation from measurements of towing resistance test **Lang (2020)**, or possibly calculated resistance from CFD simulations should be performed. While this may decrease the usage generalisability of the proposed methodology, the specificity of this method could potentially improve the accuracy of energy prediction and enable a closer simulation of real-world sailing conditions.

Bibliography

Abebe, Misganaw, Yongwoo Shin, Yoojeong Noh, Sangbong Lee, and Inwon Lee. 2020. "Machine Learning Approaches for Ship Speed Prediction towards Energy Efficient Shipping." *Applied Sciences* 10 (7): 2325. doi:10.3390/app10072325.

Aertssen, G. 1975. "The effect of weather on two classes of container ship in the North Atlantic." In *Naval Architect*, .

Bal Beşikçi, E., O. Arslan, O. Turan, and A. I. Ölçer. 2016. "An artificial neural network based decision support system for energy efficient ship operations." *Computers & Operations Research* 66: 393–401. doi:10.1016/j.cor.2015.04.004.

Bergstra, James, and Yoshua Bengio. 2012. "Random Search for Hyper-Parameter Optimization." *J. Mach. Learn. Res.* 13: 281–305.

Bertram, Volker. 2000. *Practical ship hydrodynamics*. Oxford: Elsevier Butterworth-Heinemann.

Bialystocki, Nicolas, and Dimitris Konovessis. 2016. "On the estimation of ship's fuel consumption and speed curve: A statistical approach." *Journal of Ocean Engineering and Science* 1 (2): 157–166. doi:10.1016/j.joes.2016.02.001.

Biran, Adrian, and Rubén López-Pulido. 2014. "Chapter 1 - Definitions, Principal Dimensions." In *Ship hydrostatics and stability*, edited by Adrian Biran, Rubén López-Pulido, and Javier de Juana Gamo, 1–21. Amsterdam: Butterworth-Heinemann. doi:10.1016/B978-0-08-098287-8.00001-3.

Birk, Lothar. 2019. *Fundamentals of ship hydrodynamics: Fluid mechanics, ship resistance and propulsion / Lothar Birk*. 1st ed. Hoboken, New Jersey: John Wiley & Sons. doi:10.1002/9781119191575.

Bitner-Gregersen, Elzbieta M. 2005. "Joint Probabilistic Description for Combined Seas." In *24th International Conference on Offshore Mechanics and Arctic Engineering: Volume 2*, 169–180. ASMEDC. doi:10.1115/OMAE2005-67382.

Blendermann, Werner. 1994. "Parameter identification of wind loads on ships." *Journal of Wind Engineering and Industrial Aerodynamics* 51 (3): 339–351. doi:10.1016/0167-6105(94)90067-1.

Breiman, Leo. 2001. "Random Forests." *Machine Learning* 45 (1): 5–32. doi:10.1023/A:1010933404324.

Breslin, John P., and Poul Andersen. 1994. *Hydrodynamics of ship propellers*. Vol. 3 of *Cambridge ocean technology series*. Cambridge: Cambridge University Press. doi:10.1017/CBO9780511624254.

Bretschneider, Charles L. 1965. *Generation of waves by wind. State of the art*. National Engineering Science Company Washington, DC.

Commons, Wikimedia. 2010. “Routen der Bornholmslinjen.” https://de.wikipedia.org/wiki/Datei:Bornholmerf%C3%A6rgen_route_map.svg.

Coraddu, Andrea, Luca Oneto, Francesco Baldi, and Davide Anguita. 2017. “Vessels fuel consumption forecast and trim optimisation: A data analytics perspective.” *Ocean Engineering* 130: 351–370. doi:10.1016/j.oceaneng.2016.11.058.

Danaf, Mazen, C. Angelo Guevara, and Moshe Ben-Akiva. 2023. “A control-function correction for endogeneity in random coefficients models: The case of choice-based recommender systems.” *Journal of Choice Modelling* 46: 100399. doi:10.1016/j.jocm.2022.100399.

Danish Maritime Authority. 2023. “Safety at Sea, Navigational Information, AIS Data.” Accessed 25/06/2023. https://dma.dk/safety-at-sea/navigational-information/ais-data.

Dietterich, Thomas G. 2000. “An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization.” *Machine Learning* 40 (2): 139–157. doi:10.1023/A:1007607513941.

Du, Yuquan, Qiang Meng, Shuaian Wang, and Haibo Kuang. 2019. “Two-phase optimal solutions for ship speed and trim optimization over a voyage using voyage report data.” *Transportation Research Part B: Methodological* 122: 88–114. doi:10.1016/j.trb.2019.02.004.

Géron, Aurélien. 2019. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems / Aurélien Géron*. Second edition ed. Sebastopol, CA: O’Reilly.

Geurts, Pierre, Damien Ernst, and Louis Wehenkel. 2006. “Extremely randomized trees.” *Machine Learning* 63 (1): 3–42. doi:10.1007/s10994-006-6226-1.

Gkerekos, Christos, Iraklis Lazakis, and Gerasimos Theotokatos. 2019. “Machine learning models for predicting ship main engine Fuel Oil Consumption: A comparative study.” *Ocean Engineering* 188: 106282. doi:10.1016/j.oceaneng.2019.106282.

Guldhammer, H. E., and S. A. Harvald. 1974. “Ship Resistance - Effect of form and principal dimensions. (Revised).” *Danish Technical Press, Danmark, Danmarks Tekniske Hojskole, kademisk Forlag, St. kannikestrade 8, DK 1169 Copenhagen* .

Halevy, Alon, Peter Norvig, and Fernando Pereira. 2009. “The Unreasonable Effectiveness of Data.” *IEEE Intelligent Systems* 24 (2): 8–12. doi:10.1109/MIS.2009.36.

Haranen, Michael, Pekka Pakkanen, Risto Kariranta, and Jouni Salo. 2016. “White, Grey and Black-Box Modelling in Ship Performance Evaluation.” .

Hastie, Trevor, Robert Tibshirani, and Jerome H Friedman. 2009. *The elements of statistical learning: Data mining, inference, and prediction*. 2nd ed., Springer series in statistics. New York: Springer. doi:10.1007/b94608.

Hollenbach, Uwe. 1999. “Estimating Resistance and Propulsion for Single-Screw and Twin-Screw Ships in the Preliminary Design.” In *10th international conference on computer applications in shipbuilding*, edited by Chryssostomos Chryssostomidis and Kaj. Ed Johansson, International conference on computer applications in shipbuilding, 237–250.

Holthuijsen, Leo H. 2007. *Waves in oceanic and coastal waters*. Cambridge: Cambridge University Press.

Holtrop, J. 1984. "A statistical re-analysis of resistance and propulsion data." *Published in International Shipbuilding Progress, ISP, Volume 31, Number 363* .

Holtrop, J., and G.G.J. Mennen. 1978. "A statistical power prediction method." *Netherlands Ship Model Basin, NSMB, Wageningen, Publication No. 603, Published in: International Shipbuilding Progress, ISP, Volume 25, Number 290, October 1978* .

Holtrop, J., and G.G.J. Mennen. 1982. "An approximate power prediction method." *Netherlands Ship Model Basin, NSMB, Wageningen, Publication No. 689, Published in: International Shipbuilding Progress, ISP, Volume 29, Nr 335, 1982* .

IMO. 2015. "Revised Guidelines for the Onboard Operational Use of Shipborne Automatic Identification Systems (AIS)." Accessed 25/06/2023. https://www.imo.org/en/OurWork/Safety/Pages/AIS.aspx.

IMO. 2020. "Fourth IMO GHG Study 2020." *International Maritime Organization London, UK* .

ITTC. 2014. "Analysis of Speed/Power Trial Data." *ITTC Recommended Procedures and Guidelines* 25–33.

Jensen, G. 1994. "Moderne Schiffslinien." *Handbuch der Werften* 22: 93.

Kam Ho, Tin. 1995. "Random decision forests." In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, 278–282 vol.1. doi:10.1109/ICDAR.1995.598994.

Kim, Seong-Hoon, Myung-Il Roh, Min-Jae Oh, Sung-Woo Park, and In-Il Kim. 2020. "Estimation of ship operational efficiency from AIS data using big data technology." *International Journal of Naval Architecture and Ocean Engineering* 12: 440–454. doi:10.1016/j.ijnaoe.2020.03.007.

Knudsen, Stig Staghøj. 2013. "Sail Shape Optimization with CFD." .

Kracht, Alfred M. 1978. "Design of Bulbous Bows." *Publication of: Society of Naval Architects and Marine Engineers* (Paper No. 7).

Kristensen, Hans Otto, and Marie Lützen. 2012. "Prediction of resistance and propulsion power of ships." *Clean Shipping Currents* 1 (6): 1–52.

Kuhn, Max, and Kjell Johnson. 2013. *Applied predictive modeling*. New York: Springer.

Kwon, Y. J. 2008. "Speed loss due to added resistance in wind and waves." *Nav Archit* 3: 14–16.

Lang, Xiao. 2020. "Development of Speed-power Performance Models for Ship Voyage Optimization." PhD diss.

Li, Xiaohe, Yuquan Du, Yanyu Chen, Son Nguyen, Wei Zhang, Alessandro Schönborn, and Zhuo Sun. 2022. "Data fusion and machine learning for ship fuel efficiency modeling: Part I – Voyage report data and meteorological data." *Communications in Transportation Research* 2: 100074. doi:10.1016/j.commtr.2022.100074.

MAN. 2011. "Basic principles of ship propulsion." *MAN Diesel & Turbo, Copenhagen* .

Molland, Anthony F. 2011. *The maritime engineering reference book: A guide to ship design, construction and operation / edited by Anthony F. Molland*. Butterworth-Heinemann. doi:10.1016/B978-0-7506-8987-8.X0001-7.

Molland, Anthony F., Stephen R. Turnock, and Dominic A. Hudson. 2017. *Ship resistance and propulsion*. 2nd ed. Cambridge: Cambridge University Press.

Montaño Moreno, Juan José, Alfonso Palmer Pol, Albert Sesé Abad, and Berta Cajal Blasco. 2013. "Using the R-MAPE index as a resistant measure of forecast accuracy." *Psicothema* 25 (4): 500–506. doi:10.7334/psicothema2013.23.

Nielsen, Ulrik D., and Jesper Dietz. 2020. "Ocean wave spectrum estimation using measured vessel motions from an in-service container ship." *Marine Structures* 69: 102682. doi:10.1016/j.marstruc.2019.102682.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, et al. 2011. "Scikit-learn: Machine Learning in Python." *Journal of Machine Learning Research* 12 (85): 2825–2830.

Petersen, Joan P., Ole Winther, and Daniel J. Jacobsen. 2012. "A Machine-Learning Approach to Predict Main Energy Consumption under Realistic Operational Conditions." *Ship Technology Research* 59 (1): 64–72. doi:10.1179/str.2012.59.1.007.

Petersen, Jóan Petur. 2011. "Mining of Ship Operation Data for Energy Conservation." .

Petersen, Jóan Petur, Daniel J. Jacobsen, and Ole Winther. 2012. "Statistical modelling for ship propulsion efficiency." *Journal of Marine Science and Technology* 17 (1): 30–39. doi:10.1007/s00773-011-0151-0.

Psaraftis, Harilaos N., and Christos A. Kontovas. 2013. "Speed models for energy-efficient maritime transportation: A taxonomy and survey." *Transportation Research Part C: Emerging Technologies* 26: 331–351. doi:10.1016/j.trc.2012.09.012.

Rakke, Stian Glomvik. 2016. "Ship emissions calculation from AIS." .

Ronen, D. 2011. "The effect of oil price on containership speed and fleet size." *Journal of the Operational Research Society* 62 (1): 211–216. doi:10.1057/jors.2009.169.

Schneekluth, H., and Volker Bertram. 1998. *Ship design for efficiency and economy*. 2nd ed. Oxford: Butterworth-Heinemann.

Schøyen, Halvor, and Svein Bråthen. 2015. "Measuring and improving operational energy efficiency in short sea container shipping." *Research in Transportation Business & Management* 17: 26–35. doi:10.1016/j.rtbm.2015.10.004.

Smith, Tristan, J. Jalkanen, B. Anderson, James Corbett, J. Faber, S. Hanayama, E. O'Keeffe, et al. 2015. "Third IMO Greenhouse Gas Study 2014." .

Soner, Omer, Emre Akyuz, and Metin Celik. 2018. "Use of tree based methods in ship performance monitoring under operating conditions." *Ocean Engineering* 166: 302–310. doi:10.1016/j.oceaneng.2018.07.061.

Stopford, Martin. 2008. *Maritime economics*. 3rd ed. Routledge.

Torsethaugen, Knut, and Sverre Haver. 2004. "Simplified Double Peak Spectral Model for Ocean Waves." In *ISOPE International Ocean and Polar Engineering Conference*, ISOPE–I. ISOPE.

van den Bos, Gertjan, and Bart Wiegmans. 2018. "Short sea shipping: a statistical analysis of influencing factors on SSS in European countries." *Journal of Shipping and Trade* 3 (1): 1–20. doi:10.1186/s41072-018-0032-3.

Wang, Shuaian, and Qiang Meng. 2012. “Sailing speed optimization for container ships in a liner shipping network.” *Transportation Research Part E: Logistics and Transportation Review* 48 (3): 701–714. doi:10.1016/j.tre.2011.12.003.

Wijnolst, N., Tor Wergeland, and Kai Levander. 2009. *Shipping Innovation*. IOS Press.

Yan, Ran, Shuaian Wang, and Yuquan Du. 2020. “Development of a two-stage ship fuel consumption prediction and reduction model for a dry bulk ship.” *Transportation Research Part E: Logistics and Transportation Review* 138: 101930. doi:10.1016/j.tre.2020.101930.

Yan, Ran, Shuaian Wang, and Harilaos N. Psaraftis. 2021. “Data analytics for fuel consumption management in maritime transportation: Status and perspectives.” *Transportation Research Part E: Logistics and Transportation Review* 155: 102489. doi:10.1016/j.tre.2021.102489.

Yang, Dong, Lingxiao Wu, Shuaian Wang, Haiying Jia, and Kevin X. Li. 2019. “How big data enriches maritime research – a critical review of Automatic Identification System (AIS) data applications.” *Transport Reviews* 39 (6): 755–773. doi:10.1080/01441647.2019.1649315.

Yang, Liqian, Gang Chen, Jinlou Zhao, and Niels Gorm Malý Rytter. 2020. “Ship Speed Optimization Considering Ocean Currents to Enhance Environmental Sustainability in Maritime Shipping.” *Sustainability* 12 (9): 3649. doi:10.3390/su12093649.

Appendix

Python Code

The code use in this thesis is developed using Python 3.9.15. The following code snippets highlight the most important part of the script. Full code is available at https://github.com/hiwafi/thesis-ais.git.

Package Loading

```
import pandas as pd
import numpy as np
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
import math
import datetime
import pickle
import joblib
import time
```

Loading Dataset

```
# Load the data to the script

dfmain = pd.read_csv("AIS_weather_H_ok2_copy.csv",parse_dates=["Time"])
dfmain = dfmain[dfmain['LAT'] > 55.04 ]


dfpre = pd.read_csv("AIS_weather_h_rename_copy.csv",parse_dates=["Time"])
dfpre = dfpre[dfpre['LAT'] > 55.04 ]
```

Splitting Datasets `KNNImputer`

```
from sklearn.model_selection import train_test_split
train_set, test_set = train_test_split(df, test_size=0.25, random_state=42)
```

Feature Selection

```
df_ship = df.drop(['Unnamed: 0','Time','LON','LAT','Air density above oceans',
'Surface pressure','Width','Length'],axis=1)

df_ship2 = df_ship.rename({'Max wave height': 'waveheight', 'Draught': 'draught',
```

```
                        'SOG': 'sog', 'Wind Speed': 'windspeed',
                        'True Wind Direction': 'truewinddir','Temperature above
                            oceans' : 'oceantemperature',
                        'COG': 'cog', 'Current Speed' : 'curspeed','True Wave
                            Direction' : 'truewavedir',
                         'Swell period': 'swellperiod','Wind wave period':
                             'windwaveperiod','Sea surface temperature': 'surftemp',
                         'Combined wind waves and swell height':
                             'windwaveswellheight','Swell height': 'swellheight','Wind
                             wave height': 'windwaveheight',
                         'Heading': 'heading','True Current Direction':
                             'truecurrentdir','True Swell Direction': 'trueswelldir',
                         'True Wind Wave Direction': 'truewindwavedir','Wave period':
                             'waveperiod',
                         'True North Wind Direction' : 'truenorthwinddir' , 'True
                             North Current Direction' : 'truenorthcurrentdir'
                        }, axis=1)

df_ship2 = df_ship2.drop(['waveheight','swellheight','windwaveheight',
                        'windwaveperiod','swellperiod',
                        'truewindwavedir','trueswelldir',
                        'truenorthcurrentdir','truenorthwinddir'],axis=1)
```

Imputing Dataset using `KNNImputer`

```
# Impute for training data

import numpy as np
from sklearn.impute import KNNImputer

imputer = KNNImputer(n_neighbors=50)
imputer.fit(df_ship2)

# Transform the imputed dataset

X = imputer.transform(df_ship2)

# Set column heading to make sure they have same name

df_ship2tr = pd.DataFrame(X, columns=df_ship2.columns, index=df_ship2.index)
```

Selecting training label and features

```
x_train = df_ship2tr.drop(['sog'],axis=1)
y_train = df_ship2tr.sog
```

Training optimised model

```
from sklearn.ensemble import RandomForestRegressor
model_rfr_ftr_hpov = RandomForestRegressor(n_estimators = 100, min_samples_split = 2
    ,min_samples_leaf = 1, max_features = 10, max_depth=120, random_state=42)
model_rfr_ftr_hpov.fit(x_train,y_train)

from sklearn.ensemble import ExtraTreesRegressor

model_etr_hpov = ExtraTreesRegressor(random_state=42 n_estimators=800,
    min_samples_split=9,min_samples_leaf=1, max_features=12, max_depth=120,)
model_etr_hpov.fit(x_train,y_train)

from sklearn.tree import DecisionTreeRegressor

```

```
model_dtr_hpov = DecisionTreeRegressor(min_samples_split=7,
    min_samples_leaf=10,max_features=12, max_depth=8)
model_dtr_hpov.fit(x_train,y_train)
```

Saving trained model

```
# # Saving the model to local directory

filename = 'savemodel_rfr_ftr_hpov.sav'
joblib.dump(model_rfr_ftr_hpov,filename)

filename = 'savemodel_etr_hpov.sav'
joblib.dump(model_etr_hpov,filename)

filename = 'savemodel_dtr_hpov.sav'
joblib.dump(model_dtr_hpov,filename)

filename = 'savemodel_mlr_ftr.sav'
joblib.dump(model_mlr,filename)
```

Hyperparameter Optimisation for RFR

```
from pprint import pprint
from sklearn.model_selection import RandomizedSearchCV
# Modify the search space of RFR here

# Number of trees in random forest
n_estimators = [100,200,300,400,500,600,700,800,900,1000]
# Number of features to consider at every split
max_features = [6,7,8,9,10,11,12]
# Maximum number of levels in tree
max_depth = [int(x) for x in np.linspace(10, 200, num = 20)]
max_depth.append(None)
# Minimum number of samples required to split a node
min_samples_split = [2, 5, 10]
# Minimum number of samples required at each leaf node
min_samples_leaf = [1, 2, 3,4,5,6,7,8,9,10]
# Method of selecting samples for training each tree
# bootstrap = [True]# Create the random grid
random_grid = {'n_estimators': n_estimators,
               'max_features': max_features,
               'max_depth': max_depth,
               'min_samples_split': min_samples_split,
               'min_samples_leaf': min_samples_leaf}
pprint(random_grid)

# Use the random grid to search for best hyperparameters
# First create the base model to tune
rf = RandomForestRegressor()
# Random search of parameters, using 3 fold cross validation,
# search across 100 different combinations, and use all available cores
rf_random = RandomizedSearchCV(estimator = model_rfr_ftr, param_distributions =
    random_grid, n_iter = 100, cv = 5, verbose=2, random_state=42,n_jobs=-1)# Fit the
    random search model
rf_random.fit(x_test, y_test)
```

Visualising trained tree

```
# Plot tree using graphviz, generate 1st tree in RFR (Graphviz must be installed in
    local computer)

```

```
from IPython.display import display
from sklearn import tree
import graphviz

dot_data_rfr = tree.export_graphviz(model_rfr_ftr_hpov.estimators_[1],
                  feature_names=x_train.columns.values.tolist(),
                #   class_names=class_names,
                  filled=True, rounded=True,
                  special_characters=True,
                   out_file=None,
                   max_depth=3,
                           )

display(graphviz.Source(dot_data_rfr))

graph = graphviz.Source(dot_data_rfr)
graph.format = 'png'
graph.render('rfr_mod_it1',view=True)
```

Cross-validation of model

```
def evaluate(model, features_x, labels_y):
    from sklearn.model_selection import cross_val_score

    score_r2 = cross_val_score(model,features_x,labels_y,
                        scoring='r2',cv=10)
    rsquared = score_r2.mean()
    stadev_rsquared = score_r2.std()
    max_rsquared = score_r2.max()
    min_rsquared = score_r2.min()

    score_expVar = cross_val_score(model,features_x,labels_y,
                        scoring='explained_variance',cv=10)
    expVar = score_expVar.mean()
    stadev_expVar = score_expVar.std()
    max_expVar = score_expVar.max()
    min_expVar = score_expVar.min()

    score_MAE = cross_val_score(model,features_x,labels_y,
                        scoring='neg_mean_absolute_error',cv=10)
    MAE = -score_MAE.mean()
    stadev_MAE = score_MAE.std()
    max_MAE = -score_MAE.max()
    min_MAE = -score_MAE.min()

    score_MAD = cross_val_score(model,features_x,labels_y,
                        scoring='neg_median_absolute_error',cv=10)
    MAD = -score_MAD.mean()
    stadev_MAD = score_MAD.std()
    max_MAD = -score_MAD.max()
    min_MAD = -score_MAD.min()



    score_MSE = cross_val_score(model,features_x,labels_y,
                        scoring='neg_root_mean_squared_error',cv=10)
    score_RMSE = np.sqrt(-score_MSE)
    RMSE = score_RMSE.mean()
    stadev_RMSE = score_RMSE.std()
    max_RMSE = score_RMSE.max()
    min_RMSE = score_RMSE.min()


    print(f"Model Performance of {model}")
    print(f"R^2 = {rsquared:0.4f}, std = {stadev_rsquared:0.4f}, max =
        {max_rsquared:0.4f}, min = {min_rsquared:0.4f}")
```

```
    print(f"explained Variance = {expVar:0.4f}, std = {stadev_expVar:0.4f}, max =
        {max_expVar:0.4f}, min = {min_expVar:0.4f}")
    print(f"MAE = {MAE:0.4f}, std = {stadev_MAE:0.4f}, max = {max_MAE:0.4f}, min =
        {min_MAE:0.4f}")
    print(f"RMSE = {RMSE:0.4f}, std = {stadev_RMSE:0.4f}, max = {max_RMSE:0.4f}, min
        = {min_RMSE:0.4f}")
    print(f"MAD = {MAD:0.4f}, std = {stadev_MAD:0.4f}, max = {max_MAD:0.4f}, min =
        {min_MAD:0.4f}\n")

    return score_r2,score_expVar,score_MAE,score_RMSE,score_MAD
```

Loading of trained model

```
import joblib

model_rfr_hpov = joblib.load('savemodel_rfr_ftr_hpov.sav')

model_etr_hpov = joblib.load('savemodel_etr_hpov.sav')

model_dtr_hpov = joblib.load('savemodel_dtr_hpov.sav')

model_mlr_ftr = joblib.load('savemodel_mlr_ftr.sav')
```

Evaluating predictive performance for SOG using testing data

```
def evaluate_SOG(model,x_date,y_date):
    from sklearn.metrics import
        mean_squared_error,mean_absolute_percentage_error,r2_score,explained_variance_score,median_abs

    def label_predict(model,test_features):
        predictions = model.predict(test_features)
        return predictions

    predictions = label_predict(model,x_date)

    Rsquared_SOG = r2_score(y_date,predictions)
    expVar_SOG = explained_variance_score(y_date,predictions)
    MAE_SOG = mean_absolute_error(y_date,predictions)
    RMSE_SOG = np.sqrt(mean_squared_error(y_date, predictions))
    MAD_SOG = median_absolute_error(y_date,predictions)
    MAPE_SOG = mean_absolute_percentage_error(y_date, predictions)


    print(f"Model Performance of {model}")
    print(f"R^2 SOG = {Rsquared_SOG:0.4f}")
    print(f"Explained Variance SOG = {expVar_SOG:0.4f}")
    print(f"MSE SOG = {MAE_SOG:0.4f} Knots")
    print(f"RMSE SOG = {RMSE_SOG:0.4f} Knots")
    print(f"MAD SOG = {MAD_SOG:0.4f} Knots")
    print(f"MAPE SOG = {MAPE_SOG*100:0.4f} %")
```

Function to convert SOG to STW

```
def sog_corr(sog,gamma,heading,current_speed):
    # Conversion of predicted SOG to m/s
    vgms = sog/1.9438
    rad_gamma = np.deg2rad(gamma)
    rad_cog = np.deg2rad(heading)
    # Calculation of the predicted x-component of SOG

    vgx = vgms * np.sin(rad_cog)
```

```
    vcx = current_speed * np.sin(rad_gamma)
    stw_x = vgx - vcx

    # Calculation of the predicted y-component of SOG

    vgy = vgms * np.cos(rad_cog)
    vcy = current_speed * rad_gamma
    stw_y = vgy - vcy

    vwms_p = np.sqrt(stw_x**2 + stw_y**2)
    stw_pred = vwms_p*1.9438

    return stw_pred
```

Power estimation function

```
def foc_fun(stw,T_dyn,windspeed,truewindir,H_s,truewavedir):
    # Ship Information, that are readily available in ship specification
    loa = 158 # ship overall length
    lwl = 144.8 # ship waterline length, m
    lpp = 0.97*lwl # ship perpendicular length , m, according to information
    B = 24.5 # Ship breadth, m
    depth = 13.8 # Ship depth. m
    T_n = 5.85 # Nominal max draught , m
    # T_n = 5.7 # Nominal design draught , m
    dwt = 5110 # ship dead weight , t
    V_n = 17.7 # ship design speed, knots
    # V_n = 18 # ship design speed, knots


    # Environmental Constants

    g = 9.805 # gravity, kg/ms^2
    rho_sea = 1025 # kg/m3
    nu_sea = 0.00000118 # Dynamic viscosity of sea m^2/s
    rho_air = 1.25 # density air

    # Any other additional ship parameters beyond here are approximated based on
        literature review.

    # Convert STW to m/s, stw with only current correction

    stw_ms = stw / 1.94384

    # Switch between actual and predicted here
    # Calculation for Block coefficient,C_b, according to Schneekluth and Bertram
        1998
    # Then Froude number is required

    V_n = 17.7/1.94384
    # V_n = 18/1.94384

    Fr_n = V_n / math.sqrt(g*lwl)

    C_b = -4.22 + 27.8*math.sqrt(Fr_n) - 39.1*Fr_n + 46.6*(Fr_n)**3

    # calculation for midship section coefficient, C_m according to Jensen from Birk

    C_m = 1 / (1+(1-C_b)**3.5)

    # prismatic coefficient C_p can be calculated according to Biran

    C_p = C_b/C_m

    # Displacement calculation according to Barras

    dsp = C_b * lwl * B * T_n
```

```
50
51      # coefficient c14 to account for stern shape according to holtrop mennen
52
53      C_stern = 10 # assume u shaped stern
54      c14 = 1 + 0.011*C_stern
55
56      # Calculate length of run according to holtrop mennen
57
58      # lcb = -2/100 # according to Barras
59      lcb = -(0.44*Fr_n - 0.094) # according to Guldhammer and Harvald
60
61      # L in holtrop mennen is lwl
62
63      lr = lwl*(1-C_p+(0.06*C_p*lcb/(4*C_p-1)))
64
65      # now the (1+k1) can be calculated
66
67      k1a = 0.487118*c14*(B/lwl)**1.06806
68      k1b = (T_dyn/lwl)**0.46106
69      k1c = (lwl/lr)**0.121563
70      k1d = (lwl**3/dsp)**0.36486
71      k1e = (1-C_p)**-0.604247
72
73      k1_const = 0.93 + k1a*k1b*k1c*k1d*k1e
74
75      # Calculate Reynold number and Coefficient of Friction C_f. Here, the C_f will
            be dynamic and depend on the velocity of the ship
76
77      Re =( stw_ms * lwl ) / nu_sea
78      C_f = 0.075 / (np.log10(Re-2)**2)
79
80      # Calculate the appendage area of bare hull S_bh
81      # Formula according to Holtrop Mennen
82
83      # Calculate the waterplane area coefficient
84      # Formula according to Schneekluth and Bertram
85
86      C_wp = (1+2*C_b)/3
87
88      # Calculate transverse bulb area A_bt, Transom area A_t and immersed midship
            section area A_m according to Kim 2019
89
90      # dfprog['A_m'] = B*dfprog['draught']*C_m
91      # Borrow estimation of Am from Guldahmmer and Harvald
92      A_m = dsp/(lpp*C_p)
93      A_t = 0.051 * A_m
94      A_bt = 0.085*A_m # From approximation of Kracht78, Similar to Charcalis
95
96      sbh_a = lwl*(2*T_dyn+B)*math.sqrt(C_m)
97      sbh_b = 0.453
98      sbh_c = 0.4425*C_b
99      sbh_d = 0.2862*C_m
100     sbh_e = 0.003467*(B/T_dyn)
101     sbh_f = 0.3696*C_wp
102     sbh_g = 2.38*A_bt/C_b
103
104     S_bh = sbh_a*(sbh_b+sbh_c-sbh_d+sbh_e+sbh_f)+sbh_g
105
106     # Calculate R_f
107
108     R_f = 0.5 * rho_sea * stw_ms**2 * C_f * S_bh * k1_const
109
110     # Calculate resistance due to appendage
111
112     # Assume S_app
113     # Taken from Holtrop Mennen worked example
114     # S_app = 50 # m^2
115
116     # Calculation of appendage area according to Hollenbach method, the formula is
            for twin screw ship
```

```
117
118     # Lower limit
119     S_app_lo = S_bh.mean()*(0.028+0.01*math.exp(-(lpp*T_n)/1000))
120
121     # Upper limit
122     S_app_hi = S_bh.mean()*(0.0325+0.045*math.exp(-(lpp*T_n)/1000))
123
124     # The following appendage area are scaled from the picture of the ship
125     # Constant k here means (1+k_2) !
126
127     D_shaft = 0.55 # m, approx
128     l_shaft = 13.54 # m, approx
129
130     S_app_shaft = 2*math.pi * D_shaft * l_shaft
131     k2_shaft = 3
132
133     h_rudder = 4.06 #m, approx
134     B_rudder = 1.99 #m, approx
135     S_app_rudder = 2 * h_rudder * B_rudder #m, two side
136     k2_rudder = 3
137
138     h_skeg = 4.41 #m, approx
139     l_skeg = 26.23 #m, approx
140     S_app_skeg =  h_skeg * l_skeg #two side (triangle)
141     k2_skeg = 1.5
142
143     S_app = S_app_shaft + S_app_rudder + S_app_skeg
144
145     k2_const = (k2_shaft*S_app_shaft + k2_rudder*S_app_rudder +
            k2_skeg*S_app_skeg)/S_app
146
147     # # from holtrop mennen, take case of twin screw
148     # k2_const = 2.8
149
150     # Add resistance due to Bow Thrusters
151
152     d_th = 2.15 #m, approx
153
154     # Use formula from Hollenach
155     C_dth = 0.003 + 0.003*((10*d_th/T_n)-1)
156     # C_dth = 0.003 # The picture shows that the thruster are fairly parallel to
            midship area
157     # There are two bow thruster in this ship
158     R_th = rho_sea*stw_ms**2*math.pi*d_th**2*C_dth
159
160     R_app = (0.5 * rho_sea * stw_ms**2 * C_f * S_app *k2_const) + 2*R_th
161
162     # Calculate wave-making and wave-breaking resistance
163
164     c7 = B/lwl
165     T_fwd = T_dyn # See reasoning from Rakke16
166     h_b = 0.6*T_n # must not exceed 0.6 T_f, here T_n = T_f (design), reasong and
            coefficient value taken from Rakke
167
168
169     # All formulas here are listed by Holtrop Mennen
170
171     c3 = 0.56 * A_bt**1.5 / (B*T_dyn*(0.31*np.sqrt(A_bt)+T_fwd-h_b))
172     c2 = np.exp(-1.89*np.sqrt(c3))
173     c5 = 1 - 0.8*(A_t/(B*T_dyn*C_m))
174     lambda_const = (1.446 * C_p) - 0.03*(lwl/B)
175     c16 = 8.07981*C_p - 13.8673*C_p**2 + 6.984388*C_p**3
176     m_1 = 0.0140407 * (lwl/T_dyn) - 1.75254*(dsp**(1/3)/lwl) -  4.79323*(B/lwl) - c16
177     c15 = -1.69385
178
179     # Use dynamic Froude here to refect the actual resistance due to ship movement
180
181     Fr_n_dyn = stw_ms / math.sqrt(g*lwl)
182     # Updated formula use m_4
183     m4 = 0.4 * c15 * np.exp(-0.034*Fr_n_dyn **-3.29)
```

```
184
185     i_e = 1 +
            89*math.exp(-(lwl/B)**0.80856*(1-C_wp)**0.30484*(1-C_p-0.0225*lcb)**0.6367*(lr/B)**0.34574*((1
186     c1 = 2223105 * c7**3.78613 * (T_dyn/B)**1.07961*(90-i_e)**-1.37565
187     d = -0.9
188
189     # Use updated formula with m4
190
191     R_w = c1*c2*c5*dsp*g*rho_sea*np.exp(m_1*Fr_n_dyn
            **d+m4*np.cos(lambda_const*Fr_n_dyn **-2))
192
193     P_b = 0.56*np.sqrt(A_bt)/(T_fwd-1.5*h_b)
194     Fn_i = stw_ms / np.sqrt(g*(T_fwd-h_b-0.25*np.sqrt(A_bt))+0.15*stw_ms**2)
195     R_b = 0.11 * np.exp(-3*P_b**-2)*Fn_i**3*A_bt**1.5*rho_sea*g/(1+Fn_i**2)
196
197     #Calculate Transom Resistance
198
199     Fn_tr = stw_ms / np.sqrt(2*g*A_t/(B+(B*C_wp)))
200
201     # Use condition to calculate Froude due to transom
202
203     cond_Fn_tr = [Fn_tr < 5 ]
204     cond_c6 = [0.2*(1-0.2*Fn_tr)]
205
206     c6 = np.select(cond_Fn_tr,cond_c6,0)
207     R_tr = 0.5*rho_sea*10**2*A_t*c6
208
209     # Model ship correlation resistance
210
211     cond_Tf_lwl = [(T_fwd/lwl) <= 0.04 ]
212     cond_c4 = [T_fwd/lwl]
213     c4 = np.select(cond_Tf_lwl,cond_c4,0.04)
214
215     C_a = 0.00546*(lwl+100)**-0.16 - 0.002 +
            0.003*math.sqrt(lwl/7.5)*C_b**4*c2*(0.04-c4)
216
217     R_a = 0.5*rho_sea*stw_ms**2*C_a*(S_bh+S_app)
218
219     # Calculate Additional Resistance, consist of wind resistance and wave resistance
220     # Calculate Apparent velocities and Apparent Angle
221
222     V_aw = np.sqrt(windspeed**2 + stw_ms**2 +
            2*windspeed*stw_ms*np.cos(np.deg2rad(truewindir)))
223
224     awa_c1 = (windspeed/V_aw)*np.sin(np.deg2rad(truewindir))
225
226     # Epsilon is Apparent Wind Angle AWA
227
228     epsilon = np.rad2deg(np.arcsin(awa_c1))
229
230     # Values and method from Blendermann
231
232     C_DlAf = 0.45
233     A_f = 325.3
234     A_l = 2125.8
235     C_Dt = 0.9
236     delta = 0.8
237     C_Dl = C_DlAf * A_f / A_l
238     L_bwl = 43.75 # m, acquired from picture
239
240     Raa_const1 = (rho_air/2) * V_aw**2 * A_l * C_Dl
241     Raa_const2 = np.cos(np.deg2rad(epsilon))
242     Raa_const3 = 1 - (delta/2) * ((1-(C_Dl/C_Dt))*(np.sin(np.deg2rad(2*epsilon)))**2)
243
244     R_aa = Raa_const1 * Raa_const2 / Raa_const3
245
246     # Calculate Wave Resistance according to STAWAVE-1
247
248     Rawl = (1/16) * rho_sea * g * H_s**2 * math.sqrt(B/L_bwl) * B
249
```

```
    condwave = [truewavedir<=45]
    choicewave = [Rawl]

    R_awl = np.select(condwave,choicewave,0)

    R_tot = (R_f + R_app + R_w + R_b + R_a + R_tr  + R_aa + R_awl)/1e3

    # Calculate Efficiencies

    # Diameter value for ship estimated from Bertram

    # D = 0.215*16 #m
    # Revised D, 08.07.23
    D = 4 # m, from flyer
    PD_const = 1.135  # From Bertram

    # Update C_v formula

    C_v = (k1_const*R_f + R_app + R_a) / (0.5*rho_sea*stw_ms**2*(S_bh+S_app))
    w = 0.3095 * C_b + 10*C_v*C_b - (0.23*D)/np.sqrt(B*T_dyn)
    t = 0.325*C_b - 0.1885*D/np.sqrt(B*T_dyn)
    eff_h = (1-t) / (1-w)
    eff_r = 0.9737 + 0.111*(C_p - 0.225*lcb) - 0.06325*PD_const
    eff_s = 0.99 # Set according to holtrop mennen and man
    eff_o = 0.7 # Approximation from Wageningen Line from Breslin94, since Holtrop
        perform their measurement in Wageningen basin

    eff_tot = eff_h* eff_r* eff_s*eff_o # consider sea margin

    # Calculate power and FOC

    P_b = (R_tot * stw_ms)/eff_tot # in kW
    SFOC = 169.4 # g/kWh, taken from datasheet Waertsilla 8V31
    FOC = (P_b * SFOC)/1e6 # get FOC t/h
    FOC_day = FOC * 11 #Per day 11 hour journey

    return R_f,R_app,R_w,R_b,R_tr,R_a,R_aa,R_awl,R_tot,eff_tot,P_b,FOC
```

Function for FOC regression fit

```
def poly_reg_best_fit(DataSet,STW,FOC):
    from sklearn.model_selection import train_test_split
    from sklearn.preprocessing import PolynomialFeatures
    from sklearn.linear_model import LinearRegression
    from sklearn.metrics import mean_squared_error
    from matplotlib.ticker import MultipleLocator,FixedLocator
    from matplotlib.transforms import ScaledTranslation

    plt.rcParams['figure.dpi'] = 300

    sorted_Xreg = np.sort(STW)
    sorted_Yreg = np.sort(FOC)

    Xreg = sorted_Xreg.reshape(-1,1)
    Yreg = sorted_Yreg

    Xreg_train, Xreg_test, Yreg_train, Yreg_test = train_test_split(Xreg, Yreg,
        test_size=0.25, random_state=42)

    train_errors = []
    test_errors = []
    coefficients_list = []
    scores_poly = []


    # Loop through different orders
    for order in range(1, 6):
```

```
        # Create polynomial features for the current order
        poly = PolynomialFeatures(degree=order)
        X_poly_train = poly.fit_transform(Xreg_train)
        X_poly_test = poly.transform(Xreg_test)

        # Fit the linear regression model
        model = LinearRegression()
        model.fit(X_poly_train, Yreg_train)

        # Make predictions on training and test data
        y_pred_train = model.predict(X_poly_train)
        y_pred_test = model.predict(X_poly_test)

        # Calculate the score (R-squared) of the model
        score = model.score(X_poly_test, Yreg_test)

        # Calculate mean squared errors for training and test data
        train_error = mean_squared_error(Yreg_train, y_pred_train)
        test_error = mean_squared_error(Yreg_test, y_pred_test)

        # Append the errors to the lists
        train_errors.append(train_error)
        test_errors.append(test_error)
        coefficients_list.append(model.coef_)
        scores_poly.append(score)
        # # Uncomment to get each order's performance
        # print(score)
        # print(test_error)

    # # Find the best model (lowest test error)

    # best_order = np.argmin(test_errors)

    # Brute force, seems that there is a bug with summer dataset actual dataset, in
        general order 4 is the most acceptable performance
    best_order = 4

    best_coefficients = coefficients_list[best_order]

    # Create polynomial features for the best model
    poly = PolynomialFeatures(degree=best_order)
    X_poly = poly.fit_transform(Xreg)

    # Fit the best model on the entire dataset
    best_model = LinearRegression()
    best_model.fit(X_poly, Yreg)

    # Get coefficients of the best model
    coefficients = best_model.coef_
    intercept = best_model.intercept_

    # # Print the polynomial equation
    # equation = "y = {:.4f}".format(best_model.intercept_)
    # for i, coef in enumerate(best_coefficients[1:], 1):
    #     equation += " + {:.4f}x^{}".format(coef, i)

    # print("Best Polynomial Equation:")
    # print(equation)

    # LaTeX format for polynomial equation
    def format_equation(coefficients, intercept):
        equation = f"$y = {intercept:.4f}"
        for i, coef in enumerate(coefficients[1:], 1):
            equation += f" + ({coef:.4f})x^{i}"
        equation += "$"
        return equation

    # Print the best polynomial equation
    equation = format_equation(coefficients, intercept)
    print("Best Polynomial Equation:")
```

```
    print(equation)
    # print(score.max())
    # get score for the 4th order
    Rsquared = scores_poly[3]

    # Generate points for plotting the best-fitted line
    X_plot = np.linspace(Xreg.min(), Xreg.max(), 100).reshape(-1, 1)
    X_plot_poly = poly.transform(X_plot)
    y_plot = best_model.predict(X_plot_poly)

    # Plot the original data points and the best-fitted line
    # Follow definition from 3rd GHG study
    slow_steam = 0.2*9760*(169.4/1e6)
    normal = 0.65*9760*(169.4/1e6)
    max_Pb = 9760*(169.4/1e6)

    # Actual Plot
    plt.scatter(STW,
        FOC,marker='d',linewidths=.8,facecolors='none',edgecolors='blue',label =
        'Predicted STW',s=12)
    # # # Temporary plot for actual STW
    # plt.scatter(STW, FOC,marker='x',linewidths=.8,color='black', label = 'Actual
        STW',s=12)
    plt.plot(X_plot, y_plot, color='black',label='Regression',linewidth=.8)
    plt.title(f'Regression fit for ${DataSet}$')
    plt.xlabel('STW [kn]')
    plt.ylabel('FOC [T/h]')
    # plt.xticks(range(6, 22, 1))
    plt.xlim(5,21)
    plt.ylim(0,2)
    plt.yticks([i/10 for i in range(21)])
    # Show only the values at every 0.5 interval on the y-axis
    ax = plt.gca()
    ax.yaxis.set_major_locator(MultipleLocator(base=0.5))
    # Show minor ticks at every 0.1 interval on the y-axis
    ax.yaxis.set_minor_locator(FixedLocator([i/10 for i in range(1, 20)]))
    # Show minor ticks at every 1 interval on the x-axis
    ax.xaxis.set_minor_locator(MultipleLocator(base=1))
    plt.axhline(y=slow_steam,linestyle = 'dotted',c='k',alpha=0.6)
    plt.axhline(y=normal,linestyle = 'dotted',c='k',alpha=0.6)
    plt.axhline(y=max_Pb,linestyle = 'dotted',c='k',alpha=.6)

    plt.text(6.1,1.01*max_Pb,'Max Power',rotation=360,alpha=.6,fontsize=6)
    plt.text(6.1,1.1*slow_steam,'Slow Steaming',rotation=360,alpha=0.6,fontsize=6)
    plt.text(6.1,1.03*normal,'Normal Crusing',rotation=360,alpha=0.6,fontsize=6)
    plt.text(4.2, -.25, equation, bbox=dict(facecolor='white',
        alpha=0.9),fontsize=12)
    plt.text(5.6, 1.5, rf'$R^2$ = {Rsquared:0.4f}', bbox=dict(facecolor='white',
        alpha=0.9),fontsize=14)

    # plt.grid(linestyle = '--', linewidth = 0.25,which='both')
    plt.legend(loc='upper left')
    # plt.show()

    return best_model
```

Evaluation of prediction performance of SOG using testing data

```
def evaluate_FOC(model,FOC_act,FOC_pred):
    from sklearn.metrics import
        mean_squared_error,mean_absolute_percentage_error,r2_score,explained_variance_score,median_abs

    Rsquared_FOC = r2_score(FOC_act,FOC_pred)
    expVar_FOC = explained_variance_score(FOC_act,FOC_pred)
    MAE_FOC = mean_absolute_error(FOC_act,FOC_pred)
    RMSE_FOC = np.sqrt(mean_squared_error(FOC_act, FOC_pred))
    MAD_FOC = median_absolute_error(FOC_act,FOC_pred)
```

```
    MAPE_FOC = mean_absolute_percentage_error(FOC_act, FOC_pred)

    print(f"Model Performance of {model}")
    print(f"R^2 {Rsquared_FOC:0.4f}")
    print(f"Explained Variance {expVar_FOC:0.4f}")
    print(f"MAE {MAE_FOC:0.4f} T/h")
    print(f"RMSE FOC {RMSE_FOC:0.4f} T/h")
    print(f"MAD {MAD_FOC:0.4f} T/h")
    print(f"MAPE FOC {MAPE_FOC*100:0.4f} %")
```

Declaration in lieu of oath

I hereby solemnly declare that I have independently completed this work or, in the case of group work, the part of the work that I have marked accordingly. I have not made use of the unauthorised assistance of third parties. Furthermore, I have used only the stated sources or aids and I have referenced all statements (particularly quotations) that I have adopted from the sources I have used verbatim or in essence.
I declare that the version of the work I have submitted in digital form is identical to the printed copies submitted.
I am aware that, in the case of an examination offence, the relevant assessment will be marked as 'insufficient' (5.0). In addition, an examination offence may be punishable as an administrative offence (Ordnungswidrigkeit) with a fine of up to €50,000. In cases of multiple or otherwise serious examination offences, I may also be removed from the register of students.
I am aware that the examiner and/or the Examination Board may use relevant software or other electronic aids in order to establish an examination offence has occurred
I solemnly declare that I have made the previous statements to the best of my knowledge and belief and that these statements are true and I have not concealed anything.
I am aware of the potential punishments for a false declaration in lieu of oath and in particular of the penalties set out in Sections 156 and 161 of the German Criminal Code (Strafgesetzbuch; StGB), which I have been specifically referred to.

Section 156 False declaration in lieu of an oath
Whoever falsely makes a declaration in lieu of an oath before an authority which is competent to administer such declarations or falsely testifies whilst referring to such a declaration incurs a penalty of imprisonment for a term not exceeding three years or a fine.

Section 161 Negligent false oath; negligent false declaration in lieu of oath
(1) Whoever commits one of the offences referred to in Sections 154 to 156 by negligence incurs a penalty of imprisonment for a term not exceeding one year or a fine. (2) No penalty is incurred if the offender corrects the false statement in time. The provisions of Section 158 (2) and (3) apply accordingly.

Place,date

Signature