Vessel Performance Evaluation Using A Digital Twin

A Black-box Model Approach

Developing various black-boxes based on supervised machine learning algorithms in order to evaluate voyage performance of a vessel based on motions and operational features

R. De Vocht

Vessel Performance Evaluation Using A Digital Twin

A Black-box Model Approach

by

R. De Vocht

to obtain the degree of Master of Science
at the Delft University of Technology,

Student number: 4922913
Project duration: February 24, 2021 – October 7 2021
Thesis committee:

dr. ir. I. Akkerman,	TU Delft, supervisor
dr. ir. Yusong Pang,	TU Delft
ir. R. Campe,	CMB
ir. N. De Bruyn,	CMB

This thesis is confidential and cannot be made public until December 31, 2024.

TU Delft

Preface

After writing this master thesis in a challenging time, during the Covid-19 pandemic, this is the right place to thank the persons who supported me throughout the process.

First of all I must say thank you to dr.ir. Ido Akkerman, my supervisor from TU Delft. Whenever I was in doubt or required guidance he provided me with a good and swift response. I am also glad he gave me the opportunity to do research on my own chosen subject with him as supervisor.

The chosen subject came out of a discussion with CMB, which is represented by ir. Roy Campe and ir. Nathan De Bruyn who were also my supervisors from the company side. Roy, thank you for providing me the opportunity for this specific subject and make use of a CMB owned vessel and your enourmous practical knowledge. Nathan, thank you very much for standing by my side during the research even if you were very busy in South-Korea. No border or time difference could stop you for helping me out.

Also to my parents, Caroline Van Rompuy and Ludwig De Vocht, I cannot say thank you enough. They gave me the chance to study my second master at TU Delft and be the person who I am today. It was also a great support my mother worked from home during the pandemic. At noon we always went for a walk ”through the forest” such that we could get fresh air during the day.

At last but certainly not least, I thank my girlfriend, Jasmine Heyvaert for supporting me. She was always there for me and was always ready to listen when I could not stop talking about my thesis, even though she did not fully understand the technical aspects of the subject as she is not a maritime expert. Also thank you for being patience whenever we should had time together but in stead I was discussing my thesis with your brother Wouter Heyvaert. With him I was able to discuss some technical and programming related issues. Thank you for that too, Wouter.

R. De Vocht
Antwerp, September 2021

Abstract

An important step in determining the performance of a commercial ship is to determine the vessel fuel consumption in off-design conditions. In previous studies, the vessel fuel consumption is obtained using machine learning algorithms which use navigational data and meteorological data to train the models. Due to the inaccuracy of the weather hindcasts, the accuracy of these machine learning techniques is not satisfactory for Compagnie Maritime Belge (CMB) which is the company for which this study is performed. Since the vessel motions are a direct result of the meteorological conditions an approach is investigated in this research to replace the meteorological data by motion data to train a machine learning algorithm and obtain accurate results. The vessel motions are captured by a motion sensor on board the vessel.

Multiple machine learning techniques were tested in this research. The data used to train the best performing learning algorithm represent the motions by the first and second moment of the non-derivative motion and the second moment of the first derived motion. These moments are obtained from the motion distributions of a time window of 5 minutes. The obtained best parameters for representing the motions were obtained by testing if a spectrum must be provided, how many motion derivatives are required, which moments are needed and how large the time window must be.

Although the accuracy of the developed learning algorithm is determined as satisfied no digital twin could be made and tested. Unfortunately, due to the limited amount of data available no separate algorithm could be made which uses the Extra Tree learning, as digital twin base, to determine the over-consumption of the vessel. Only data from one voyage was available. All this data was used to train the learning algorithm to have as much variance as possible in the data set such that a good learning algorithm could be developed.

Contents

1 Introduction

In the ongoing economic environment, many of the currently used vessels were designed in the ”boom years” [32] of shipping. In those years, the fuel prices were low. This is not the case anymore. Many ship operators and designers have initialized projects in order to reduce the fuel consumption of ships. In the past, ships were designed for a calm sea, one design speed and one draft. At the time this was a good method but in reality many ships sail in off-design conditions. Meaning that the vessel does not operate in the condition she was designed for. Often the ship sails at different draft. Also ship motions occur in off-design conditions. This leads to a bigger fuel cost. When the fuel prices were low, this was no issue. Now that the fuel prices are higher and the International Maritime Organisation (IMO) has accepted strong environmental rules, such as the Energy Efficiency Design Index (EEDI), Energy Efficiency Operational Index (EEOI) and Ship Energy Efficiency Management Plan (SEEMP), interest became higher in operating ships in off-design conditions. Therefore, analyzing ship performance under different off-design conditions is necessary such that a better understanding is created. The understanding is needed in order to make sure that the requirements of the adopted rules are met. Based on the ship performance, the vessel owner can take actions to increase the overall performance and reduce the required energy of the vessel. One company that initialized a project to monitor the performance of their owned vessels in off-design conditions is Compagnie Maritime Belge (CMB). CMB is a company which owns a large number of vessels of various types.

1.1. Problem Statement

CMB already made big steps in monitoring the performance of their vessels considering fuel consumption. At the moment they receive a stream of information from the owned vessels. This information contains operational and environmental data. From the current data, CMB is already capable of measuring the performance of the vessels but CMB wants to go one step beyond. Currently the vessel performance in off-design conditions is quantified by means of empirical formulas. These provide correction factors in order to correct the performance of the vessel in off-design condition to design condition. CMB now wants to use machine learning techniques to determine the performance of their owned vessels directly in off-design conditions, which is also the purpose of this research. With machine learning a more accurate performance evaluation can be performed. This leads to a faster intervention when vessel performance is not as satisfied. An intervention can be a cleaning of the ship hull, a repair or removing fouling from propeller or rudder.

Due to environmental conditions (wind, waves, swell) a vessel consumes more fuel in off-design conditions compared to a vessel sailing in calm weather conditions, which are on-design conditions when the vessel is just delivered by the shipyard. The relation between the wind or waves and the vessel fuel consumption has already been determined [7]. The same counts for the coupling of trim and vessel consumption [16]. The individual effects of various factors that occur in off-design conditions are possible to model in computational fluid dynamics (CFD). This computation method provides good insights in the couplings between individual circumstances and the vessel fuel consumption or required engine power. Unfortunately, CFD is not capable to model a combination of different effects that influence the

vessel's consumption because of computer power restrictions. The hull fouling, the trim, power plant tuning, the swell or the current will all have their effects on the ship. An overview of all the influencing factors is given in figure 1.1. To model all these influences on a full-scale ship, considering different loading and environment conditions is nearly impossible using CFD. This inconvenience has led to a search for new methods.

The modern method to determine the fuel consumption of a vessel in off-design conditions is a black-box using machine learning algorithms. This method uses gathered data from operational vessels in order to find a set of rules and relations that can describe the fuel consumption behaviour whilst taking into account all the environmental and operational effects.

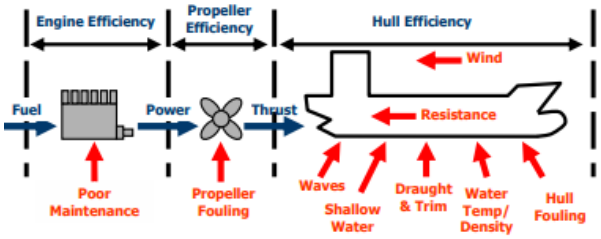

Figure 1.1: Influencing factors to fuel consumption [24]

1.2. Proposed Solution

In this research, different machine learning algorithms will be developed to determine the fuel consumption of the vessel by using vessel motion data instead of inaccurate environmental hindcasts. Using vessel motion data as data source instead of weather hindcasts differentiate this research from the published literature. Since motion data is a direct result of the weather conditions it is highly useful to use as replacement of environmental conditions. Additionally, vessel motion data is of high precision because the exact circumstances at the ships positions are known. This precision is not reached when weather hindcasts are used. As such the data capturing can be performed completely independently and not dependent on third party weather organisations.

The best machine learning method is used to make a black-box digital twin of the in real-life clean (no hull or appendage fouling) vessel. The results for the fuel consumption obtained for the clean digital twin are compared to the fuel consumption of the real operational vessel of a certain voyage. The comparison leads to a quantification of the performance of the operational vessel in off-design condition, thus subjected to motions and hull/appendage fouling, which is called over-consumption. The algorithm will be developed for an operational container vessel which crosses the Atlantic ocean every 2-3 weeks. It is believed this route is beneficial to the research because of its long ocean voyage where different environmental conditions and thus different vessel motion situations will occur.

1.3. Content Overview

In chapter 2 an insight will be provided why a vessel consumes more fuel while sailing in off-design conditions and thus also in motion. Next, an overview of the already performed research about machine learning methods to predict vessel fuel consumption is given in chapter 3 together with a brief explanation of the most common algorithms and evaluation methods. Afterwards in chapter 4, the knowledge gap is determined and explained why the investigation performed in this research is useful. In chapter 5, the used research method is explained together with a plan of approach. Chapter 6 covers how the pre-processing of the captured data is performed. Next, in chapter 7 multiple machine learning algorithms are developed and analysed. At last, the best algorithm is improved in chapter 8 to create the most accurate digital twin.

I

Literature Review

2

Fuel Consumption

The fuel consumption of a ship depends on the power needed to propel itself. This power is directly related to the resistance of the ship via the vessel speed. Therefore, an increase in resistance will lead to a greater fuel consumption. The required power in design conditions can be estimated by the method proposed by Holltrop and Mennen [14]. This is a statistical approach that uses data of full-scale vessels to predict required engine power needed in calm water. On top of the resistance the ship experiences when sailing in calm water, it will experience additional increase in resistance due to a seaway, which can be as much as 15-30% [26], and can directly be translated into an increase in wave making resistance due to vessel motions and wave reflection. The energy used for making waves, is energy that cannot be used to propel the ship [18]. The total wave resistance consists of three parts. The reflection or dissipation, ship motions and waves due to forward speed. These different parts are discussed in this chapter.

Next to wave resistance, frictional and air resistance also have an influence on the fuel consumption. The frictional resistance is influenced by viscosity, density, fouling, trim and draft. An increase in resistance is also observed due to air resistance, the wind. Other components such as current or power plant tuning will not have a direct influence on the resistance of the vessel but will influence the fuel consumption in a certain time span. An overview of the factors influencing the fuel in a seaway is given in figure 2.1 and are explained in following sub-chapters. As such an understanding is created under which circumstances fuel consumption will increase in a certain seaway. The form resistance is not taken into account because this resistance will always be more or less the same if a seaway is present.

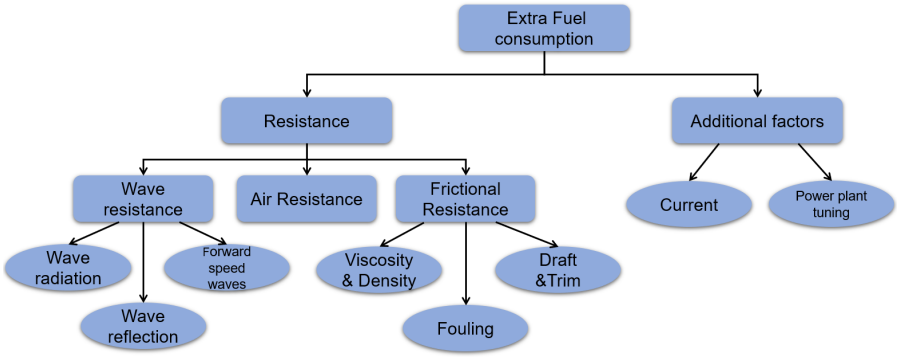

Figure 2.1: Summary of fuel consumption increasing factors

2.1. Resistance Increase

In this section all the resistance increasing factors of the vessel are explained. Due to an increase in resistance, the required power to propel the ship at a certain speed is also increased. An increased

power leads to an increase in fuel consumption. Therefore the goal is to minimize the resistance of the ship which will lead to a better performance of the vessel. First, the wave resistance factors are explained after which the frictional resistance and at last the air resistance is discussed as well.

2.1.1. Wave Resistance

The wave resistance is directly related to the energy used by the ship to make waves. These waves can be divided in three different wave making mechanisms. All three are explained below.

Forward Speed Waves

When the vessel is navigating, the water around the hull needs to be displaced. This displacement results in waves which are created by the ship. The energy used to produce these waves is coming from the power plant from the ship and cannot be used to propel the ship. The size of the produced waves and thus the lost energy is highly correlated with the vessel's speed.

Wave Reflection

Waves that are already present on the water surface due to wind or swell will collide with the vessel's hull and are always reflected which dissipates energy from the ship. How large the reflection and thus energy loss is, greatly depend on the relative wave angle. Head waves will dissipates the most energy from the system and stern waves dissipate the least. The energy dissipation is energy that cannot be translated into forward speed. Thus, the energy loss due to reflection can be categorized as an extra resistance, also called added resistance, which leads to an additional fuel consumption [26].

Wave Radiation

As it is generally known, a vessel reacts on waves by motions. These vessel motions are highly dependent on the position of the center of gravity and water plane area. Because of the motions of the vessel, waves are generated. These waves are called radiation waves. The vertical motions, heave and pitch in particular, of a ship have the greatest influence on these radiated waves [26]. Generation of these radiated waves extracts energy from the systems. This is energy that must be produced by the power plant of the vessel and which is not useful to propel the ship. Due to the radiation of waves, fuel consumption is increased.

2.1.2. Frictional Resistance

Most of the external factors influencing the resistance can be categorized as factors that influence the frictional resistance. The friction between the vessel hull and the water results in a force, the friction force. The factors influencing this frictional force are explained in this chapter.

Fouling

Fouling is the presence of algae, animals or plants on the wetted surface of the ship [9]. Those increase the roughness of the hull, which leads to a higher frictional resistance. Therefore the ship experiences extra resistance when fouling is present. This resistance will again result in an additional fuel consumption to remain at the same speed.

Draft & Trim

The resistance of the vessel is dependent on the frictional forces between the water and the hull. If the draft of the vessel is large, a larger area is covered with water and thus a larger area where the frictional force can work on is present. In other words, the wetted surface of a ship is larger with a large draft, which results into a large frictional force.

Due to hydrodynamic effects each draft has its own optimal trim [16]. Therefore, by a simple trim change the fuel consumption of a ship can be optimized. This is a big research subject which will not be discussed in this research as it is out of scope.

Viscosity & Density

The viscosity is highly correlated with the frictional resistance. A more viscous fluid will lead to a higher force needed to propel through the fluid. This viscosity is strongly dependent of the water temperature. A low temperature results in a fluid with a higher density and a higher viscosity. As a consequence more power is needed to propel the ship. An increase in fuel consumption is thus a direct result.

2.1.3. Air Resistance

The density of air is much smaller than the density of water. Therefore the forces created by the air resistance will be much smaller. Nevertheless, the air resistance cannot be ignored.

Most of the time when a vessel is navigating, wind is present. The wind effects on the superstructure are important when the vessel has a large structure above the waterline [26]. For container ships these effects are more pronounced than for a loaded bulkcarrier. A container vessel has most of the time a large area above the water where the wind can work on. This is not the case for a loaded bulkcarrier. The height of the structure of a loaded bulkcarrier is negligible compared to the height of the structure underwater. Therefore, wind effects on ships with a small area above the waterline are very small.

In the research of Wang [35] it is proven that also the direction of the wind on the superstructure is of great importance. Head winds have the greatest effect of the forward speed of the vessel. Wind coming from a beam of the vessel has a great influence on the drift force. What happens in between these extreme wind positions is difficult to predict [26].

The force created by the wind on the vessel, also called the air resistance, must be compensated by the power plant of the vessel in order to remain at constant velocity. The compensation for the wind force working on the vessel results in an increased fuel consumption.

2.2. Additional Factors

Various factors have a direct link to the resistance of the ship. Although there are also factors that do not have a direct link to the ship resistance but will influence the fuel consumption of the ship. These are discussed here.

2.2.1. Current

In most of the oceans and seas, current is present. This current can be beneficial for the fuel consumption or not. When the current is against the sailing direction of the ship, it must navigate at a higher speed through water to get the same speed over ground compared to when no current is present. A higher fuel consumption is obtained in that way. Nevertheless, when a current is coming from behind the ship, it is advantageous, since the ship can sail at a lower speed through water while achieving a similar or higher speed over ground. In this way fuel is saved. This shows that current is a very important factor in determining fuel consumption.

2.2.2. Power Plant Tuning

The power plant of a vessel is a complex mechanism with many different moving parts and different heat transfer systems between different machines. In this complex total system timings, temperature and moving parts must work together perfectly to get the most optimal fuel consumption. In case one of these parameters is off, a potential higher fuel consumption is likely to occur.

2.3. Combination Of Mechanisms

In literature most investigations into added resistance were performed by individually investigating the effects of waves or wind on the added resistance. Very few studies are done considering a combination of both. One study that takes into account both mechanisms is the research of Wang [35]. This research showed that the added resistance due to wind cannot be added to the added resistance due to waves. In figure 2.2 it is shown that the resistance with the combined wind speed and incoming waves is greater than the sum of both. There are three reasons for the difference in added and combined wave and wind resistance.

- The fluctuating wind fields apply an unsteady pitching torque on the ship which increases the pitch and heave motions as well as the resistance [35].
- The generated waves in the simulation are affected by the wind field. Therefore a change in wave parameters occurs which leads to a higher impact on the bow [35].

- Wave radiation in the combined load case is more severe, resulting in more energy dispersing from the system [35].

The research of Wang [35] was only performed for two wave lengths, one wind speed and wind direction, but it already indicates that the resistance increases with a longer wave length. A hypothesis is that there is a limit to the added resistance at a certain higher wave length. This limit is probably when the large wave length causes a less severe pitching motion. In this way the wind will not have a big effect on the deck of the vessel. This statement must be verified.

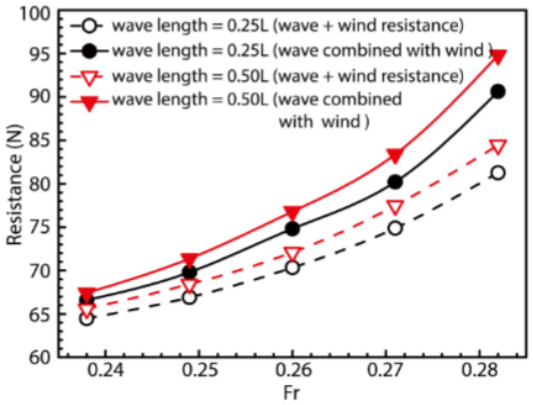

Figure 2.2: Comparison between added and combined resistance of wind and waves [35]

Very few is known about the resistance and thus power increase needed in a real environment on full-scale vessels where the factors explained above are taken into account. More investigation must be performed on the various effects in off-design conditions because the named factors will all influence the performance of the vessel [32].

Clearly a knowledge gap exists. Modeling a combination of various effects described above in a computer model is a very tedious and time consuming job. Therefore using modeling software and Computational Fluid Dynamics (CFD) will not fulfill the needs anymore. Alternative techniques must be used in order to gain insight in the fuel consumption of a ship which is susceptible of seaway when sailing in off-design conditions. The solution was found in literature by using machine learning algorithms that can predict the power and thus the fuel consumption needed to sail in a seaway. The algorithms use data of full-scale real operational vessels which are susceptible to different circumstances from day to day. As such the full picture is taken into account and no numerous CFD calculations must be performed for different circumstances. All the factors that will influence the resistance of a vessel are covered while using machine learning techniques if correct data is used. A good realistic prediction can then be performed on how much a vessel will consume in off-design conditions. The already existing machine learning solutions to fill the before mentioned gap are explained in next chapter.

3

Scientific Research

The proposed method in order to get further insight into vessel performance due to a combination of vessel features, meteorological conditions and operational features is machine learning. Machine learning is the science of programming computers so they can learn from data [4]. This relatively new computer method gained interest because it is extremely powerful. When data about the ship itself, operational features and meteorological conditions are available, this method provides a solution in order to gain insight in ship performance under different environmental and operational circumstances. In reality a big amount of data is given to a machine learning method. The method then starts to generate rules and equations for relations between different features to output a desired value. These rules and equations will eventually be true for the biggest part of the full data set. In this way an estimate can be provided when a new combination of encountered values are inserted in the learning method. The process where the machine learning finds these relations and rules is called “training” of the method.

In machine learning three major methods can be distinguished, supervised learning, unsupervised learning and reinforced learning. With supervised learning the data that is fed to the computer also contains the desired solution. This is not the case for data that is fed to unsupervised machine learning algorithms. The data in unsupervised learning is unlabeled. Reinforced learning is very different, this method can observe the environment and perform actions. These actions are punished if they are wrong or rewarded when they are correct [4], such that the model will learn the ideal actions.

In this research only supervised learning is considered since the used data is labeled. This chapter provides an overview of the already performed research with supervised learning in order to obtain ship performance. This provides the information about which research has already been performed and where knowledge gaps are still present. Also performance indicators for the machine learning algorithms will be clarified. An explanation of a digital twin is also provided. The model of digital twin is often used in combination with machine learning and will also be of further use in this research.

3.1. Supervised Machine Learning Methods

The supervised learning methods can be divided into six accurate methods which are being investigated in literature for determining predictions of ship required power or fuel consumption. The six methods are called Artificial Neural Networks, Support Vector Regression, Random Forest, Linear regression, Ridge Regression and LASSO regression. The latter three methods are the most simple algorithms and are often used as benchmark for performance of more advanced methods. All methods are presented in this section, together with their performance according to literature. This is useful information in order to reproduce these methods for the performed research in this thesis.

3.1.1. Linear regression

Linear regression is a parametric model and lies at the basis of the more enhanced methods. A parametric model is defined as a model that uses a finite set of parameters that were obtained by training the method. Because of the high non-linearity between different input features relevant for estimating

ship performance, Abebe et al. [2] concluded that the multiple linear regression (MLR) method provided inaccurate results. Nevertheless in the research of Uyank [33] it is proven that multiple linear regression can be useful since it gave as accurate results as more advanced methods. Thus, a strong contradiction in literature exist. The multiple linear regression is the same as linear regression method with that difference that with MLR more input features are taken into account.

The mathematical formulation of a linear regression method is as follows:

$$y(\boldsymbol{x}, \boldsymbol{w}) = w_0 + \sum_{j=1}^{D} w_j x_j \tag{3.1}$$

D is equal to the number of input features. If D is greater than one, the method is called multiple linear regression (MLR) [13]. w_j represents weight factors for each input parameter x_j and can be estimated by using least squares shown in equation below and y is the target value. The weight factor estimation happens during training where constantly the weight factors will be adapted in order to achieve the most accurate results.

$$\hat{w} = \arg\min_{w} \left(\sum_{i=1}^{N} \left(y_i - w_0 - \sum_{j=1}^{D} \left(w_j x_{ij} \right) \right)^2 \right) \tag{3.2}$$

Where N are the number of samples used while training the model.

3.1.2. Ridge & LASSO regression

Ridge regression, also a parametric model, is based on MLR but instead of only using the weight factor w_j for x_{ij}, Ridge regression adds an additional penalty factor, λ, on the square of each parameter w_j. The least squares method then becomes [13]:

$$\hat{w}_{RR} = \arg\min_{w} \left(\sum_{i=1}^{N} \left(y_i - w_0 - \sum_{j=1}^{D} \left(w_j x_{ij} \right) \right)^2 + \lambda \sum_{j=1}^{D} w_j^2 \right) \tag{3.3}$$

Where λ is a user defined hyperparameter which must be greater than 0. A great λ will lead to small weight factors of the different x_{ij} values.

LASSO (Least Absolute Shrinkage and Selection Operator) regression is very similar to ridge regression, the main difference being that the absolute value of the parameters w_j is taken into account instead of the square. This results in many weights that will be close to zero since the sum of the absolute values is penalised. This is different than ridge regression where the sum of the squared weight is penalized which result in less extremes and more evenly distributed weights [37]. The equation of LASSO expression is shown below. [13]:

$$\hat{w}_{LASSO} = \arg\min_{w} \left(\sum_{i=1}^{N} \left(y_i - w_0 - \sum_{j=1}^{D} \left(w_j x_{ij} \right) \right)^2 + \lambda \sum_{j=1}^{D} |w_j| \right) \tag{3.4}$$

LASSO regression was extensively investigated in the paper of Wang et al. [34]. This method provided more accurate results than the at the that time commonly used methods. Improvement of the method was still possible.

3.1.3. Support Vector Regression

Support Vector Regression (SVR) is the regression alternative for the classification algorithm Support Vector Machines (SVMs). SVR is based on the SVM which works by deriving a hyperplane between two classes with the largest possible clearance between different classes. A hyperplane is a subspace that uses one dimension less than its ambient space. An example of SVM is visualized in figure 3.1.

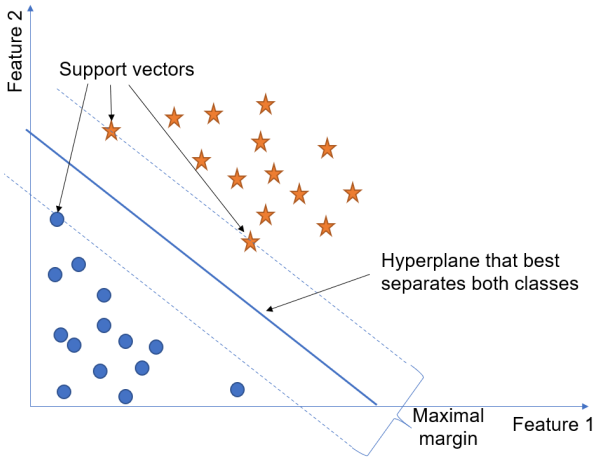

Figure 3.1: Example of SVM [28]

Supported Vector regression (SVR) by Smola and Shölkopf [30] works in a similar way as SVM but this time using a hyperplane that predicts the target values by using the training samples within a user defined margin, ε [11]. The purpose of adjusting this user-defined margin is to minimize the error and individualize the hyperplane that maximizes the margin [33].

The target values obtained by a non-linear SVR uses following expression [13]:

$$f(\boldsymbol{x}) = w_0 + \sum_{m=1}^{M} \boldsymbol{w}_m h_m(x) \tag{3.5}$$

The order of non-linearity is defined by the value m, $h_m(x)$ is a set of basis functions which can be linear or non-linear. The weight factors w are obtained by minimizing following function [13]:

$$H(\boldsymbol{w}, w_0) = \sum_{i=1}^{N} V(y_i - f(x_i)) + \frac{\lambda}{2}||w_m||^2 \tag{3.6}$$

Where N is the number of samples used for training , y_i the target value for each sample, λ a user defined hyperparamater such as in the LASSO method and the function V is defined as [13]:

$$V(r) = \begin{cases} 0 & \text{if } |r| < \varepsilon \\ |r| - \varepsilon & \text{otherwise} \end{cases} \tag{3.7}$$

Where r is the shortest distance between the hyperplane and the training points. From equations 3.6 and 3.7 it can be derived that when ε is chosen to be zero all the data points will contribute in the weighted sum which will be minimized.

In figure 3.2 a visualization of the SVR is shown. In this figure the hyperplane is represented by a line because only two dimensions are considered in the ambient space. This line symbolizes the actual used data set. The dashed lines represent the tolerance limits, the blue dots are data points inside the allowed region. Data points which fall outside the allowable region are orange.

3.1.4. Random Forest

Although different methods had promising results, other methods were still being investigated to efficiently predict fuel consumption or required engine power. For example the study of Yan et al. [38] investigated whether a Random Forest (RF) regression method could outperform other supervised machine learning algorithms. A random forest is an ensemble of decision tree regressors. This ensemble is in essence different decision trees which are trained separately. In a decision tree, different levels exist. At each of these levels a condition is stated which is created and learned by training the decision

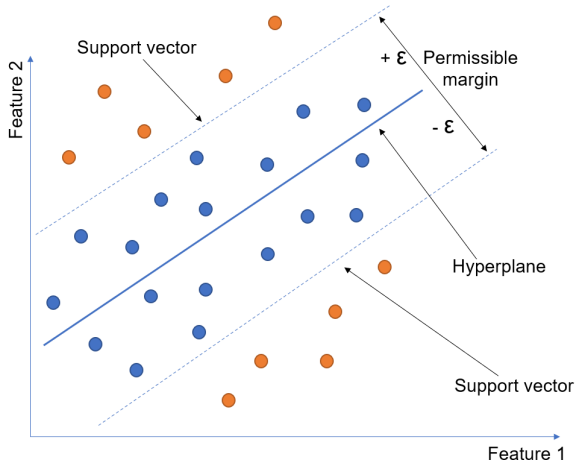

Figure 3.2: Support Vector Regression [30]

tree. Depending on whether the condition is true or false, a side of the decision tree is chosen. By going through several of these conditions a prediction of the target value can be made. A visualization of a random forest is shown in figure 3.3.

Not every decision tree will have the same conditions and thus accuracy. These depend strongly on the provided data during training which can be different for each decision tree. So when a new data point is fed to the different decision trees they predict different target values. The ensemble method then analyzes the prediction of all the the decision trees. In this way a more accurate and reliable target value is obtained [4].

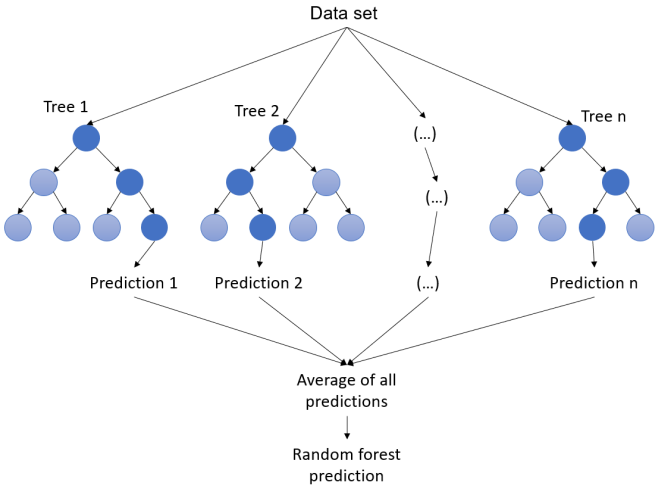

Figure 3.3: Random Forest example [33]

The research of Yan et al. [38] concluded that random forest outperforms SVR, LASSO and even artificial neural networks, discussed in next section. One big drawback of random forest is that the method cannot extrapolate. Meaning that only predictions can be made between the most extreme cases of the input features observed during training. An observation that is out of the interval of the extreme conditions of a certain feature used for training will lead to a large error.

3.1.5. Artificial Neural Networks

Artificial Neural Networks (ANN) are a major research subject for any source of data and have proven to be accurate. ANN are inspired by the human brain where in the human brain neurons receive a

stimulus and send out signals to many other neurons. These connections eventually result in an action or a thought. This also happens with artificial neural networks. A certain input is given to an input layer of neurons and a result finds its way out at the output layer of neurons. In between these two layers are several hidden layers of neurons. Each neuron in the next layer is connected with the neurons from the previous layer as can be seen in figure 3.4. Mathematically this can be represented by different matrices which contain the connections, weights, between the layers. An example of the weight matrix of the first hidden layers of the architecture presented in figure 3.4 is shown in next equation. Since the input layer has four input neurons (IN_j) and the hidden layer itself has 5 neurons (H_i) a 4x5 weight matrix is constructed.

$$\mathrm{w_{ij}} = \begin{array}{c} \\ IN_1 \\ IN_2 \\ IN_3 \\ IN_4 \end{array} \begin{array}{c} \begin{array}{ccccc} H_1 & H_2 & H_3 & H_4 & H_5 \end{array} \\ \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} & w_{15} \\ w_{21} & w_{22} & w_{23} & w_{24} & w_{25} \\ w_{31} & w_{32} & w_{33} & w_{34} & w_{35} \\ w_{41} & w_{42} & w_{43} & w_{44} & w_{45} \end{pmatrix} \end{array} \tag{3.8}$$

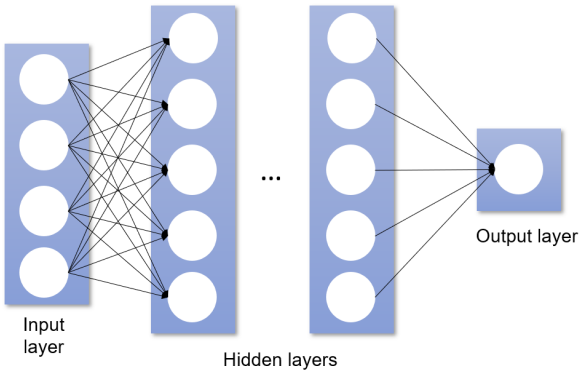

Figure 3.4: Concept of Neural Network [17]

The weight of the connections between the neurons is adapted while training the network, using back propagation, in order to get the correct results. In this way the network can learn itself in order to give accurate predictions. The general mathematical function for a neuron is presented in the formula below. In this equation the index n represent the layer, and i is a counter for the neuron in the specific layer, b represents the bias, w the weights, x the input to the neuron and j is a counter from 1 to J where J are the number of inputs from the incoming layer, n-1.

$$x_i^n = f\left(\sum_{j=1}^{J} x_j^{n-1} w_{ij}^n + b^n\right) \tag{3.9}$$

The weighted sum a neuron receives, serves as input variable x for the activation function in the neuron. This activation function is represented by the function in equation 3.9. All in between the parentheses represents the input variable for the input of the activation function. These activation functions make sure that non-linear data can be predicted. In figure 3.5 the mostly used activation functions are shown together with their mathematical expression. In recent research concerning neural networks the ReLu activation function is preferred [29, 8]. Mainly because the gradient of the ReLu function is easier to compute compared to the gradient of Tangent Sigmoid and Sigmoid activation functions. A ReLu function also easily leads to sparsity, when x < 0, which means that the output of the neuron is set to zero and thus the neuron gets deactivated. On top, a Relu activation function always leads to a strong gradient which is not the case for increasing absolute values of x in the Sigmoid and Tangent Sigmoid function. A small gradient can lead to vanishing gradients along the hidden layers of the neural network which causes a zero learning rate. Nevertheless, Tangent Sigmoid and Sigmoid activation function always produce outputs in the range of (-1,1) and (0,1) respectively. This is best in terms of optimization because these result in smaller differences in weight factors which lead to a faster convergence of an

accurate neural network. The output range of a ReLu function is (0,∞) which can lead to large jumps in weight factors and thus not beneficial for convergence. Therefore, in the research of Jeon et al. [17] it was found that a ReLu activation function needs a deeper (more hidden layers) and wider (more neurons in hidden layers) network in order to be as stable and accurate as the the Sigmoid or Tangent Sigmoid activation functions for predicting ship fuel consumption.

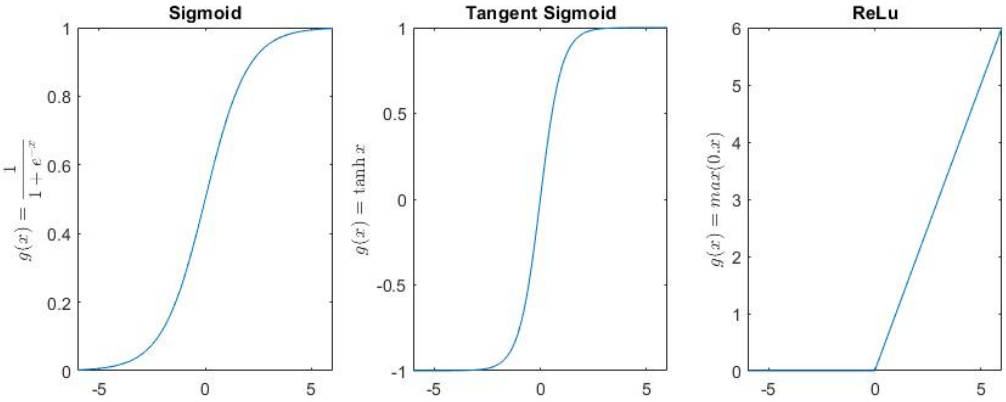

Figure 3.5: Activation functions

The research of Pedersen et al. [24] was one of the first to investigate whether ANN can be used to predict vessel's needed propulsion power. In this research few input parameters for the input layer of the network were considered and the network was only trained with 5 and 20 hidden layers. As a consequence, a relatively big error occurred but in this research it was concluded that ANN can be used to predict required power for a sea going vessel. This conclusion was picked up by Petersen [27] who performed a more thorough research in order to model a ship's fuel consumption by ANN. The results were very similar to the research of Pedersen et al. [24] but nevertheless a good comparison could not be achieved between the research of Pedersen [24] and Petersen [27] because different input data was used. After both researches had proven neural networks can be used in practice, further research was performed.

The applicability of ANN for predicting ship resistance was investigated by Couser et al. [6]. This research was to propose ANN as an alternative for the more common statistical methods. In their opinion the need for a deep network was not high. A network with a single hidden layer suffices. However, the researchers used data from a model that was used in a towing tank. The data was ideal and influence of external or meteorological factors was not considered.

Furthermore the size of the networks were investigated in researches of Ortigosa and Grabowska [23, 12]. These researches concluded that a network with 20 input features required 24 neurons in one hidden layer to obtain the most accurate results for resistance prediction. Although accurate results were obtained, these researches had no clear guidance for the general size of the network using ship data. It greatly depends on the number of input neurons. Nevertheless, it can be stated that no deep or wide networks are required in order to make predictions of ship resistance. It is possible to extent this conclusion for prediction ship fuel consumption because the input parameters are generally the same.

3.2. Comparison of Supervised Methods

Different studies have investigated the performance of ANN, MLR, RR, LASSO, SVR and RF separately. Other studies investigated the performance of these method to each other. This comparison was difficult to achieve from literature because every research uses a different data set. Data sets were often obtained from operational ships, but the same input features were not available for different ships throughout various researches. From the investigation performed by Gkerekos et al. [11], where different methods were trained with the same data set, it could be concluded that a method using a decision tree is very good. Although the accuracy's were very close to each other. Therefore the investigation

concluded that decision tree, SVR and ANN yielded the best results. But also Linear Regression performed reasonably well in the study. This is an opposition to what was stated in the research of Abebe et al. [2]. This indicates that when the hyperparameters are chosen correctly and model training is performed in a convenient way, any method which was explained in this chapter can predict accurate target values during testing of the learning method. This statement was also established in the research of Uyanik [33].

3.3. Performance Indicators

For the sake of objectively assessing the performance of the learning methods, various methods exist which are discussed in this section. In order to objectively assess the captured data set used for the learning algorithm is divided into three different subsets. The first subset is the training set which commonly is 80% of the total data set and is used to train the learning algorithm. The second subset, usually half of the remaining data, is the test set which is used to test the learning method on new instances that it has not encountered before. Based on these results hyperparameter tuning will be performed. The third set, the other half of the remaining data, is the validation set which is used to validate the results obtained from the test set after hyperparameter tuning. In order to quantify the performance of a machine learning method that predicted the test set and/or validation set, divers methods exist. In this section these methods are eliducated. Also a solution is provided for efficiently fine-tuning the learning algorithms in order to achieve more desirable performance.

3.3.1. Validation Of Methods

In most recent research like that of Uyanik and Gkerekos[33, 11] the validation of models is performed by applying the k-cross validation method. This is also the preferred method in literature and avoids overfitting [33, 27]. k-cross validation means that the data set is divided into a number of equal subsets. These subsets then receive a role for the first training [33]. The roles for the subsets are validation set, test set and training set. With these categorized subsets, the machine learning algorithm is trained, tested and validated for the first time.

For the second training, the roles of the subsets are shuffled and a different combination of subsets are used to train, test and validate the machine learning method for a seconds time [4] independent of the first training. The number of independent trainings is the number k. As such all the data can be used for validation, while the algorithm has never seen the validation set, in order to quantify the generalization of the learning algorithm. In the figure 3.6 a 4-cross validation method is shown with 8 different subsets. By applying this method, k validation values are obtained. These evaluation values are values that indicates the performance on predicting the validation set. A good general machine learning method should provide k values which are about the same such that the standard deviation is small.

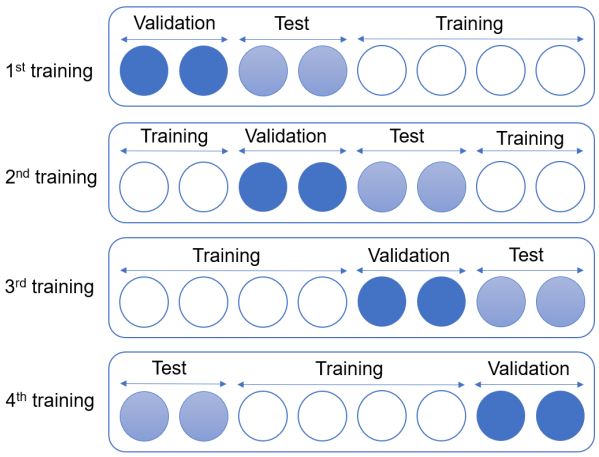

Figure 3.6: k-cross validation with 8 subsets and 4 iterations [33]

3.3.2. Error Metrics

In order to evaluate the predicted results by the machine learning model the error between the predicted value and the target value is evaluated. A quantitative comparison between both values can be performed by following formulas which are used in literature to evaluate machine learning models.

Root Mean Square Error (RMSE)

This error is the distance between the predicted value and the actual value on the regression curve. Taking the root of the sum of this value is the root mean square error. This translates into the following formula, which can have a value from 0 to ∞ [33]. This value gives an idea of how much error the algorithm typically makes in its predictions [4].

$$\text{RMSE}(y, \tilde{y}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2} \tag{3.10}$$

where y_i represent the actual data from the data set, $\tilde{y}_i$ represent the predicted value by the algorithm and n is the size of the used data set.

Mean Absolute Error (MAE)

In most cases the RMSE is the preferred evaluation criterion. Although, in certain researches where ouliers are present, the Mean Absolute Error is more useful [4]. With outliers the relative error will become small for the greater part of the data set which can give a false representation of the actual error. The equation for MAE is shown below which will produce a value between 0 and ∞:

$$\text{MAE}(y, \tilde{y}) = \frac{1}{n}\sum_{i=1}^{n}|y_i - \tilde{y}_i| \tag{3.11}$$

where y_i represent the actual data from the data set, $\tilde{y}_i$ represent the predicted value by the algorithm and n is the size of the used data set.

Coefficient Of Determination (R^2)

This error metric provides a number between 0 and 1. The performance and thus the quality of the method is at its best when R^2 is equal to 1. This indicates that the model predicts exactly the same value as the provided data. It can be stated that this error metric provides a number to determine how accurate the prediction is. Since the coefficient of determination is expressed in percentages it gives a better metric when small or large values are being predicted. It is also understood that the success of the model is higher when the coefficient of determination approaches 1 [36]. The formula for coefficient of determination is defined as:

$$R^2(y, \tilde{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3.12}$$

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{3.13}$$

Where y_i represent the actual data from the data set, $\tilde{y}_i$ represent the predicted value by the algorithm, $\bar{y}$ represents the mean values of y and n is the size of the used data set.

3.3.3. Tuning of Hyperparameters

After analysing the values obtained by using the methods obtained in sections 3.3.1 and 3.3.2, it is fundamental to tune the learning algorithms. This fine tuning is performed by changing the hyperparameters of the learning algorithm itself. These hyperparameters are pre-defined before the training of the learning algorithm starts and thus stays constant during training of the learning algorithm [4].

For most machine learning algorithms, different hyperparameters must be tuned, each having a different magnitude, in order to achieve an accurate and general learning algorithm. This leads to a large

number of combinations that are possible the user can define in the model in order to get the most accurate learning model. Therefore, the most common method found in literature to fine tune different machine learning algorithms is the Randomized Search [4]. This method randomly picks hyperparameters for the chosen method. By applying this random pick and training a large number of times, a large space of combinations of hyperparameters is explored. The number of chosen random picks is controlled by the user and therefore, a good control of computation time is obtained. Once the Random Search method is done and the user has an idea on the values of the hyperparameters, a more detailed search can be performed.

The proposed method to perform a detailed search of the hyperparameters is a Grid Search. With this method the user defines different numbers for the hyperparameters. By first applying a randomized search, the user is already able to narrow down these defined numbers for the different hyperparameters. In the Grid Search, all the different combinations of user defined values for the hyperparameters are tested. The hyperparameters are in this way determined accurately. Accurate hyperparameters will lead to an enhanced general machine learning algorithm. This Grid Search is usually also performed in combination with the k-cross validation method [4].

3.4. Feature Selection

In order to design an accurate machine learning model the input parameters must be chosen carefully. The study for the choice of input parameters is also called feature selection by machine learning experts. How the features are selected in literature and which are needed for the research presented in this thesis are discussed in this section.

Prior to the paper of Hu et al. [15] researchers focused only on ship features as input parameters. However the environment certainly also has its influence on ship performance. In the paper of Hu et al. [15] environmental input features for the neural network were also taken into consideration. The research of Hu concluded that while taking environmental input features into account, the prediction of ship fuel consumption becomes more accurate.

Most researches like Hu, Wang or Uyanik [15, 34, 33] take into account a large number of basic ship and environment features like speed, trim, draught, main engine power, wind speed, wind direction, loading configuration, etc. Because of the large number of possible input features researchers must be mindful to not include features which do not add any value. This to prevent the used data becomes cluttered and noisy. Using not unrequired features will also lead to an increase in computation time. In the research of Le [19], where a container ship was investigated, a big correlation was found between the cargo weight, number of containers and displacement. Since these 3 are strongly correlated, only one of these features is needed in the data set to determine an accurate target value. More than one of these features will lead to a greater training time of the machine learning method to eventually predict the same target value. It is concluded that in literature no clear guidance is described about which input features are the best. Every research performs a feature selection and extraction study of the available data for the research. Nevertheless there are resemblances between the features used in different researches with different data sets. These resemblances found in literature are:

- Course over ground
- Draft
- Wave height
- Wave direction
- Wind speed
- Wind direction
- Speed over ground
- Sea current
- ME fuel consumption
- Trim
- Cargo weight
- Displacement

3.5. Digital Twin

A digital twin will be used in a later stage of this research in order to assess to performance of a real vessel. It is because of this reason following explanation of a digital twin is provided in this section. In this thesis a digital twin will duplicate the behaviour of the real vessel and also serve as a base.

According to Gartner which is a leading organisation in research and giving advice to company leaders in, but not solely, information technology the definition of a digital twin is:

> A digital twin is a digital representation of a real-world entity or system. The implementation of a digital twin is an encapsulated software object or model that mirrors a unique physical object, process, organization, person or other abstraction. Data from multiple digital twins can be aggregated for a composite view across a number of real-world entities, such as a power plant or a city, and their related processes. ([10])

The bespoken software object or model that represents a real world entity, in this case a ship, often results in a white-box model (WBM), black-box model (BBM) or grey-box model (GBM).

A white-box model expresses the system based on mechanisms involving physical principles [21], empirical formulas are used. To determine ship performance while using a white-box model usually Holltrop and Mennen [14] is used, which is a method to determine still water resistance which can be modified to get a resistance and thus required power in a seaway. Since Holltrop and Mennen [14] used existing data from ships to find empirical formulas, the method can thus be described as a predecessor of machine learning. Since WBM are based on first principles and empirical data, the associated implicit uncertainty and assumptions will effect the applicability and the predictability of the models [20].

The opposite of a white-box model is a black-box model. A BBM describes input -output relations based on provided data [21]. This results in a machine learning model where the user does not need to understand and know the links or equations between different parameters in order to obtain a reliable value. Most researches that investigated the use and performance of machine learning techniques like Gkerekos, Pedereson, Peterson and Yan [11, 24, 27, 38] can all be classified as researches in a BBM. The results of a BBM are often better than a WBM. Although, the main drawback of a BBM is that the performance is highly dependent on the used data [21]. If the data inserted in the BBM is not enough or of high quality the accuracy of the BBM drastically decreases. A WBM will then perform better in such a case.

A grey-box model is a model which uses a combination of a WBM and BBM. It combines the physical principles with data driven modeling results [21]. Leifsson [20] created a GBM, considering the WBM and BBM in series or in parallel for estimating the fuel consumption of a container vessel. WBM and BBM in series means that one model, black or white, predicts or calculates a variable used by the successive box model, white or black. Both models in parallel means that one box model, black or white, determines a functional relationship between the input parameters and adds the values obtained by the other model, white or black, for obtaining an approximation of the desired value. The research of Leifsson [20] concluded that the GBM could significantly improve the performance of both WBM and BBM. The research of Liu [21] extended the analogy of Leiffson [20] and made a better GBM. Both researches were performed by using a limited amount of input features. Therefore, the GBM performed better than a BBM which requires more information but is usually more accurate with recent technology when more data is used than in the research of Leifsson [20]. GBM's are typically used less because a good understanding is required between different parameters of the data set which is often difficult due to the complexity and amount of data.

4

Research Goals

From the literature review performed in chapter 3, it is clear that machine learning is useful in the maritime world. Researches investigated mainly the applicability and accuracy of various machine learning methods. Comparisons between the different methods was also extensively executed. In this chapter the knowledge gap in literature is identified that will be filled in by this research. Also the goal of the research is discussed.

The published literature which investigated the performance of different supervised learning methods took into account a combination of ship parameters and environment parameters. Under environment parameters the wind and sea conditions are classified. Under ship parameters, the characteristics of the vessel are classified but also the operational data. These can be the speed, course, center of gravity, draft, trim etc. None of the researches took into account the vessel motions, although as shown in chapter 2 the vessel motion has influence on the required energy to propel the ship. A knowledge gap is identified in literature. No research has been performed that has taken vessel motions into account for determining vessel fuel consumption using machine learning algorithms. Although, when the vessel motions are taken into account in the machine learning methods a deeper understanding of the vessel fuel consumption due to vessel motions might be created.

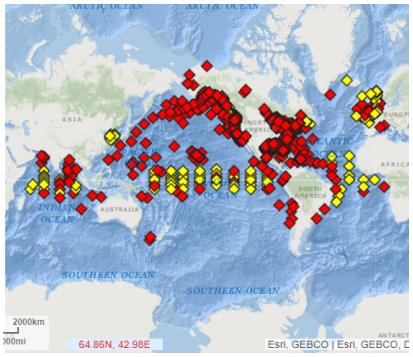

Figure 4.1: NOAA Maritime stations [22], yellow squares are stations with recent data, red squares are stations which provided no data in last 8 hours at a arbitrary point in time

The identified gap is filled in this research by developing machine learning methods using motion data instead of third party weather data and operational features as data source. Since the observed motions of the vessel are a result of the environmental conditions, such as waves, wind and swell at that specific position it lends itself to replace the environmental features. The used environmental data in literature is often coming from the National Oceanic and Atmospheric Administration (NOAA) and from noon reports. These noon reports are filled in by the crew on board once a day and contains human error. NOAA only uses a small number of measurement stations at few locations in the ocean thus the observations in open ocean are often inaccurate. The measurement stations can be observed in figure

4.1. Therefore, in this research no environmental input features from third parties are used. Accurate motion data replaces the inaccurate environmental features in a machine learning black-box model to determine vessel fuel consumption.

To develop the machine learning algorithms motion data, vessel features and operational features will be used from a clean ship that went to dry-dock recently. Which means that no hull fouling or appendage fouling is present. The cleanliness of the ship is also constantly being monitored with an old, less responsive and less accurate model which gives an indication if the new observed data is from a clean ship and thus can be used for the machine learning method or not. Using clean ship data to train the learning algorithm leads to estimation of fuel consumption for a clean ship. In fact a basic digital twin is made of the clean ship to determine vessel fuel consumption. In this way the fuel consumption of the real vessel can be compared to the consumption of the digital twin to determine the over-consumption which is a measure of ship performance. The over-consumption is thus defined as difference in fuel consumption between a fouled and a clean ship. A possible bad performance of the vessel can be noticed by applying this method and thus a better performance indication is obtained.

The above paragraphs can be summarized in following statement which captures the main goal of the research: **in order to monitor the performance of an operational ship in off-design conditions, a machine learning black-box digital twin is made which has as input vessel motion data and operational features to model the vessel fuel consumption which can be compared to real vessel fuel consumption for obtaining the fuel over-consumption during a sea-voyage.**

In order to achieve the model to monitor the performance of the vessel and thus make a digital twin, the architecture in figure 4.2 will be made. In the architecture two blocks can be divided to obtain fuel consumption. One block is the real vessel, which will have a real vessel consumption due to operational features, vessel features and motion data. The motion data serves as improved replacement for environmental data. For the digital twin these input features are the same but in order to obtain a fuel consumption, the measured features must first be processed through a machine learning algorithm for obtaining the fuel consumption of a clean ship. Afterwards, both obtained fuel consumption's can be compared to determine the vessel's over-consumption during the covered distance of the measurement time span.

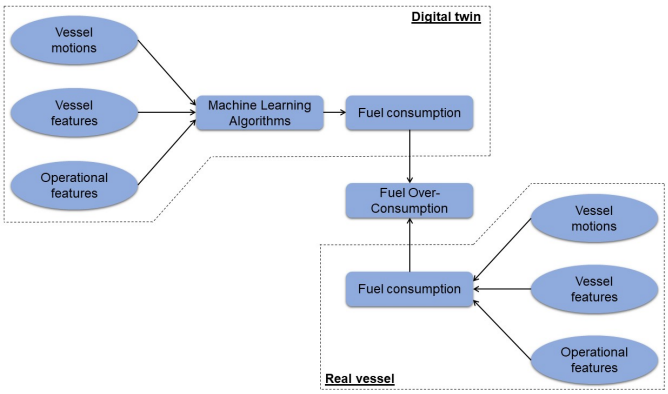

Figure 4.2: Architecture of Research

The main goal of this research is to develop the digital twin using machine learning techniques without the use of environmental data for determining fuel consumption. In this way CMB can be independent of third party weather organisations to determine vessel fuel over-consumption which is the difference in fuel consumption between the clean digital twin and the real vessel. Fuel over-consumption is important information for CMB in order to monitor the energy consumed by their vessel. This information might lead to actions which reduce energy/fuel consumption when the consumption increases by 35%. A small increase in fuel consumption of 1-3% will be declared as inaccuracies in measuring devices. By applying this method, there is no need anymore to analyze the vessel in CFD for numerous environmental conditions. On top of this an insight gets created under which conditions ship performance is not optimal due to vessel motions.

5 Research Approach

In this chapter an overview is given about which data sources will be used and how the sources will be pre-processed for the machine learning algorithm. Next, an explanation is provided for how the machine learning will be developed and evaluated using the data from the different sources. Also a research plan and timeline is provided in order to get an overview of the project.

5.1. Data Gathering

In this research different data sources are used and explained in this section. The equipment used for collecting data is installed on board of a container ship which regularly crosses the Atlantic ocean. It is believed that this regularly crossing is beneficial for the research because of the long sea-voyage under different weather and operational parameters. The main particulars of the considered vessel are given in table 5.1. Of the different systems installed on board the vessel, following will be used to create a data sets for the machine learning application:

Table 5.1: Vessel characteristics

Parameter	Value
Length over all	210 m
Breadth	30 m
Max draft	11.5 m
Deadweight	33 434 t
Installed power	21.56 MW

1. Veinland data: Every 15 seconds this system saves data coming from the navigational bridge. This data contains speed, position, course over ground, heading, wind speed, wind direction and much more. The Veinland system also receives data coming from the engine room. This data is the output of flow meters from the fuel lines. These are flows for the main engine fuel oil but also for the auxiliary engines and boilers.
2. Motion sensor data: The motion sensor MTi-670 from Xsens is used to measure the vessel motions. These are measured twice per second. From this sensor Heel, Pitch, Yaw, Angle velocities and (free) accelerations in 3 degrees of freedom are available.
3. Noon reports: Every noon on board of the vessel a form is filled in by the crew. This form contains information about the draft, covered distance, etc. This form contains data that is stable and does not fluctuate excessively.

In order to make a useful data set for the machine learning algorithm the three data capturing systems must be combined together in one data set. Every data source has different intervals of data capturing, these time intervals will be aligned to create one data file that contains all the data. Also the outliers which are likely to be a measuring error will be filtered out of the data set. With the resulting data file, feature analysis and engineering will be performed.

5.2. Model development

For this research machine learning black-box models using MLR, Lasso, Ridge, SVR, Random Forest and ANN algorithms will be developed in order to predict vessel fuel consumption using vessel motion data, vessel features and operational features. No data coming from an external weather organisation will be used. Also the learning algorithms will be tuned by performing a research on the optimal hyperparameters. These are optimal when the most accurate results from the learning algorithms are obtained after validation. This validation will be performed by using the k-fold cross validation method. The hyperparameters are determined by performing a Randomized Search and the performance indicators that will define the accuracy of the method are the Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and the coefficient of evaluation (R^2).

For the implementation of the machine learning algorithms a Python environment will be used. Therefore the model will depend on the Python platform and libraries published for Python. The used Python libraries are Sci-kit learn [25] for supervised machine learning methods such as MLR, RR, LASSO, SVR and RF. The Keras [5] library which is part of Tensorflow [1] is used for artificial neural networks.

Consecutive to the development of machine learning algorithms a study will be performed to find out which models are the most accurate and why. The most accurate model will then be used as black-box for the digital twin. The fuel consumption of this digital twin will be compared to the actual fuel consumption of the considered ship for a specific voyage. The difference in fuel consumption leads to the over-consumption which is a quantification of the vessel performance. Based on these results, the company can take actions in order to increase the performance of the vessel. This can be a hull cleaning or a maintenance for example.

5.3. Research Plan

In this section the different processes of the research are discussed and placed in a flow chart. The first process was to perform a thorough literature review. In this literature review two main aspects were investigated: the relation between vessel motions and the machine learning algorithms that predict ship performance parameters. Next to the literature review the already captured data needs to be pre-processed. The data from the three earlier described systems are coupled together in order to have one big data set with all the parameters. In the data pre-processing also a solution will be found in order to deal with time dependent data like motions of the vessel. The complete data set will be updated regularly because data gathering continues during the development of the machine learning methods.

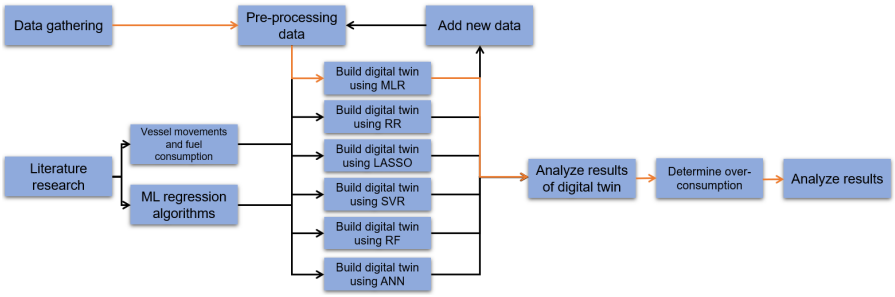

Figure 5.1: Flow chart of tasks

After pre-processing of the data the actual design of a solution starts by building methods for predicting the over-consumption of the vessel. First the black boxes that make use of the MLR, RR, LASSO, SVR and RF for building a digital twin will be developed. Next, an ANN network will be built for determining the fuel consumption of the digital twin. The results of all the methods will then be analyzed, trained with updated data and conclusion will be drawn on which algorithm is the best and why. Next, the best algorithm will be used to make a model that determines the over-consumption using data from the real vessel. These results will be analyzed and conclusion will be drawn why the ship performance might not be optimal. Also an analysis will be performed on the relation between vessel motions and the fuel consumption of the vessel under certain environmental conditions.

The processes described in this subsection are presented in a flow chart in figure 5.1 in order to shown the inter dependencies between each task. The orange path shown on the figure represents the critical path. For the critical path the learning method MLR is chosen as this is the easiest to implement. After this method is implemented, more advanced learning algorithms can be developed.

5.4. Research Planning

The flowchart presented in previous section is translated in the planning shown in table 5.2. All tasks are performed in a consecutive order. Currently, the literature review has already been performed. Next, a considerable amount of time is planned in order to prepare the data in the solution design (first five tasks). The building and analyzing of the different machine learning algorithms are then performed. Consequently, the best model will be used to determine the over-consumption of the real vessel. After this, the model results will be analyzed and updated with new data. Lastly, the results of the analysis will be written down in the writing phase. The planning presented in table 5.2 results into milestones where certain parts of the research must be finished. The milestones and corresponding dates are presented in table 5.3.

Table 5.2: Planning of tasks

Phase	Task	Duration in weeks
Data Gathering	Capturing data of Maersk Nimes	34.0
Literature study	Research papers about motions and consumption	2.0
	Research ML regression algorithms	2.0
	Finishing literature reviw report	2.0
Solution Design	Deal with time dependent data	1.0
	Get every source in one data set	5.0
	Make setup for filtering outliers	1.0
	Correlation Study	1.0
	Prepare data set to feed to ML	1.0
	Building MLR for digital twin	1.0
	Building LASSO for digital twin	1.0
	Building Ridge for digital twin	1.0
	Building SVR for digital twin	1.0
	Building Random forest for digital twin	1.0
	Building ANN for digital twin	3.0
	Connect twin and real vessel consumption	2.0
	Update and analyze models and results	3.0
Writing results	Finishing report	6.0

Table 5.3: Milestone planning

Milestone	Date
Finish literature review	10-05-2021
Data pre-processing ready	21-05-2021
Finish all ML algorithms	23-07-2021
Finish result analysis	27-08-2021
Green light meeting	11-10-2021
Graduation presentation	29-10-2021

5.5. Hypothesis

As from the literature review it can be concluded that machine learning algorithms can predict fuel consumption of a vessel accurately while making use of weather data. Therefore it is expected that the accuracy of the estimation using motion data, which is a direct results of meteorological conditions, will be close to the accuracy of the earlier performed researches.

The estimations of the developed algorithms will be useful in case the error made is in the level of accuracy of the measuring devices. This level of accuracy is set by CMB on 1-3% of the fuel consumption. In this range of inaccuracy, the company will not perform any maintenance on the vessel in order to reduce the fuel consumption.

II

Black-box Development

6 Data Pre-processing

In this chapter the captured data from the three different data sources are discussed and processed. The measuring frequency of the three different data capturing systems are equalized in order to feed the machine learning algorithm. The parameters measured by the different sensors are also adapted such that they represent the situation of the specific time window. Also extra parameters are created using a combination of multiple measuring units. The procedure is described in this chapter together with filtering and scaling of the captured data set in order to prepare the data set for machine learning purposes.

6.1. Re-Sampling

The measuring frequency of the multiple data sources is dissimilar, as can be seen in table 6.1. Therefore the data of the different sources are equalized to a common frequency such that it can be used for machine learning algorithms. The frequency chosen is 20/min because it gives a good trade off between robust estimation of fuel consumption and time-scale of change in the variables [27]. This frequency corresponds to a time window of 3 minutes. In order to re-sample the data to such a time window, different operations are performed on the measured data such that little information is lost on the time series. The required operations on the distribution obtained from the 3 min measuring interval are determined in next section, partly by investigating a sine wave. Next the determined operations are performed on the measured data of the different systems and feature engineering is performed.

Table 6.1: Measuring frequency

	Initial measuring frequency	New frequency
Veinland data	4/min	20/hour
Motion data	2/s	20/hour
Noon reports	1/day + End of sea passage	20/hour

6.1.1. Describing Time Series

The time series will be described by characteristic describing the value distribution in the 3 min time interval. In this section it is discussed which operations are performed such that the distribution is described without losing essential information of the time series. A time series and in this case a motion can be described by the amplitude and the frequency.

For describing the amplitude of the motion, the mean and standard deviation will be used. The mean gives information around which point the time dependent data fluctuates. By making use of the standard deviation an estimation can be made on the strength of fluctuation. There is a probability of 99.7% the time series is in between the interval of $[\mu - 3\sigma, \mu + 3\sigma]$. Also the kurtosis of the distribution is determined. This value gives an indication on how much the extreme values of the time series are achieved. Since it is not guaranteed that every 3 min distribution is a normal distribution also the skewness of the

distribution is determined.

To describe the period and thus frequency of the motion, a monochromatic wave, sinus wave, is investigated. This curve represents a simplified motion of the vessel with frequencies ω and amplitude A. The symbol "t" represent the time series and ϕ the phase angle. From the sinusoidal motion also the derivative (y') is determined.

$$y = A\ \sin(\omega t + \phi) \tag{6.1}$$
$$y' = \omega A \cos(\omega t + \phi) \tag{6.2}$$

With the known relationship between period (T) and frequency, $\omega = \frac{2\pi}{T}$, the expressions for the monochromatic wave becomes:

$$y = A\ \sin\left(\frac{2\,\pi\,t}{T} + \phi\right) \tag{6.3}$$
$$y' = \frac{2\,A\,\pi}{T} \cos\left(\frac{2\,\pi\,t}{T} + \phi\right) \tag{6.4}$$

Equations 6.3 and 6.4 result in two distributions. In order to evaluate the dependency of the frequency to the 4 moments of both the distributions, the mean, standard deviation, kurtosis and skewnness are analytically calculated. The prime (') represents the derivative with respect to time.

- The first moment, the mean is:

$$\mu = \frac{1}{T}\int_0^T y\ dt = 0 \tag{6.5}$$
$$\mu' = \frac{1}{T}\int_0^T y'\ dt = 0 \tag{6.6}$$

- The second moment, standard deviation is:

$$\sigma^2 = \frac{1}{T}\int_0^T (y-\mu)^2\ dt = \frac{A^2}{2} \tag{6.7}$$
$$\sigma'^2 = \frac{1}{T}\int_0^T (y'-\mu')^2\ dt = \frac{2\pi^2 A^2}{T^2} \tag{6.8}$$

- The third moment, skewness is:

$$skew^3 = \frac{1}{T}\int_0^T (y-\mu)^3\ dt = 0 \tag{6.9}$$
$$skew'^3 = \frac{1}{T}\int_0^T (y'-\mu')^3\ dt = 0 \tag{6.10}$$

- The last moment, kurtosis is:

$$kurt^4 = \frac{1}{T}\int_0^T (y-\mu)^4\ dt = \frac{3\,A^4}{8} \tag{6.11}$$
$$kurt'^4 = \frac{1}{T}\int_0^T (y'-\mu')^3\ dt = \frac{6\,\pi^4 A^4}{T^4} \tag{6.12}$$

As can be seen from equations 6.5, 6.6, 6.9 and 6.10 there is no relation between the frequency and the first and third moment. In fact, in case the measuring interval ends with a full period, the first and third moments remain zero for both the non-derivative and first derivative distributions. This leads to the conclusion that mean and skew will not have any influence on the describing the frequency of the motion. From the second and fourth moment, only the moments describing the first derivative solution

are dependent on the period and thus the frequency of the motion, shown in equations 6.8 and 6.12. The second and fourth moment describing the non-derivative motion are only dependent on the amplitude of the the monochromatic wave. Nevertheless, the four moments of both the non-derivative and first derivative distributions will be determined as it might have underlying information that the machine learning algorithms might find useful in case the time window does not end exactly on the end of a full period. While training the machine learning algorithms an automatic feature selection process is carried out in order to filter the possible unnecessary parameters.

6.1.2. Motion Sensor Data

The following parameters are measured by the motion sensor and are used for creating a data set. The motion sensor measures the motions of the ship in a frequency of twice per second.

- Roll
- Pitch
- angular velocity around longitudinal ship axis
- angular velocity around transverse ship axis
- angular velocity around vertical ship axis
- Acceleration around longitudinal ship axis
- Acceleration around transverse ship axis
- Acceleration around vertical ship axis
- Velocity in North, West, Up-axis system

Since the motion data is received in a high frequency, this data must be down-sampled. It is chosen to down-sample to a frequency of 20/hour which corresponds with a time window of 3 minutes. For each sampling frame of 3 minutes the four moments of each motion distribution are determined. Also for the distribution of the derivative of each motion, the four moments are determined. It is believed that these eight parameters gives a good representation of the ship motions in the chosen time frame based on the analysis performed in section 6.1.1.

Additionally a Fourier transformation of the motion data is also performed. Due to this transformation a spectrum can be made of the motion signal using the amplitudes and frequency of the Fourier terms. In this research only the spectrum until a frequency of 0.4 Hz is considered. Based on figure 6.1, where random spectra are shown, there are no significant amplitudes above 0.4 Hz. The spectrum terms above this limit are classified as superfluous. The continues spectrum until 0.4 Hz is distributed in 20 different frequency bins. The value of each bin is the sum of the amplitudes where from the frequency falls inside the limits of the specific bin. An example of a binned continues spectrum is shown in figure 6.2. The values of the 20 bins for every 3 min time window is also passed to the machine learning algorithms.

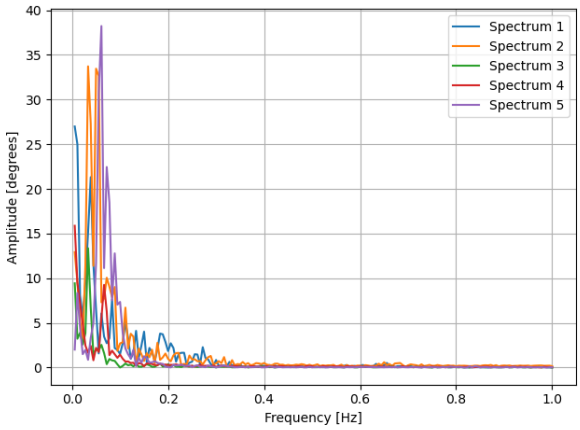

Figure 6.1: Spectra of random time windows

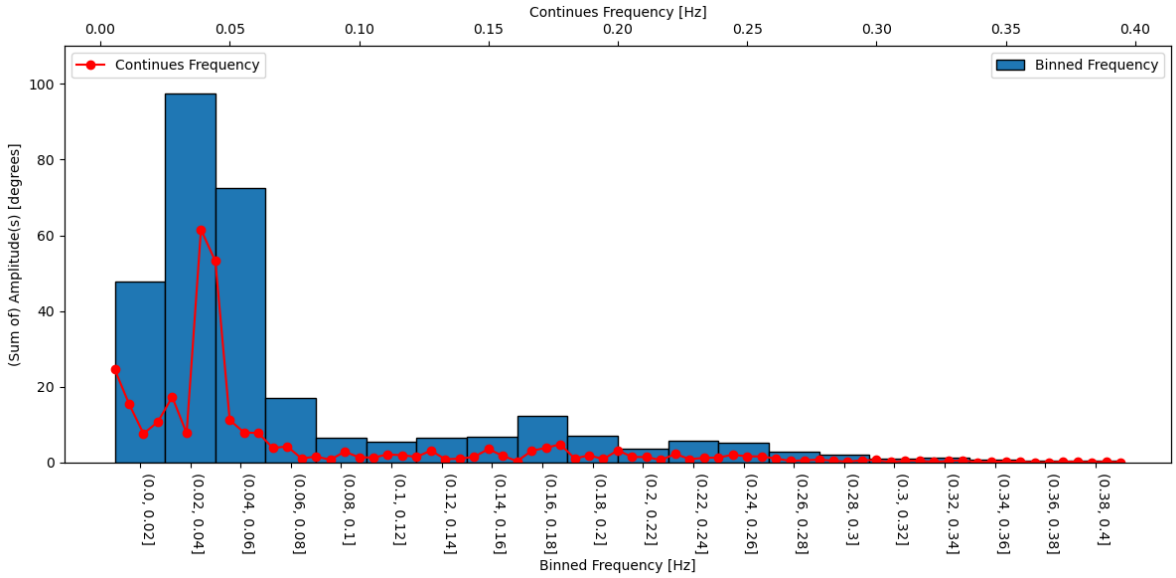

Figure 6.2: Binning of random continues spectra

Eventually the use of the moments and Fourier terms leads to 28 features that describe one motion. Each motion has:

- 4 moments describing the non-derivative motion distribution.
- 4 moments describing the first derivative motion distribution.
- 20 bins describing the sum of amplitudes in a frequency interval.

The motion sensor also had the option to measure the ship heading at a higher frequency than the measurement of the heading of the Veinland system. Unfortunately the heading measurement of the motion sensor is very unstable because the motion sensor uses earth magnetic field to the heading. Since the motion sensor is installed on board a metal ship, earths magnetic field is distorted drastically. This leads to an unstable heading measurement even with software that constantly calibrates for the magnetic distortions. A comparison of the high frequency heading measurement of the motion sensor and the lower frequency heading measurement by the Veinland system is shown in figure 6.3. The high frequency sensor data is very unstable and it also physically not possible to get such a large heading differences in such a short time span.

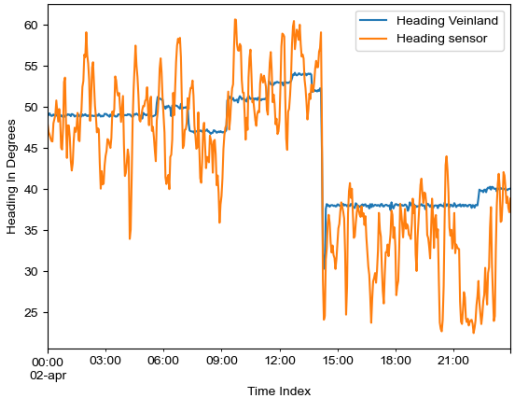

Figure 6.3: heading from ship and motion sensor

6.1.3. Veinland Data

Veinland data is a summary of data from numerous ship sensors which can be divided in two big parts, namely bridge data and Engine room data. The systems receives bridge data from the on board installed Electronic Chart Display Information Unit (ECDIS). The ECDIS itself receives information from the GPS-receiver, anemometer, gyro-compass, depth meter, speed meter and other numerous type of sensors used for navigation. The data coming from the engine room contains flow data of fuel lines to the installed engines (auxiliary and main), torque meters and revolution meters. All the data received from the various measuring devices are saved every 15 seconds in the Veinland system. The used data from the system is listed below.

- Speed through water
- Relative wind speed
- True wind speed
- Relative wind direction
- True wind direction
- Ship's heading
- Main engine actual fuel consumption
- Rudder angle indication

For all the Veinland data except the main engine actual fuel consumption, which is the target value, again the four moments of a 3 minutes time window distribution is calculated. This is done for the derivative and non-derivative distributions. As such, information about the less fluctuating time series is also obtained and can be used for creating a data set.

Since machine learning algorithms cannot handle circular values, like the wind direction or heading, these are converted in complex numbers. Complex numbers are chosen because these are not circular but still gives a precise angle representation. The imaginary and real part of the complex number are divided in the data set and are labeled differently. In this way only real numbers in the data set are present.

6.1.4. Noon Reports

A Noon Report is a data source that is sent every noon on board the ship and at every end of sea passage to the company. All the reports contains static features. Because of the low measuring frequency of the features, only the required data which was not available from other data sources with higher measuring frequency was used. These parameters are listed below:

- Ship's trim
- Ship displacement
- Mean draft
- Fuel Calorific Value
- Distance between meta-center and center of gravity

In order to match the sample frequency of 20/hour the samples of the Noon Reports are up-sampled with a linear interpolation in between real measured sample points. It is chosen to interpolate linearly in between the data points because it is assumed there is a linear progress in between the data points. The up-sampling using linear interpolation provides a value at the required time on the linear curve between the real measured data points. The fault made by up-sampling is small since the features covered by the noon report does not alter heavily in between the reports. Care must be taken during start of a sea passage because in port is the only time these values change heavily. Because the update of the noon report is hours after the start of the sea passage an error is made. The error made is diminished by the fact that speeds lower than 6 knots (corresponds to dead slow speed) will be left out of the data set. These are speeds that are often reached during manoeuvring in port. Once these data points are already left out, only a smaller number of data points with less accurate noon report data remain in the data set which will not lead to large consequences whilst training the algorithm. It could be stated that it might be better to not use the data between departing at port and a Noon Report update. Nevertheless, since it is also the goal to estimate the fuel consumption at lower speed it is important to not remove a vast majority of the data points in the training set at low speed. In the interval of departing at the port and the update of the Noon Report the ship sails at lower speeds (6 to 10 kts) in port and at the start of the sea passage. Thus data points at speed lower than 6 knots will be removed.

The data points with a speed through water higher than 6 knots will not be removed such that also fuel consumption at lower speeds is included in the data set.

Another option was to perform a constant interpolation in between the real measured data points. This option was not chosen since a large error is made at the start of the sea passage. During port stay the static features change dramatically and are only updated at the noon after start of sea passage. When a constant interpolation is performed, the start of the sea passage would use the parameters of the end of sea passage report before port stay. Most probably the static features will be very different. This would result in data points in the data set that are incorrect. While using linear interpolation between the end of sea passage report and the noon report after the start of sea passage, the error made will be diminished. This results in more accurate and more reliable data points in the training data set.

6.2. Feature Engineering

From the data obtained by the three different data sources extra features are calculated in order to create more useful dimensions for the machine learning algorithm. These extra created features are explained in the following sections.

6.2.1. Velocity Transformation

The velocity obtained by the motion sensor is in an earth fixed axis system. This is transformed to a motion sensor fixed axis system where the x-axis corresponds with the longitudinal ship axis, the positive y-axis with the port side transverse axis of the ship. The vertical axis of the vessel remains the same in both axis system, namely upwards. By applying the transformation below with the heading obtained from the Veinland data the vessel speed overground longitudinal and sideways was obtained. Sideways speed is also called sway. To describe this time dependent data again the four motions of the non-derivative and derivative distributions are determined for these parameters.

$$\begin{bmatrix} V_x \\ V_y \\ V_z \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_N \\ V_W \\ V_U \end{bmatrix} \tag{6.13}$$

Where ϕ is the rotation around the z axis of the earth fixed system, the heading in this case. In this formula the heading in degrees is still used but is later converted to a complex value. V_x, V_y and V_z are the speeds of the vessel in motion sensor fixed axis system and V_N, V_W and V_U are the speeds of the vessel in earth fixed axis system. Where N, W and U stands for North, West and Up, respectively.

6.2.2. Velocity due to Yaw and Pitching

Since the motion sensor could not have been installed close to the center of gravity, which is also the point of rotation for a ship, due to practical considerations the pitching and yawing of a vessel will induce a velocity upward and sideways respectively. These velocities must be captured in the data set. Every start of sea passage the position of the ship's center of gravity is determined by the loading computer. The information about the center of gravity is added to the data set of the motion sensor in order to calculate the radius between the center of gravity and motion sensor position according to below formulas and figure 6.4. The position of the center of gravity is updated only once per voyage since it is believed that small changes in the position of center gravity will not influence the velocity due to yaw and pitch heavily.

$$R_x = X_z - KG \tag{6.14}$$

$$R_z = LCG - X_x \tag{6.15}$$

$$R_y = \sqrt{R_x^2 + R_z^2} \tag{6.16}$$

Where X_z and X_x are the height above the keel and position in front of aft perpendicular of the motion sensor, respectively. KG and LCG are the height and position in front of aft perpendicular of the center of gravity, respectively and R_x, R_z and R_y are the radii of circles on which the angular velocities obtained for each axis are measured. These radii are then multiplied with the angular velocity around the y- and

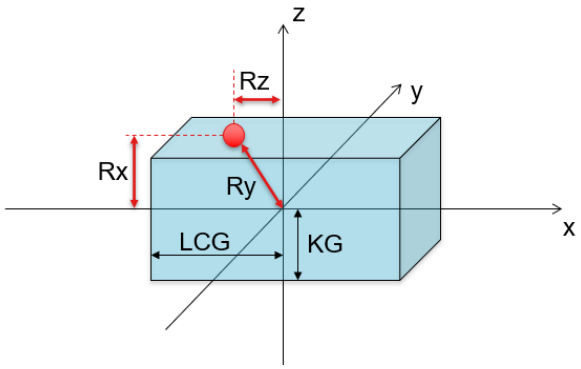

Figure 6.4: Schematic drawing of position of the sensor, red dot is the sensor

z-axis (ω_z and ω_y) for obtaining the velocity due to yaw and pitch (V_{yaw} and V_{pitch}), respectively, at the position of the motion sensor. The equation is depicted below:

$$V_{yaw} = R_z \omega_z \tag{6.17}$$
$$V_{pitch} = R_y \omega_y \tag{6.18}$$
$$\tag{6.19}$$

The velocity due to yaw and pitch have the greatest influence on the sideways and upward speed respectively. Other influences on the vessel motion measurements due to the placement of the sensor out of center of gravity are negligent and are therefore not determined.

6.2.3. Energy Consumption

The ship switches to different fuels with different density. Thus, the fuel consumption in ton/day gives a wrong image of the fuel consumption. Switching from fuel consumption to energy consumption tackles this inconvenience. By multiplying the main engine actual fuel consumption obtained from the Veinland data with the calorific value obtained from the Noon Reports, the energy consumption per day is determined. This energy consumption is independent of which type of fuel is used. The energy consumption per day is also the target value the machine learning algorithms must determine.

6.3. Filtering Data

Filtering of the obtained data is required in order to remove wrong measured values and remove outliers. First of all, only the data is selected that has a speed through water above 6 knots and which has an energy consumption of greater than 0. By performing this first selection process, the large amount of points measured during port stay and manoeuvring is already filtered out. A speed through water is taken of 6 kts since the dead slow speed of the vessel is at this level. A dead slow speed is only used while manoeuvring and not during sea passage. In addition there must be an energy consumption such that the points of time where the ship has still a forward speed without propulsion are also removed. Otherwise, these would lead to wrong information for the machine learning algorithm which results in wrong estimation of energy consumption. In addition, filtration on different features is also performed such that outliers and unrealistic data points are removed. In appendix A.1, a part of the data that is left after the first filtration, is shown to illustrate the filtration process. The multiple features where a filtration is performed on are discussed below.

6.3.1. Transverse Velocity

As can be seen from the figure 6.5 a clear outlier is observed in Vy velocity (transverse velocity), and Vx velocity (longitudinal velocity). This is probably a point observed while manoeuvring and was not yet filtered out. In this research it is chosen to filter the transverse velocity instead of the longitudinal velocity. This could also be switch. On which feature this specific filtration is performed is not important as long as the outlier is removed.

Since the transverse velocity is not normally distributed, thus not symmetrical, and extreme values

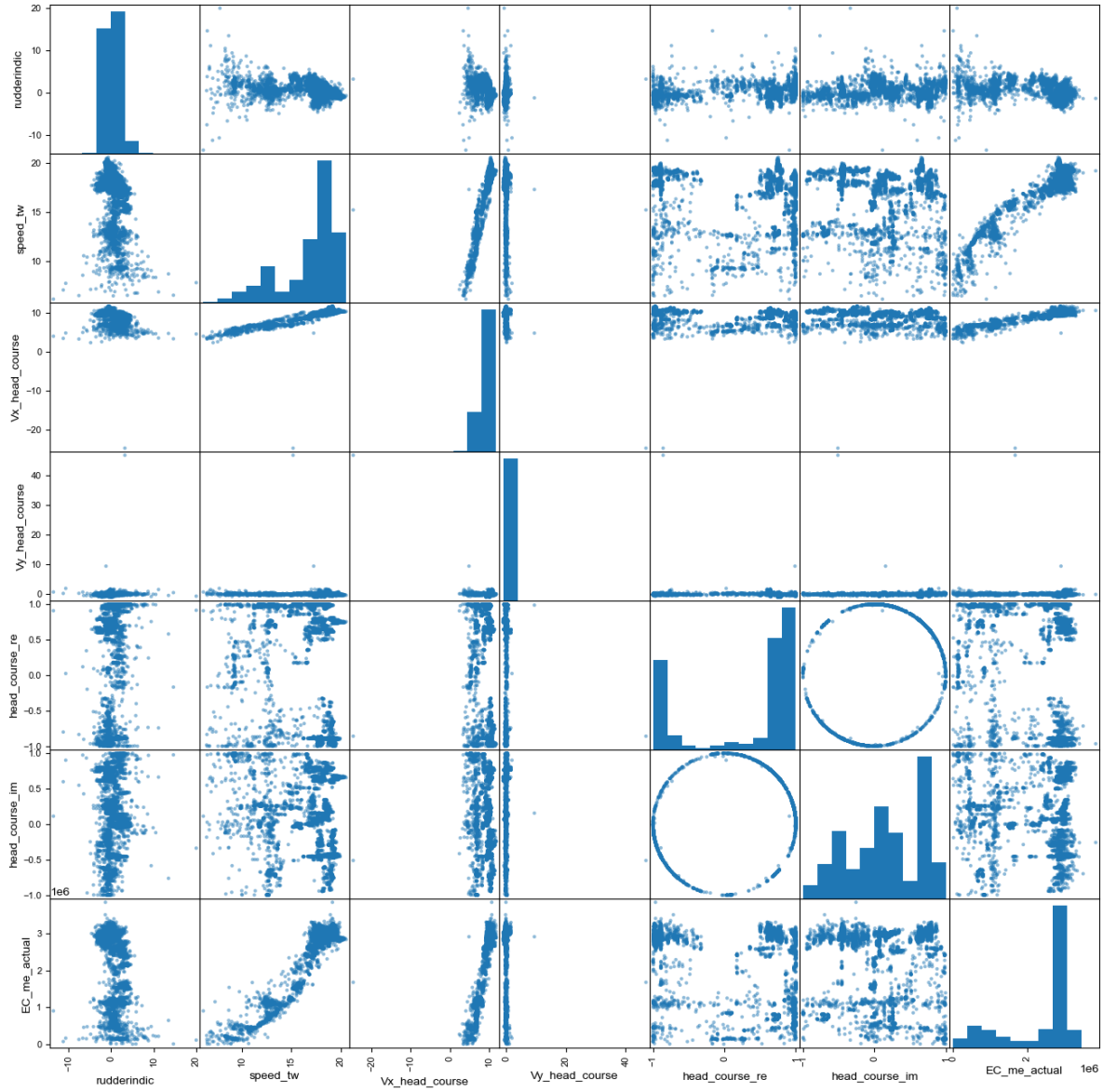

Figure 6.5: Scatter matrix of selected features, label explanations are provided in appendix C

might be correct observations it is chosen to filter the feature using the Inter Quantile Range (IQR). This filtration method determines the range between the first and third quantile and then used as a value for obtaining the upper and lower boundaries.

$$IQR = Q1 - Q3 \tag{6.20}$$

$$\text{Upper Limit} = Q3 + 1.5IQR \tag{6.21}$$

$$\text{Lower Limit} = Q1 - 1.5IQR \tag{6.22}$$

Where Q1 and Q3 are the values that corresponds with the first and third quantile respectively. Any data point that situates itself outside the upper and lower limit is classified as outlier and is removed from the data set. The remaining data points are shown in appendix A.2

6.3.2. Mean Derivative Longitudinal Velocity

Based on the results after filtration of velocity in transverse direction it is chosen to also filter the mean derivative of the longitudinal velocity. In such a way the points in time where the ship is accelerating or decelerating are removed from the data set. This would otherwise lead to relatively high energy

consumption data points at slow speeds due to the inertia of the vessel and slip of the propeller. Since the mean derivative of the longitudinal indicates this phenomena a filtration is performed on mean derivative of the longitudinal velocity using IQR range as shown in equations 6.20 to 6.22. In appendix A.3 the result after this filtration is illustrated.

6.3.3. Rudder Angle

The data set is also filtered on rudder angle such that large rudder angles (uncommon in sea passage) are filtered out and the remaining points during manoeuvring are removed. Since the rudder angles are normally distributed it is chosen to set an upper and lower limit using the standard deviation and mean of the set.

$$\text{Upper Limit} = \mu + 3\sigma \tag{6.23}$$

$$\text{Lower Limit} = \mu - 3\sigma \tag{6.24}$$

Where μ is the mean of all rudder angles and σ is the standard deviation of all rudder angles. It is chosen to take 3 times the standard deviation because this corresponds with 99.7% of rudder angles that are within the interval of the upper and lower limit. This is found to be satisfactory as the goal was to filter out the extreme that will not occur during sea passage. A snapshot of the remaining data points are shown in appendix A.4

6.3.4. Filter On Energy Consumption

At last, the Energy consumption of the main engine was filtered. From the figures shown in figure 6.6 it can be concluded that multiple data points have unrealistic low energy consumption. Most probably this is caused by the re-sampling to 3 minutes of the Veinland data. Since the mean of the energy consumption is taken of a 3 min interval it is likely that multiple data points will not have an energy consumption during slowing down of the ship. But in this interval it can happen that the engine is started and the energy consumption is different than 0 which results in an unrealistic low mean energy consumption for that specific time interval. These inconveniences are filtered out by using the IQR on the energy consumption of the main engine. The eventual result of the snapshot of data points is shown in appendix A.5 [ht]

6.3.5. Filter Results

The data set after filtration on transverse speed, mean derivative of longitudinal speed, rudder angle and Energy consumption is further used for training and testing of the machine learning methods in this study. No filtration on the different motion measurement was performed in order to not drop the information captured in the motion outliers. These motion outliers will enhance the performance of the machine learning method on predicting fuel consumption in a heavy sea state. Therefore no further filtration is performed.

6.4. Data Scaling

In order to improve the accuracy of the machine learning algorithms the data is scaled such that all the data have the same magnitude. This to prevent that the weight factors must have large values which can lead to unstable estimations of the target value. In this research it is chosen to scale the features by using a Robust Scaler. This scaler uses the median, the first quantile and third quantile.

$$X_{Scaled} = \frac{X - \tilde{x}}{Q_3 - Q_1} \tag{6.25}$$

Where $\tilde{x}$ is the median of the feature, X is the original value, X_{scaled} is the scaled value, Q_3 and Q_1 are the third and first quantile respectively.

A Robust Scaler is chosen since most of the features are not normally distributed and extreme values are present for certain features. By using the median and the quantiles for a skewed distribution with outliers the scaling can be performed such that few data information is lost. Before scaling the data, the full data frame is split in two, 75% is used for training and 25% used for training and validation. In order to scale the data the training data is first fitted to the target data such that the algorithm can

learn the median and quantiles of every feature. The median and quantiles are then stored in internal memory and are used for transforming/scaling the original data set. Afterwards, the test set is also transformed by the internally saved median and quatiles learned by the training set. In such a way a scaled training and test set are obtained for creating accurate machine learning algorithms.

6.5. Data Overview

After the data processing and feature engineering were performed, all the data features were combined into one data set. The parameters used for feature engineering were replaced by the result of the operation with the selected features. The definitive features used for machine learning techniques are listed in appendix B.

After resampling and filtration, there are 4861 data points obtained during a measuring period of 2 weeks. Unfortunately, due to technical difficulties on board the vessel no extension of the measuring period was possible. Nevertheless, the amount of data was found satisfactory in order to perform the research. Since the vessel performs ocean passages which happens at high speeds for longer periods, there are considerably more data points in the higher consumption region than in the lower region of fuel consumption. The number of occurrences of the fuel consumption are shown in figure 6.7.

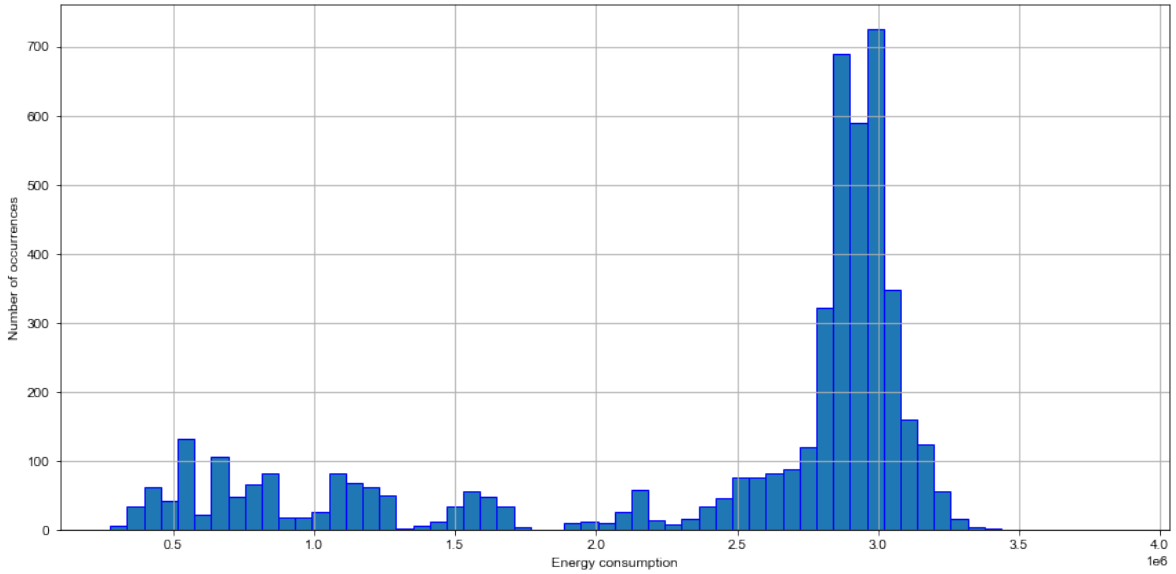

Figure 6.7: Data points distribution

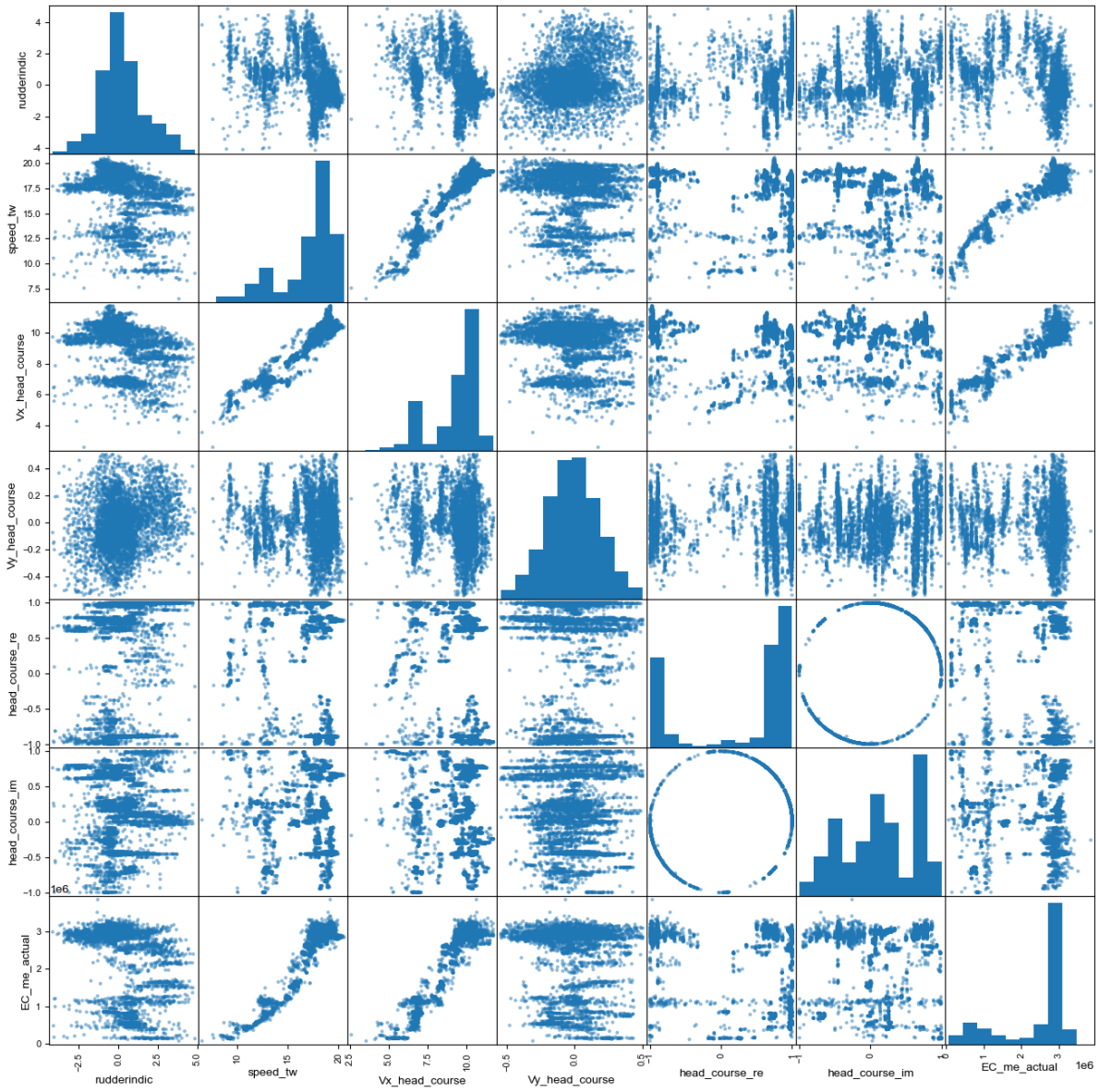

Figure 6.6: Scatter matrix of selected features after filtration of transverse velocity, mean derivative longitudinal velocity and rudder angle, label explanations are provided in appendix C

7

Machine Learning Algorithms

In this chapter the machine learning methods explained in chapter 3 are developed for the obtained data set. First the general method is explained and then the result of each individual machine learning algorithms is described. Based on the obtained result, a choice for best algorithm is made for further investigation.

7.1. General Method

In order to qualitatively determine and compare the performance of the chosen machine learning algorithms a fixed process is followed. Due to this fixed process only the quality of the machine learning algorithm influences the accuracy of the estimations. The process followed for each algorithm is called a pipeline. This pipeline is displayed in figure 7.1 and further discussed in this section.

Figure 7.1: Pipeline of process

As discussed in chapter 6.4 first the data is scaled such that each input feature has the same magnitude. After scaling, a process of feature selection will be performed. Since it might be possible features provide the algorithm wrong information or are just not useful, it is possible the learning algorithm will perform better using only part of the input features. In the pipeline, a choice will be made about the percentage of number of input features. These percentages are 100%, 90%, 80% or 75% of number of input features. Which specific features are selected are automatically chosen by the learning algorithm.

After the automatic feature selection process, the hyperparameters of the learning algorithm are determined. These are chosen by performing a Random Search and later a grid search to find the most optimal hyperparameters. The range of hyperparameters is first chosen arbitrarily while performing a random search. After the random search, the grid search is performed using specifically chosen and more accurately chosen values for the hyperparameters. For each combination of chosen hyperparameters and feature selection percentage a k-fold validation process is performed with 5 iterations. Where the training size is 75% of the full data set. After testing and validating the trained algorithm with the remaining data using the k-fold validation, error metrics are used to identify the quality of the specific algorithm. The hyperparameters of the algorithm with the best error metrics are saved and used to retrain and save the best algorithm.

Using this process for each algorithm discussed in chapter 3, the performance of each algorithm is determined and compared to the performance of alternative machine learning algorithms. Conclusions can then be made on quantitative performance indicators on which algorithms fits best for estimating fuel consumption for a digital twin.

7.2. Performance Of The Algorithms

In this section the details of the different learning algorithms are discussed together with the error metrics and results of the specific algorithm.

7.2.1. Linear Regression

Linear regression is the most basic model since there are no hyperparameters to tune. The only action required in pre-processing is the feature scaling and additionally, the automatic feature selection.

The performance of the Linear Regression Algorithm is summarized by the error metrics in table 7.1. This performance was achieved by using all the input features. No input feature was classified as superfluous.

Table 7.1: Error Metrics of Linear Regression

R^2	MAE	RMSE
0.98734	69595.30	96655.95

The mean absolute error and root mean squared error have a large magnitude. This magnitude must be placed in perspective to the predicted value, energy consumption, which is done in figures 7.2 and 7.3. In figure 7.2 the error in percentage of the true value is given in function of the true value. Most ideally these points must all be situated on the 0% axis which indicates there is no error. The second figure, figure 7.3 shows the predicted value in function of the true value. In case the learning algorithms predicts exactly the same values as the true values, a linear line will develop. This ideal line is shown in a red dashed line. The figures shows that in the regions of larger energy consumption, the error made is of greater magnitude. Nevertheless, percentage wise the error is smaller than compared to the predicted values of the lower energy consumption. Also in the mid range of fuel consumption the error in percentage remains stable, apart from one outlier. Unfortunately in the region of low energy consumption the relative error is large. Due to these large errors the Linear Regression algorithm is classified as not accurate enough for predicting energy consumption.

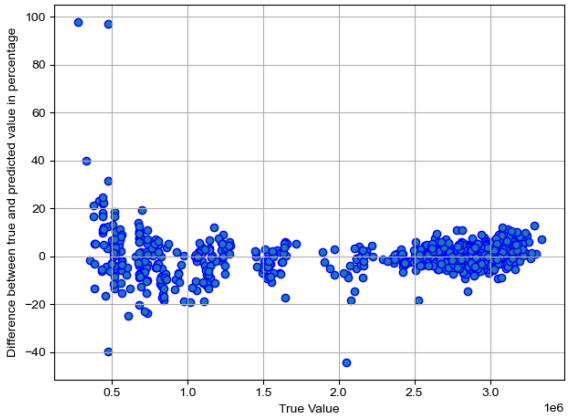

Figure 7.2: Error in function of the true value for LR

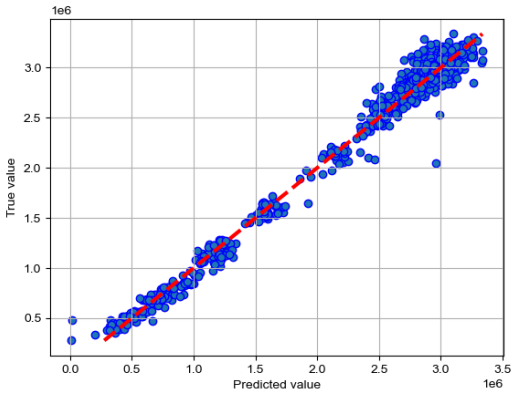

Figure 7.3: Predicted value in function of true value for LR

7.2.2. Ridge Regression

The learning algorithm using Ridge Regression is tuned with one hyperparameter, lambda (λ), as shown in equation 3.3. Together with the percentile of selected features it is concluded after a grid search that the best performance of Ridge Regression is reached when $\lambda = 0.005$ and 100% of the input features are used. These optimal hyperparameters result in the error metrics summarized in table 7.2.

In figures 7.4 and 7.5 again, a good overview of the estimated and the actual value is given. Not

much of a difference is observed compared to Linear Regression which is as expected. Ridge Regression only uses an extra weight factor but still remains linear. The use of the extra factor does not improve the results.

Table 7.2: Error Metrics of Ridge Regression

R^2	MAE	RMSE
0.98733	69602.04	96690.71

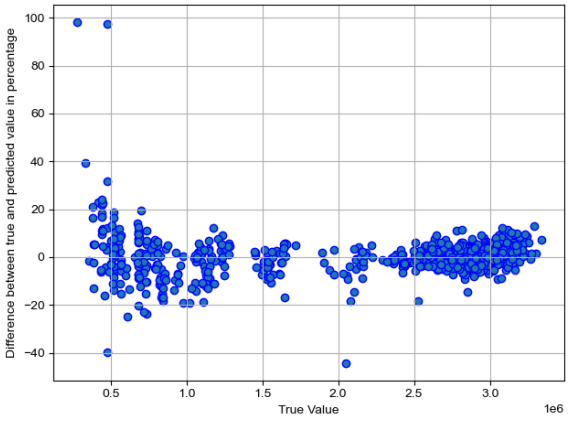

Figure 7.4: Error in function of the true value for RR

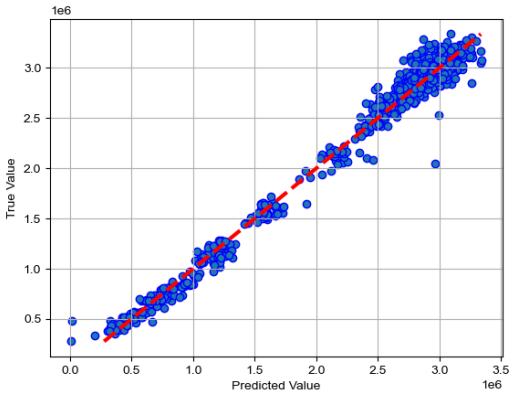

Figure 7.5: Predicted value in function of true value for RR

7.2.3. Lasso

Lasso is very similar to Ridge Regression. It uses the same mathematical formulation with that difference that for one term the absolute value is taken instead of the square, as shown in equation 3.4. Therefore Lasso also requires only one hyperparameter to tune, namely λ. The tuning of the hyperparameter after the random search and grid search resulted in the optimal value for λ at 0.0005. Together with the full set of input features, it gave the best performance for this learning algorithm. The error metrics which indicate the algorithm performance are listed in table 7.3. Since the method is similar to the Linear Regression and Ridge regression, the results are also similar. Also the same conclusion from figures 7.6 and 7.7 can be drawn.

Table 7.3: Error Metrics of LASSO

R^2	MAE	RMSE
0.98734	69597.01	96672.19

7.2.4. Supported Vector Regression

Supported vector Regression has multiple hyperparameters to tune. After performing the randomized search, it was concluded that a third degree polynomial with a constant value of 50 is the best fit for mapping the data points in higher dimensions. The function for this mapping is also called the kernel. This mapping results in the hyperplane that helps predicting the target value. The gamma value of SVR, which indicates the deflections in the hyperplane is optimal at 0.001. The C-value, the regularization parameter, of the algorithm determines the importance of each data point and how many data points may remain inside the margin. In fact, the C-value defines the weight of how much samples inside the margin contribute to the overall error. A large C-value results in a large penalty of samples inside the margin. For the training set established in this study, a C-value of 1200 is the most optimal. These optimal hyperparameters together with 90% of all the input features resulted in the learning algorithms which performs best. A summary of the tuned hyperparameters is given in table 7.4.

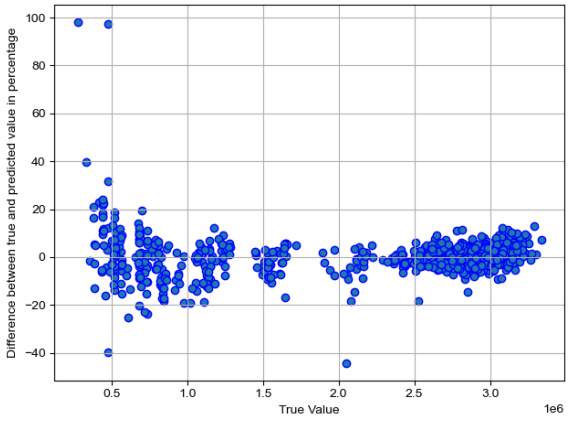

Figure 7.6: Error in function of the true value for Lasso

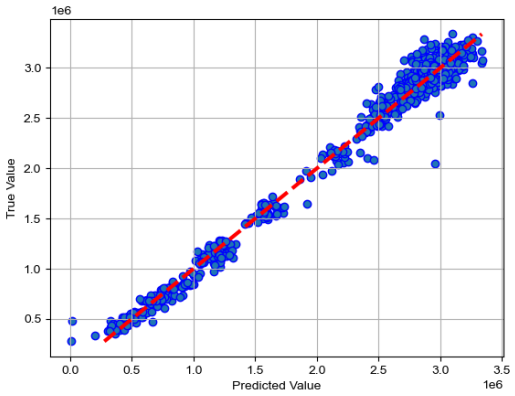

Figure 7.7: Predicted value in function of true value for Lasso

Table 7.4: Optimal hyperparameters of SVR

Hyperparameter	Value
Kernel	Polynomial
Degree	3
Coeff. 0	50
C	1200
Gamma	0.001
Percentile	90

With the optimal hyperparameters k-fold cross validation with 5 iterations is performed with a size for the training set of 75%. This results in the error metrics listed in table 7.5. A small drop in performance is observed compared to the previous methods. Based on the figures 7.8 and 7.9 more outlier causes this drop in performance. Therefore also the RMSE is higher but the MAE is lower compared to the linear models. Aside from the outliers it can be stated that again the relative error in the lower energy consumption region is higher compared to the relative error of the higher energy consumption.

Table 7.5: Error Metrics of Supported Vector Regression

R^2	MAE	RMSE
0.98676	64830.37	98857.74

7.2.5. Random Forest

Random Forest is an ensemble of different decision trees. Therefore one of the hyperparameters to tune is the number of decision trees from which the forest is built of. To determine how many trees are required, a simple experiment is performed. A Random Forest learning algorithm was used and trained each time with an extra decision tree in the ensemble. Each time the coefficient of determination was determined. The result of the experiment is shown in figure 7.10. In this figure an exponential development is observed. No further increase of coefficient of determination is observed when more than 30 decision trees are used in the random forest. Due to this conclusion the Random Forest algorithm is developed with 30 decision trees.

After the determination of number of decision trees, the hyperparameters of each individual tree are determined. These include the maximum number of layers (max layers) of which a decision can be made as well as the maximum number of nodes a tree can end on (max end nodes). Also, how many data points that must be classified in one of these last nodes (min samples end). The last considered hyperparameter determines the minimum number of data points which are required to split an inter-

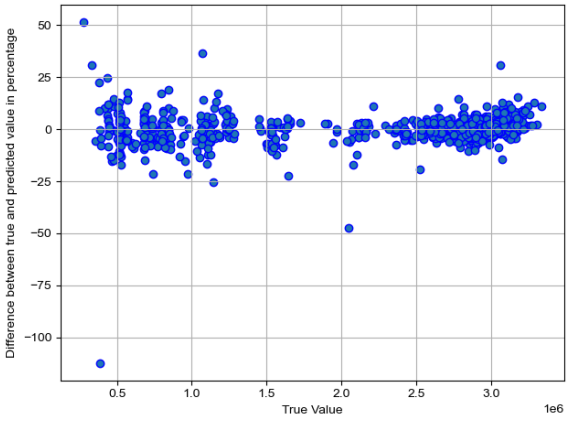

Figure 7.8: Error in function of the true value for SVR

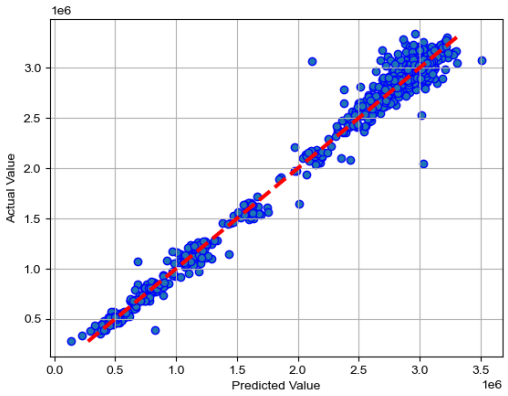

Figure 7.9: Predicted value in function of true value for SVR

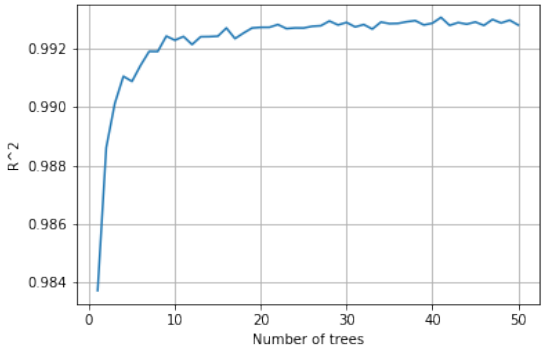

Figure 7.10: R^2 in function of number of decision trees

nal node (min samples split) [25]. The optimal values for the before mentioned hyperparameters are shown in table 7.6 which are determined after a thorough randomized search and grid search. Due to these hyperparameters together with 100% of the input features the random forest learning algorithm performs the most optimal. The error metrics of this learning algorithms are shown in table 7.7.

Table 7.6: Optimal hyperparameters of RF

Hyperparameter	Value
Max layers	15
Max end nodes	100
Min samples end	3
Min samples split	10
Percentile	100

Observing the error metrics, a great improvement in accuracy is made compared to the previous machine learning techniques. A possible reason is that in Random Forest no curve was tried to fit to the data. In Decision trees different scenarios can be distinguished in the data. This is exactly the case considering fuel consumption, since it is known the fuel consumption is highly correlated with the resistance, which on its turn is highly correlated with the wetted surface and speed trough water. Since the draft of the vessel often changes, different resistance curves exist and no unique line can describe the behaviour of the fuel consumption for different loading conditions. The accuracy of the Random Forest algorithm can be observed in figures 7.11 and 7.12, less significant outliers are observed in both of the figures compared to previous methods.

Table 7.7: Error Metrics of Random Forest

R^2	MAE	RMSE
0.99381	41976.64	67592.42

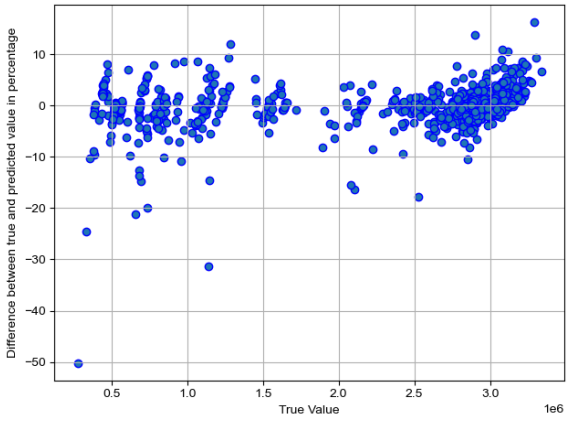

Figure 7.11: Error in function of the true value for RF

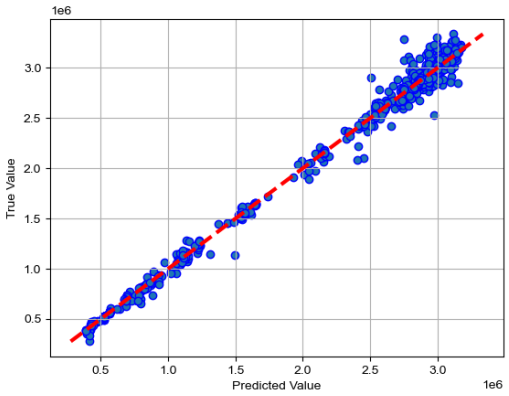

Figure 7.12: Predicted value in function of true value for RF

7.2.6. Extra Tree Regression

Because of the promising results obtained by the Random Forest machine learning algorithm, an Extra Tree Algorithm (XT) is also tested. This algorithm is based on a Random Forest algorithm. In the decision trees used in both RF and XT, different levels exist where separation of data is performed. This separation is based on a value which is determined during training of the decision trees. In a Random Forest this value is determined by calculating the local optimum in order to get the best accuracy. In an Extra Tree algorithm the value used to make the split is chosen more at random and not the optimum is taken. Due to more randomness in the different decision trees in the extra tree algorithm, a better generalization might be obtained. This should result in better error metrics.

Due to the resemblances between Random Forest and Extra Tree regressors the hyperparameters to tune are the same. The optimal hyperparameters obtained after random and grid search for the XT algorithm are shown in table 7.8. In this table also the quantity of used input features of the total data set is mentioned, namely 100%. The algorithm categorized 0% of the full data set input features as not required to get an accurate result.

Table 7.8: Optimal Hyperparameters of XT

Hyperparameter	Value
Max layers	20
Max end nodes	350
Min samples end	3
Min samples split	8
Percentile	100

The error metrics of the Extra Tree algorithm are shown in table 7.9. An improvement in error metrics is observed compared to Random Forest regression. In the figures 7.13 and 7.14 it clearly shows that predictions in the lower regions of energy consumption are more accurate. Furthermore, in figure 7.13 it can be observed that in the full range of energy consumption the learning algorithm performs equally well. There is no section of the energy consumption which performs better or worse, probably due to

the different scenarios a decision tree can distinguish.

Table 7.9: Error metrics of XT

R^2	MAE	RMSE
0.99484	37530.65	61715.13

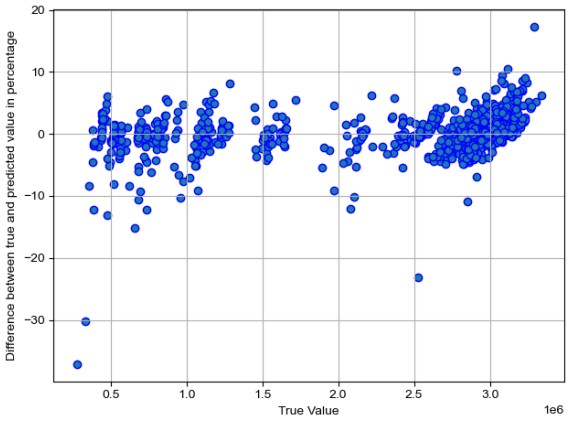

Figure 7.13: Error in function of the true value for XT

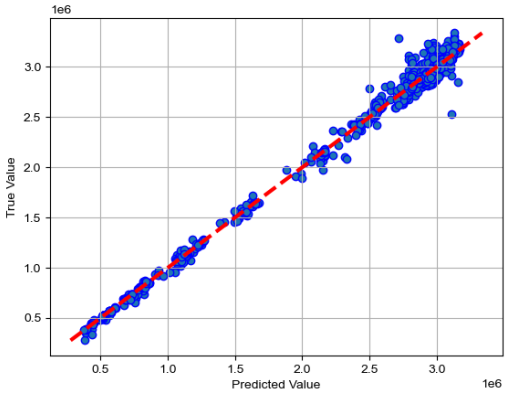

Figure 7.14: Predicted value in function of true value for XT

7.2.7. Artificial Neural Networks

In this section Neural networks are developed and evaluated. Based on the researches of Jeon, Tarelko, Hu and le [17, 31, 15, 19], the choice was made to develop Multi Layer Perceptron Artificial Neural Networks (MLP-ANN) with few hidden layers, pictured in figure 7.15. Additionally, a "Residual Neural Network" (ResNet) is developed (figure 7.16), again using few hidden layers. A ResNet network has a direct connection from the input layer to the hidden layers but also to a concatenation layer which collects two sources. One source is the input layer and the second source is the output of the neurons after being processed through the hidden layers. A ResNet has an advantage in estimating target values when simple (linear) relations exist between the input and target value. These simple relations are then covered by the direct connection to the concatenation layer and are not made too complex in the hidden layers.

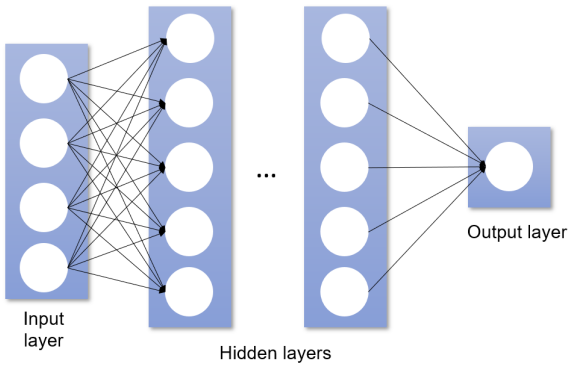

Figure 7.15: Conventional Architecture for ANN

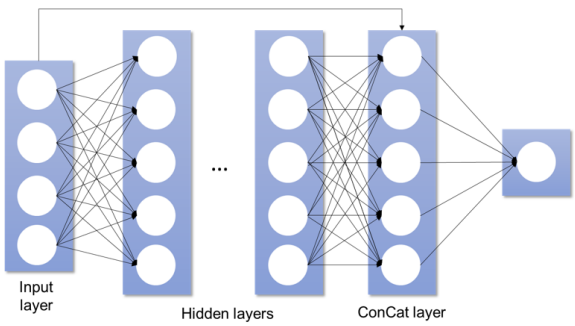

Figure 7.16: Residual Neural Network

With both the proposed architectures three different alternatives of the architectures are tested. These alternatives are:

- Conventional: the neural networks as shown in figure 7.15 and 7.16.

- Dropout: Behind the input layer and each hidden layer a dropout layer is added. This dropout layer will turn off random neurons during training such that specific neurons do not contribute in the network for one training iteration. When the network is fully trained and used, the drop out layer will have no influence anymore. The use of dropout layers should lead to a better generalization of the learning algorithm
- Pyramid: In case of multiple hidden layers the first hidden layers have more neurons compared to the next hidden layers. In this alternative also dropout layers are added.

For the different neural networks a random search and grid search was performed in order to find the optimal hyperparameters. the hyperparameters that were tuned are:

- Number of neurons in each layer
- Number of hidden layers
- Activation function
- The neuron extension of the first layer in case of the pyramid alternative
- The dropout rate of layers during training in case dropout layers were present
- The learning rate, this value controls how much the weights alternate in the neurons at each iteration of training
- Batch size which controls the number of samples used during one epoch of training.
- Selected Percentile, unfortunately it is very complex to perform an automatic features selection process with neural networks. This is also a large subject for research which is out of scope of this study. Therefore it is chosen to use the feature importances obtained from the random forest. Using the classification from the random forest algorithm, the neural networks are trained with 100%, 90% or 75% of the most important features according to Random Forest.

Training the networks and tuning the hyperparameters of each network led to the results shown in table 7.10. This table contains all the result of the different architectures and alternatives from which it can be concluded that a conventional neural network is the best alternative for estimating fuel consumption. Only the results of this neural network will be further discussed.

Table 7.10: Error Metrics for ANN alternatives

Alternative	R^2	MAE	RMSE
Conventional NN	**0.98857**	**60453.64**	**87156.87**
Conventional ResNet	0.98348	74447.87	100067.26
Dropout NN	0.98292	76745.80	102548.91
Dropout ResNet	0.98534	69985.21	95342.43
Pyramid NN	0.98265	75410.57	102467.29
Pyramid ResNet	0.98213	78056.89	103767.26

The configuration of the best conventional neural network consists of three hidden layers with each 231 neurons. The best activation function in the neurons is a ReLu function. For training a learning rate of 0.000283 was used and a batch size of 34. The best result was also achieved by using 75% all the input features which were ranked according to importance. The hyperparameters are also summarized in table 7.11. As expected, only a shallow neural network is required to obtain the best results. This indicates that no complex relationships are present between the different input features. This was also concluded in the researches performed using meteorological input features such as Petersen, Pedersen, Jeon and Tarelko. [27, 24, 17, 31].

In the figures 7.17 and 7.18 the accuracy of the estimation is shown. It is clear that the predictions of lower enrgy consumption has a larger relative error. This is not desirable. The negative error must be

Table 7.11: Hyperparameters for conventional neural network

Hyperparameter	Value
Number of layers	3
Number of neurons	331
Activation function	Relu
Learning Rate	0.000283
Batch size	24
Percentile	75

approximately the same throughout the full energy consumption interval. Nevertheless there is a clear improvement in comparison to other algorithms where a larger errors are made in regions of low fuel consumption.

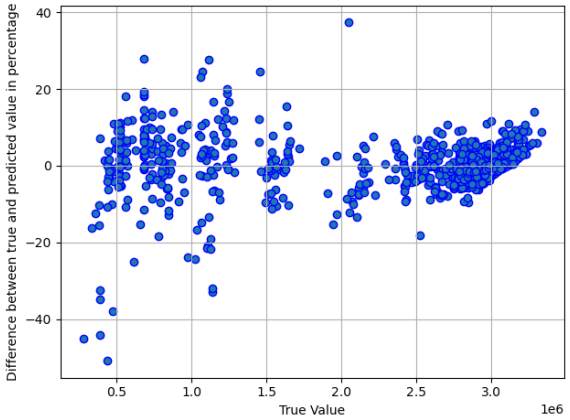

Figure 7.17: Error in function of the true value for Conv NN

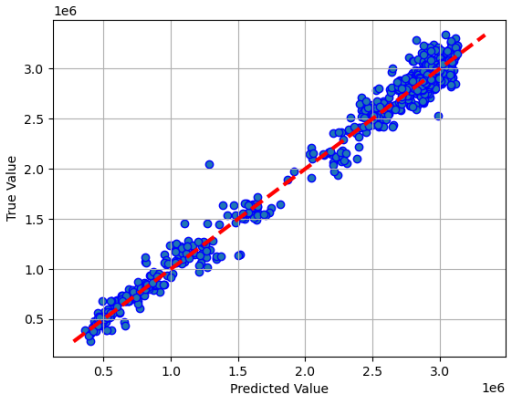

Figure 7.18: Predicted value in function of true value for Conv NN

7.3. Best Algorithm

From the results of Linear Regression, Ridge Regression and LASSO it can be concluded that there is no clear linear relationship between the input features and the energy consumption. These three algorithms showed no good performance as shown in table 7.12. Also from the graphs can be concluded that a higher relative error was made in the lower energy consumption regions. The same counts for the supported vector regression, despite the use of polynomials.

The result of the artificial neural network is not as expected. The accuracy of the neural network was expected to be higher than shown in table 7.12. Since the data only covers 2 weeks of recording few variation in data is observed. Therefore it is believed in case more data is used the accuracy of the neural network rises and might outperform the Extra Tree Regression algorithm since neural networks are a strong machine learning algorithms.

Nevertheless, with the data used in this study the Random forest and Extra Tree Regression Algorithm showed the best performances compared to the other tested algorithms, as can be seen in table 7.12. Since the Extra Tree Regression works similar to the Random Forest is it chosen to use and evaluate this algorithm further in next sections and in the digital twin. There is a believe that the tree algorithms work well because clear classifications are made in the algorithm according to different operating procedures.

In table 7.12 the error made in energy consumption is shown, additionally also the fuel consumption. To make the transition from energy to fuel consumption the lowest caloric value which is possible is

used, which is approximately 40.000 MJ/t.

Table 7.12: Result Summary

	R^2	MAE		RMSE	
unit	-	MJ/day	t/day	MJ/day	t/day
LR	0.98734	69595.30	1.740	96655.95	2.416
RR	0.98733	69602.04	1.740	96690.71	2.417
LASSO	0.98734	69597.01	1.740	96672.19	2.417
SVR	0.98676	64830.37	1.621	98857.74	2.471
RF	0.99381	41976.64	1.049	67592.42	1.690
XT	**0.99484**	**37530.65**	**0.938**	**61715.13**	**1.543**
Conv NN	0.98857	60453.64	1.511	87156.87	2.190

8

Best Algorithm Improvements

In this chapter the Extra Tree regression algorithm will be analysed such that further and more accurate results can be obtained. Only the Extra Tree regression algorithm is analysed since from the conclusion made in chapter 7 the Extra Tree regression algorithm was the most accurate. The improvements discussed in this chapter will not be of that magnitude such that alternative algorithms will outperform the Extra Tree regression algorithm.

In chapter 7 the motions are described by 8 moments of the non-derivative and first derivative motion distribution and 20 frequency bins, describing the spectrum of the time window. In the current chapter, it is investigated if the performance of the XT algorithms increases when the motions are described by either the different moments of higher order derived motion distributions or either purely the binned spectrum.

With the conclusion of how to correctly represent the motions, an analysis is done on the sampling frequency. This analysis makes it possible to validate or correct the choice concerning the sampling frequency made in chapter 6. At last, an analysis is performed which features are the most essential for the most efficient machine learning algorithm in order to predict accurate target values.

8.1. Motion Representation

This section investigates whether the initial choices for motion representation were correct or if improvements can be performed. In chapter 6 the choice was made to represent the motion data by 8 moments in total and the 20 frequency bins of the spectrum. From those 8 moments, 4 moments describe the non-derivative motion distribution and 4 moments describe the distribution of the first derivative of the motions. First an analysis is performed of what the influence is when extra moments describing higher order derivatives are added to the data set. Next, the moments describing the motions are classified if they are required in the data set or not.

Consecutive, the performance of the XT algorithm is investigated in case the motion are represented by only the frequency bins of the spectrum. The motion is then described with the spectrum of the time window. The value of the frequency bin is the sum of of the amplitudes falling inside the specific bin. In such a way a quantitative comparison can be made from which a conclusion can be drawn on how to represent motions for machine learning processes the best.

8.1.1. Motion Representation by Moments

In data pre-processing, discussed in chapter 6, the choice was made to take the non-derivative and first derivative motion distributions based on the paper of Petersen [27]. These distributions are described by the first four moments of the distributions. In this section it investigated if the performance of the prediction algorithms can be made more accurate by only using moments to describe the motions and if an increase or decrease in the motion derivative is beneficial. Also an analysis is performed on the importance of the moments describing the distributions.

Derivative Change

In this part the performance of XT algorithm is analysed using moments only to describe the motions. The pipeline presented in figure 7.1 is followed four times. One time with the four moments describing the distributions of the non-derivative motions. A second time with eight moments, four moments describing the non-derivative motion distribution and 4 moments describing the distribution of the first derivative of the motions. This setup is exactly the same as the initial setup without the frequency bins. The third and fourth time the pipeline is followed, a second and third derivative of the motions is determined respectively, which each time leads to an extra four moments that describe the second and third derivative motion distributions.

The best error metrics obtained of each of the four different runs trough the pipeline using different derivatives of the motions are shown in table 8.1 together with the initial result obtained when using 8 moments and 20 Frequency bins. In this table it can be observed that there is no large increase or decrease in performance using different alternatives for motion representations. The most significant increase in performance is observed from using no derivative motion to first derivative motion distributions. Adding the four moments describing the distribution of second derivative motions results in equal performance. Contrary, adding the moments describing the distributions of the third derivative motions decreases the performance of the algorithm a small portion due to overfitting. Notwithstanding, the performance of the algorithm using the first and second derivative motions are as good as the performance of the initial algorithm, using moments and spectra terms, since the error metrics are similar. Therefore it cannot yet be concluded which representation is the best. In section 8.1.3 a conclusion is made on which alternative is the best to represent motions after an investigation of moment importance.

Table 8.1: Derivative Change Results

	R^2	MAE	RMSE
Initial Case	**0.99484**	**37530.65**	**61715.13**
No derivative	0.99425	40747.68	65146.27
Up to 1 derivative	0.99482	37638.90	61807.46
Up to 2 derivative	0.99479	37630.26	62034.52
Up to 3 derivative	0.99463	38031.22	62972.10

Moment Importance

The use of the different moments describing the motion distributions is analysed in this section. In figures 8.1 to 8.4 the moment importance of each XT algorithm using an extra motion derivative is displayed. The explanation of the used abbreviations in the plots are provided in appendix C. The importance of each individual feature is determined using the Gini Importance which determines the total reduction of the criteria by using that specific feature [25]. The importance of each individual feature is then classified under the correct moment of each derivative and added together. Based on the four graphs which represent the importance of the different moments it can be concluded the third and fourth moment, skewness and kurtosis respectively, does not provide any vital information to the learning algorithm.

Throughout the plots 8.1 to 8.4 it can be concluded that only the first moment of the non-derived motion is useful for the algorithm. This great importance is dominated by the importance of the speed through water and the longitudinal speed over ground, Vx. The importance of each individual feature is further discussed in section 8.3. The first moment of the distributions describing the higher order derivatives is not useful. The importance of the mean value of the derivatives is always close to zero or exactly equal to zero.

The second moment of the distribution, standard deviation, describing the motion and derived motions is also of great importance to the algorithm. The standard deviation describes how much variation is present in the distribution. In figure 8.1 the greatest importance of the non-derivative standard deviation is observed. This importance decreases when more second moments of the distributions which describe derivative motions are added. This indicates that the algorithm prefers to use the standard deviation from the the derived motion distributions. Probably there is an overlap in information covered

in the non-derivative standard deviation and the higher order derived standard deviations.

Furthermore it is observed that first and second derivative combined always covers approximately 37% of the feature importance. This is a significant portion compared to the importance assigned to the third derivative of the motions. The addition of the second motion derivative clearly provides more clear information to the learning algorithm which is not provided through the third derivative. This statement can also be confirmed by the fact that there is a performance decrease in case three derived motion distributions are used, as shown in table 8.1. Most probably the use of three derived motion distribution results in redundant and noisy features which cover an overlap of information and produces less clear information to the learning algorithm which causes overfitting that leads to accuracy decrease .

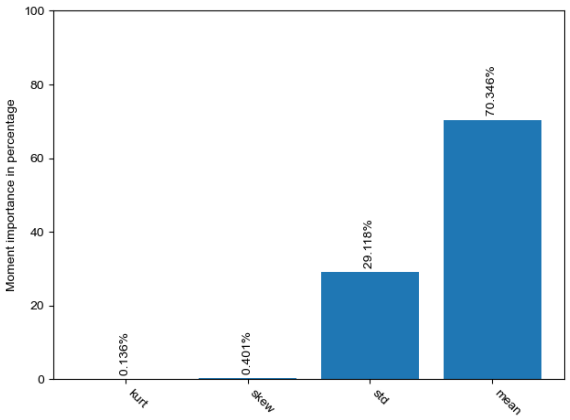

Figure 8.1: Moment importance according to XTR, sampling frequency of 20/hour, using no derivative features.

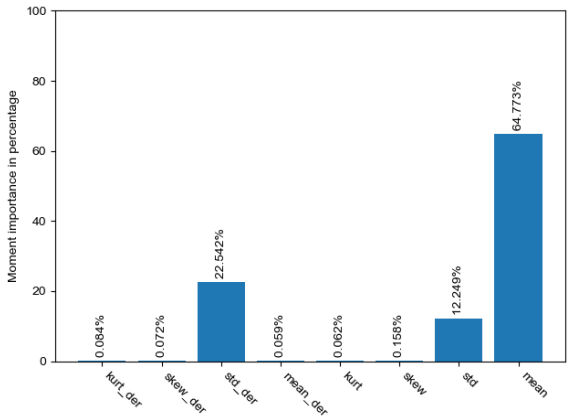

Figure 8.2: Moment importance according to XTR, sampling frequency of 20/hour, using no and first derivative features.

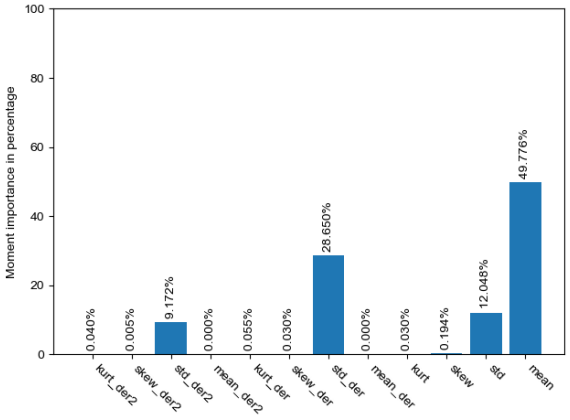

Figure 8.3: Moment importance according to XTR, sampling frequency of 20/hour, using no, first and second derivative features.

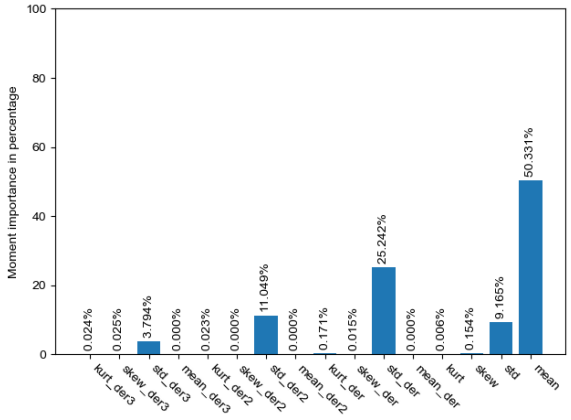

Figure 8.4: Moment importance according to XTR, sampling frequency of 20/hour, using no, first, second and third derivative features.

8.1.2. Motion Representation by Spectra

The results of the XT machine learning algorithm with motion data which is only represented by the amplitudes of the frequency bins of the spectrum concerning a time window of 3 min are discussed in this section. The representation of the motions and derived motions by moments is omitted from the data set. From the original signal a binned spectrum is created which is established by performing a Fourier analysis as discussed in chapter 6.

The initial choice was made to use 20 intervals or bins which are equally distributed until a frequency of

0.4 Hz. Since now only the spectrum is used it is also investigated if a larger or smaller number of bins results in better performance of the XT algorithm. The pipeline shown in figure 7.1 is again followed to develop the machine learning algorithm. The pipeline is followed 6 times each time with a higher amount of bins. The used amount of intervals dividing the frequency range up to 0.4 Hz are 2, 5, 10, 20, 30, 50 for each motion, meaning Roll, Pitch, accelerations around 3 axis and angular velocities around 3 axis. Additionally the amplitude of frequency 0, the mean, is also added as extra feature. Frequency 0 is not included in the intervals of the different bins. To summarize, each motion is described by the number of bins plus the amplitude of frequency 0.

The consequence on the performance of the XT algorithm due to the change in intervals of the spectra is listed in table 8.2. In the table it can be observed that the accuracy and thus performance of the algorithm decreases in case a large amount of bins are used. Using only 2 bins, the fewest amount tested, leads to the best error metrics. Using a larger amount of bins leads to overfitting of the training data and results therefore in worse error metrics.

Table 8.2: Result of change in number of frequency bins

	R2	MAE	RMSE
2 bins	**0.99497**	**39096.55**	**66151.55**
5 bins	0.99490	40289.38	66612.75
10 bins	0.99478	40388.04	67384.38
20 bins	0.99480	40860.06	67291.80
30 bins	0.99484	41055.52	67045.26
50 bins	0.99462	41995.79	68409.86

8.1.3. Concluding Motion Representation

A choice on how to represent the motion such that the most accurate results are obtained is made in this section. In table 8.3 the best performance of the XT learning algorithm from sections 8.1.1 and 8.1.2 are displayed. The coefficients of determinations are all comparable. In contrary, it is observed that the algorithm where the motions are represented by only spectra has highest MAE and RMSE. This indicates that more great errors are observed during the prediction of the test values. Therefore the algorithms which uses the motions represented by spectra are classified as least accurate. The other algorithms have all comparable error metrics. No conclusion can be made yet on which motion representation is the best out of the remaining three.

Table 8.3: Summary of error metrics of best motion representation alternatives

	R2	MAE	RMSE
By moments & spectra (initial)	0.99484	37530.65	61715.13
By spectra	0.99497	39096.55	66151.55
By moments, up to first derivative	0.99482	37638.90	61807.46
By moments, up to second derivative	0.99479	37630.26	62034.52

Based on the discussion of figures 8.1 to 8.4 a choice is also made on how the features must be represented in order to achieve the best algorithm. From the figures it is concluded that from the non-derivative motion distribution, the fist and second moment is required as well as the second moment of the higher order derived motions. Therefore XT algorithms are again developed but using only the most important moments. The pipeline in figure 7.1 is followed four times. Three times with the remaining alternatives of table 8.3. The fourth time the pipeline is followed, the used data consist out of the most important moments up to the first derivative and the spectra which is represented by only 2 bins. Only 2 bins are used since the conclusion of table 8.2 was to use as less bins as possible. The performance metrics of these are shown in table 8.4. The metrics indicate that the removal of the superfluous features again leads to an increase in accuracy of all the algorithms, less overfitting occurs. The alternative using only reduced moments up to the first derivative has gained the most performance. The error metrics of this alternative are also considerably lower than the other tested motion representations. Therefore

it is concluded that the best motion representation is by using only moments up to the first derivative. More specifically only the first and second moment of the non-derivative motion distribution and the second moment of the derived motion distribution must be used to obtain the most accurate result.

Table 8.4: Error metrics of best motion representation alternatives, using reduced moments

	R2	MAE	RMSE
By reduced moments & spectra, 20 bins (reduced initial)	0.99493	37253.96	61197.40
By reduced moments & spectra, 2 bins	0.99500	36958.71	60737.08
By reduced moments, up to first derivative	**0.99508**	**36067.50**	**60263.99**
By reduced moments, up to second derivative	0.99480	36919.87	61949.11

8.2. Sampling Frequency Change

While re-sampling the different data sources in chapter 6.1 the choice was made for a 3 min time window which corresponds to a measuring frequency of 20/hour. In this chapter the measuring frequency is alternated to validate if the initial choice was the correct choice or another time span is better to obtain the most accurate results. The alternation of time window is only performed with the best learning algorithms, Extra Tree Regression using the alternatives where the motions are represented by the reduced moments up to one motion derivative. No strong change in accuracy is expected due to time span change such that another algorithm type could outperform the Extra Tree regression. The choice was made to test the sampling frequencies of 60/hour, 20/hour, 12/hour, 6/hour, 3/hour, 1.5/hour, 1/hour. For each change in time window the pipeline presented in figure 7.1 is followed.

From the results shown in table 8.5 it can be noticed that the different sampling frequencies have a reasonable accuracy. The performance of each algorithm is very well quantified by the coefficient of determination thus in choosing the best algorithm this value dominates. The sampling frequencies of 20/hour and 12/hour provides equally good algorithms according to the coefficient of determination. But according to the RMSE a great improvement occurs when using a sampling frequency of 12/hour using one motion derivative. A smaller RMSE error indicates less extremely large errors while predicting the target values of the test set, which can also be validated from the figures 8.5 and 8.6.

Table 8.5: Error metrics of Time span change

Sampling Freq. [/hour]	R^2	MAE	RMSE
60	0.99279	43854.32	77532.65
20	**0.99508**	36067.50	60263.99
12	0.99449	**35492.07**	**55578.87**
6	0.98551	36921.71	67848.33
3	0.98595	32739.48	51200.47
1.5	0.98701	35924.53	50714.02
1	0.93245	43113.15	65428.61

For the measuring frequency of 3/hour a great decrease in MAE and RMSE can be noticed with a lower coefficient of determination compared to a frequency of 12/hour. Due to the size of the data set used, the test set used for evaluating the algorithms becomes too small to represent the full data set. Therefore the values of MAE and RMSE error are unreliable for the low sampling frequencies. Due to the lower number of test values a smaller probability exist that there will be an outlier. Thus, from table 8.5 it can be concluded that the initial choice of a sampling frequency of 20/hour with one derivative was good. Although a sampling frequency of 12/hour using one motion derivative leads to better Extra Tree algorithm performance. In order to create a digital twin, a sampling frequency of 12/hour with one motion derivative will be used. In such a way, more sampling points from the motion data are used to determine the first and second moment of the non-derivative and first derivative distribution of the multiple measured features.

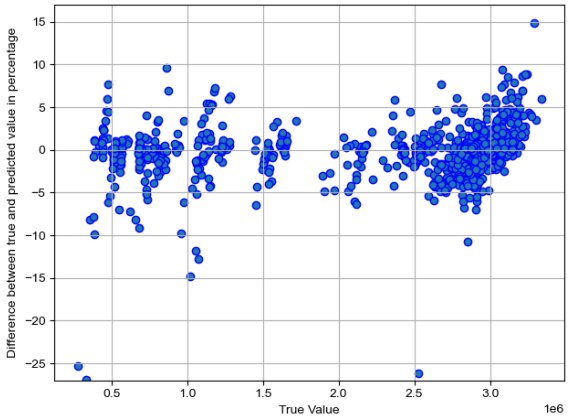

Figure 8.5: True value in function of the error for XT for a sampling frequency of 20/hour, using one derivative

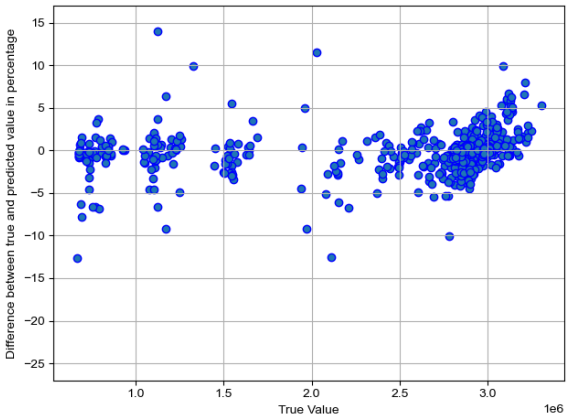

Figure 8.6: True value in function of the error for XT for a sampling frequency of 12/hour, using one derivative

8.3. Feature Importance

In this section the importance of the used features is determined. With this importance an analysis on the error metrics of the XT algorithm is performed on how many features are required according to the sum of feature importance. In this way needles features are identified and removed whilst still obtaining accurate results. Also a time period is discussed with the most important features describing motions in order to see relations between different features.

8.3.1. Feature Importance Determination

The features used to train the XT learning algorithm, using a sampling frequency of 12/hour and one motion derivative, are rated in how important they are for the algorithm. The importance rating is done by the XT learning model itself, based on the (normalized) total reduction of the criterion brought by that feature. Also called the Gini importance [25]. Where an importance of 1 indicates that the target values can fully be predicted by this feature and 0 means not used at all. However if a feature importance is low this does not mean the feature is uninformative but it rather indicates another feature covers equal information [3]. The features importance does not indicate the relation between the target value and specific feature.

Based on the feature importance displayed in figure 8.7, the speed through water is of great importance for the learning algorithm. This is no surprise since from literature it is known there is a third order relation between the energy consumption and the vessel speed through water. Also the speed over ground (Vx_head_course) is of great importance for the learning algorithm.

From the vessel motions, the values that the algorithm finds the most useful are the acceleration and more specifically the variation of accelerations around the transverse and longitudinal ship axis. The features describing the operational status of the ship, the displacement, trim, GM and rudder angle are also high in the ranking of importance. Nevertheless the magnitude of the importance of features describing the operational status of the vessel is small.

8.3.2. Developing XT According to Feature Importance

The XT regression algorithm is again developed with only part of the data set in order to validate if all the used features are required. The size of data set used is determined by the cumulative feature importance. The XT algorithm is developed with features which have an importance sum of 80%, 85%, 90%, 95%, 98%, 99% and 99.5%. The selected features are determined by the features importance which are ranked from very important to less important. In case only 80% of the feature importance is used, the features up to the standard deviation of acceleration around the longitudinal axis (std_accX)

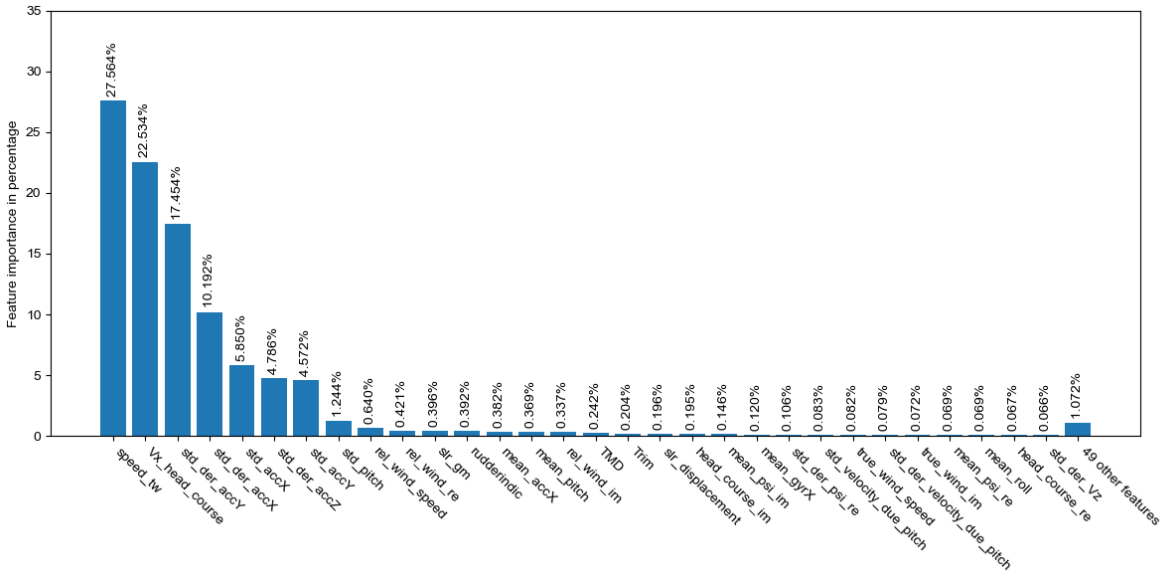

Figure 8.7: Feature importance according to XTR, sampling frequency of 12/hour, using non and first derivative features. Label explanations are provided in appendix C.

are used, shown in figure 8.8. For a cumulative sum of 85%, the features up to the standard deviation of the derived motion distribution of the accelerations around the vertical axis are used and so on for the remaining sum of feature importance.

Since only a small amount of captured data is available, no large variations in operational features are present. Therefore the operational features such as speed through water, displacement, GM, draft and trim are always taken into the sum of the cumulative feature importance because the expectation is that with a larger variation in operational features, these feature will become more important. The sum of the selected operational features is equal to 28.602% as can be seen in figure 8.8.

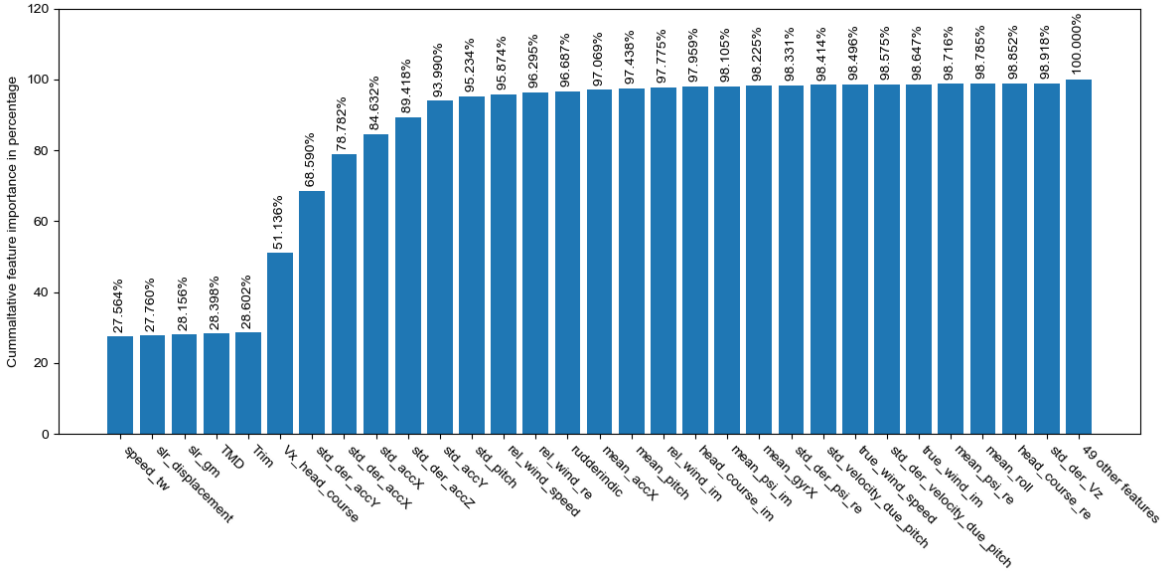

Figure 8.8: Cumulative feature importance, Label explanations are provided in appendix C.

In table 8.6 the error metrics obtained after following the pipeline shown in figure 7.1 for the seven

sums of feature importance. Also the results of the algorithm obtained in section 8.2(corresponds with a sum of feature importance of 100%), which was the best algorithm so far, are shown in this table. A clear trend is observed that if the cumulative sum of feature importance becomes larger, the error metrics also become better. This is true until a sum of feature importance of 99% is reached. From this point onward the algorithm starts overfitting which lead to less accurate results. Therefore in order to obtain the most accurate results possible only the features that have cumulative sum of 99% must be used. This is beneficial for CMB since less information from the vessel is required and less data pre-processing is necessary in order to obtain accurate results. This will lead to less data transfer from the vessel to shore, which eventually leads to smaller internet costs. The features used in these sum of features importance of 99% are listed below.

- Speed through water
- Displacement of the vessel
- GM of vessel
- Draft of the vessel
- Trim of the vessel
- Longitudinal speed
- Standard deviation of first derivative distribution describing acceleration around transverse axis
- Standard deviation of first derivative distribution describing acceleration around longitudinal axis
- Standard deviation of non-derivative distribution describing acceleration around longitudinal axis
- Standard deviation of first derivative distribution describing acceleration around vertical axis
- Standard deviation of non-derivative distribution describing acceleration around transverse axis
- Standard deviation of non-derivative distribution describing pitch
- Relative wind speed
- Real part of complex relative wind direction value
- Mean rudder angle
- Mean of non-derivative distribution describing acceleration around longitudinal axis
- Mean of non-derivative distribution describing pitch
- Imaginary part of complex relative wind direction value
- Imaginary part of complex heading value
- Mean of non-derivative distribution describing imaginary part of complex GPS-course heading value
- Mean of non-derivative distribution describing angular velocity around longitudinal axis
- Standard deviation of first derivative distribution describing real part of complex GPS-course heading value
- Standard deviation of non-derivative distribution describing motion-sensor velocity due to pitch
- True wind speed
- Standard deviation of first derivative distribution describing motion-sensor velocity due to pitch
- Imaginary part of complex true wind direction value
- Mean of non-derivative distribution describing real part of complex GPS-course heading value
- Mean of non-derivative distribution describing roll
- Real part of complex heading value
- Standard deviation of first derivative distribution describing vertical velocity
- Standard deviation of first derivative distribution describing pitch
- Standard deviation of non-derivative distribution describing angular velocity around transverse axis
- Real part of complex true wind direction value

Although the results with sum of feature importance of 99% are the most accurate, the other alternatives have similar but slightly worse error metrics. For practical uses it is not required for using 99% of feature importance. The accuracy obtained with a sum of 80% is still accurate enough for the practical use since CMB will only decide to do a clean or a check of the hull in case the fuel/energy consumption rises with more than 30%. The accuracy of the algorithm using the features with a feature importance sum of 80% has an accuracy of 1% to 10% depending on the vessel speed through water, Assuming a calorific value of 40000 MJ/ton

Table 8.6: Error metrics of XTR algorithms using only part of the data set according to feature importance

Feature Importance Sum	R2	MAE	RMSE
80%	0.99464	36186.48	54848.28
85%	0.99468	35373.78	54641.90
90%	0.99474	35683.85	54326.97
95%	0.99462	35352.90	54943.07
98%	0.99478	34654.45	54110.89
99%	**0.99480**	**34442.38**	**53992.32**
99.5%	0.99467	34508.20	54693.45
Reduced moments, up to one derivative, 12/hour, 100%	0.99449	35492.07	55578.87

8.3.3. Important Features Over Time Period

In figure 8.9 a time period of the four most important features together with the mean rudder angle and target value, energy consumption, is shown. In this figure it can be noticed that the development of the variations (standard deviations) of the derived accelerations follows the development of the energy consumption curve although the vessel speed through water drops. This indicates that the accelerations around the longitudinal and vertical axis consume energy and thus diminishes the vessel speed with and increase in energy consumption. The rolling and pitching is definitely the cause of weather conditions since no large rudder angles are observed. Generally it can be stated that if the variations in acceleration around the longitudinal or transversal axis increases, the energy consumption also increases. The exact relations between the variations of both the accelerations is subject for a different research.

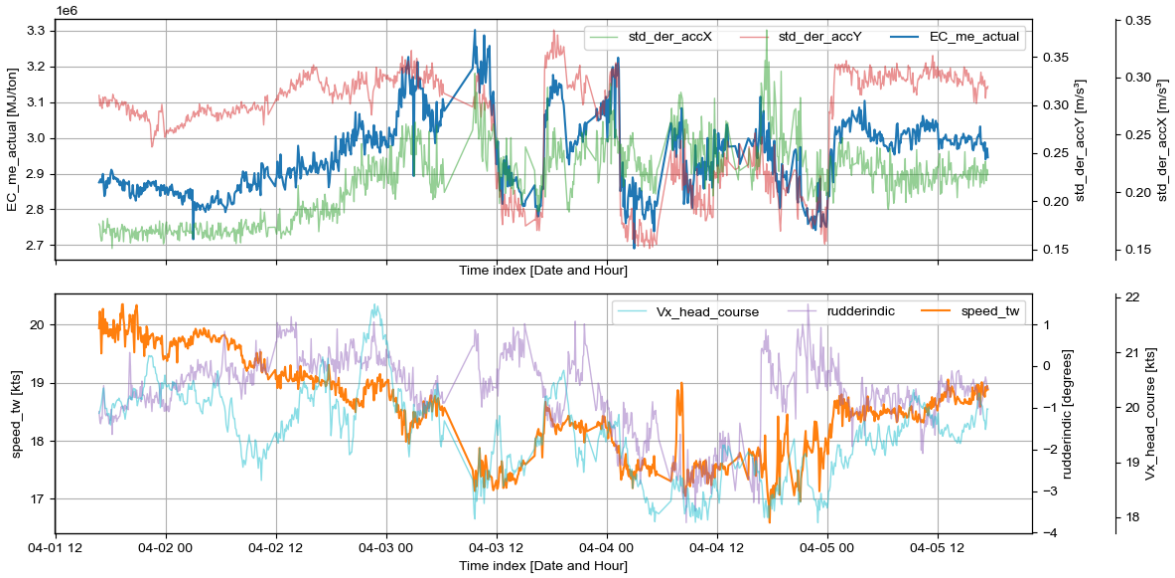

Figure 8.9: Time period of four most important features with rudder angle and target value. Label explanations are provided in appendix C.

8.4. Concluding Best algorithm Improvements

The best Extra Tree algorithm after improvements makes use of a sampling frequency of 12/hour and uses only the standard deviation of the first derivative but also the mean and standard deviation of the non-derivative for motion representation. The accuracy can furthermore be increased by using only features which have sum of feature importance of 99%. The MAE and RMSE in energy consumption of the best XT algorithm is translated in fuel consumption in table 8.7. This is done by making use of

the minimum calorific value possible of fuel oil, namely 40000 MJ/ton. Depending on the speed through water of the vessel the error in percentage changes between 1% to 10%. Since CMB will only perform a cleaning of the hull or check for fouling in case the consumption rises with more than 30%, this error is found satisfactory. Therefore the extra tree regression algorithm using a sampling frequency of 12/hour and one motion derivative is the best algorithm for representing the real vessel as a digital twin.

Table 8.7: Error metrics of best XT algorithm (12/hour and 1 derivative) in t/day using features with a importance sum of 99%

R^2	MAE	RMSE
0.99480	0.881	1.350

Due to the limited amount of data that was available no algorithm could be made for determining over-consumption of the vessel. For this algorithm the data of a full voyage had to be recorded and may not be used for training the machine learning techniques. There was only data present from one voyage of 12 days. With the extra data the digital twin made out of the in this research developed XT learning algorithm could predict the fuel consumption of a clean ship. These prediction could then be compared to the actual fuel consumption. As such a comparison of predicted and actual fuel consumption could have been made to determine the vessel over-consumption. Unfortunately, this could not be established due to the amount of available data. Whenever the technical difficulties on board the vessel are resolved, the fuel comparison program must still be made.

III

Closure

9 Conclusion

In this research different machine learning algorithms were developed to estimate vessel fuel consumption for base of the digital twin. The learning algorithms use a data set which consists of operational data and vessel motion data. The use of motion data is the new part of this research. In previous studies weather data coming from third parties were used to estimate vessel fuel consumption using machine learning techniques.

The data set used is a combination of Veinland data which contains navigational and engine related data, Noon Report which contains the static features of the vessel and motion data. At first the choice was made to represent the motions of the vessel by the four moments of the non-derivative and first derivative motion distributions and a spectrum which consisted out of 20 intervals until a frequency of 0.4 Hz. The moments and spectra were determined for every 3 min of measuring data. This 3 min corresponds with a sampling frequency of 20/hour.

Using these initial parameters a thorough research was performed on which machine learning algorithm has the best accuracy. After investigating the Linear Regression, Ridge Regression, LASSO, Supported Vector Regression, Random Forest and Artificial Neural network, the Random forest algorithm came out as best. Additionally, an alternative of Random Forest was tested namely Extra Tree Regression, which resulted in the most accurate results observed of all algorithms.

Table 9.1: Parameters of best algorithm

	Parameter
Algorithm	Extra Tree Regressor
Sampling Frequency	12/hour
Motion Derivatives	up to 1
Spectra	No
Moments	First and second

Using the Extra Tree algorithm efforts were performed in order to improve the accuracy of the algorithm. It was concluded that the Extra Tree algorithm using only one motion derivative and a time span of 5 min, which corresponds with a sampling frequency of 12/hour, resulted in the most accurate prediction of target values. In the data set used while training this algorithm, the motions are represented by only using the first moment of the non-derivative and the second moment of both the non-derivative and first derivative. The use of spectra was not beneficial for the accuracy of the learning algorithm. An overview of the best algorithm is given in table 9.1. Eventually, the conclusion could be made that with a sum of feature importance equal to 99% the performance of the XT algorithm is the best. The sum of feature importance was obtained by first ranking features from most important to less important and then summing up all the feature importance until 99% is reached. The error metrics of this algorithm are given in table 9.2.

Table 9.2: Performance indicators for best algorithm

	R2	MAE		RMSE	
Unit	-	MJ/day	t/day	MJ/day	t/day
Value	0.99480	34442.38	0.881	53992.32	1.350

The error made expressed in percentage is 1% to 10% depending on the vessel's speed through water. Since CMB performs a check or a cleaning of the hull and appendages in case the energy consumption rises with more than 30%, the accuracy of the machine learning algorithm is found satisfactory. Thus, the Extra Tree regression algorithm using a sampling frequency of 12/hour and a data set where the motions are represented by the second moment of the non-derivative and first derivative on top of the first moment of the non-derivative is considered as accurate enough for base of the digital twin.

Unfortunately, due to the limited amount of data available it was not possible to use the developed algorithm as a digital twin for estimating fuel over-consumption of the vessel. Because of technical difficulties on board the vessel, only 2 weeks, one sea voyage, of recording were possible. All this data was used to train, test and validate the algorithms. No more data was available of different voyages to make a model that determines the over-consumption of the vessel.

9.1. Recommendations

In order to use the in this research developed Extra Tree machine learning algorithm, more data of different voyages must be captured. With more data a program can be written that compares the real fuel consumption to the estimated fuel consumption by the learning algorithm. How this must be performed is a subject for a follow-up research. This study could then be combined with a comparison study of the Extra Tree algorithm trained by data which contains motion data and trained by data which contains meteorological data as replacement for the motions.

The performance of artificial neural networks were slightly disappointing in this research. It is strongly believed a better accuracy can be obtained using ANN compared to the performance presented in this project. Due to the extremely complex nature of neural networks, it takes more time and specialization to find the optimal network architecture and to tune its hyperparameters. Therefore,a study purely focused on the use of artificial neural networks with the data presented in this study might overcome this inconvenience. It might also be possible ANN are to powerful for usage in this research topic. Nevertheless, also more data is required for further development of ANN.

For CMB it is proven that machine learning techniques can be used in order to determine energy consumption or fuel consumption accurately. In order to make these accurate predictions, the measured accelerations around the three axes are the most important motions. Together with the motions, the speed through water, over ground and some static features are the most important to make predictions of the energy consumption. These features show a direct relation with the energy consumption of a seagoing vessel. In case only these features are being used, the accuracy of energy consumption predictions are still good enough for practical use.

9.2. Ending

The use of more precise motion data in stead of third party meteorological data for black-box machine learning algorithms is tested in this research. The most accurate result is obtained by representing the motions by using the first and second moment of the non-derived motion distribution and the second moment of the first derived motion distribution. These distributions resulted from a measuring time window of 5 min. The practical use of the machine learning algorithm must still be determined. This can be performed in case data of multiple voyages are available.

Bibliography

[1] Martín Abadi et al. “Tensorflow: A system for large-scale machine learning”. In: *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*. 2016, pp. 265–283.

[2] Misganaw Abebe et al. “Machine learning approaches for ship speed prediction towards energy efficient shipping”. In: *Applied Sciences (Switzerland)* 10.7 (2020). ISSN: 20763417. DOI: 10.3390/app10072325.

[3] Sarah Guido Andreas C Müller. *Introduction to Machine Learning with Python*. O’Reilly, 2016.

[4] Aurélion Gérond. *Hands-On Machine Learning with Scikit-Learn, Keras & Tensorflow*. Vol. 2. O’Reilly, Sept. 2019.

[5] Francois Chollet et al. *Keras*. 2015. URL: https://github.com/fchollet/keras.

[6] Patrick R Couser et al. “Artificial Neural Networks for Hull Resistance Prediction”. In: *COMPIT 04*. 2004.

[7] Nastia Degiuli et al. “Increase of ship fuel consumption due to the added resistance in waves”. In: *Journal of Sustainable Development of Energy, Water and Environment Systems* 5.1 (2017). ISSN: 18489257. DOI: 10.13044/j.sdewes.d5.0129.

[8] Konstantin Eckle and Johannes Schmidt-Hieber. “A comparison of deep networks with ReLU activation function and linear spline-type methods”. In: *Neural Networks* 110 (2019). ISSN: 18792782. DOI: 10.1016/j.neunet.2018.11.005.

[9] Andrea Farkas, Nastia Degiuli, and Ivana Martić. “An investigation into the effect of hard fouling on the ship resistance using CFD”. In: *Applied Ocean Research* 100 (2020). ISSN: 01411187. DOI: 10.1016/j.apor.2020.102205.

[10] Gartner. *Definition of Digital Twin*.

[11] Christos Gkerekos, Iraklis Lazakis, and Gerasimos Theotokatos. “Machine learning models for predicting ship main engine Fuel Oil Consumption: A comparative study”. In: *Ocean Engineering* 188 (2019). ISSN: 00298018. DOI: 10.1016/j.oceaneng.2019.106282.

[12] K. Grabowska and P. Szczuko. “Ship resistance prediction with Artificial Neural Networks”. In: *Signal Processing - Algorithms, Architectures, Arrangements, and Applications Conference Proceedings, SPA*. Vol. 2015-December. 2015. DOI: 10.1109/SPA.2015.7365154.

[13] Trevor et. all. Hastie. *Springer Series in Statistics The Elements of Statistical Learning*. Vol. 27. 2. 2009.

[14] Holltrop J and Mennen G. “An Approximate Power Prediction Model”. In: (1982).

[15] Zhihui Hu et al. “Prediction of fuel consumption for enroute ship based on machine learning”. In: *IEEE Access* 7 (2019). ISSN: 21693536. DOI: 10.1109/ACCESS.2019.2933630.

[16] Hafizul Islam and Guedes Soares. “Effect of trim on container ship resistance at different ship speeds and drafts”. In: *Ocean Engineering* 183 (2019). ISSN: 00298018. DOI: 10.1016/j.oceaneng.2019.03.058.

[17] Miyeon Jeon et al. “Prediction of ship fuel consumption by using an artificial neural network”. In: *Journal of Mechanical Science and Technology* 32.12 (2018). ISSN: 1738494X. DOI: 10.1007/s12206-018-1126-4.

[18] Hans Klein Woud and Douwe Stapersma. *Design of Propulsion and Electric Power Generation Systems*. IMarEst, 2002.

[19] Luan Thanh Le et al. “Neural network-based fuel consumption estimation for container ships in Korea”. In: *Maritime Policy and Management* 47.5 (2020). ISSN: 14645254. DOI: 10.1080/03088839.2020.1729437.

[20] Leifur Th Leifsson et al. “Grey-box modeling of an ocean vessel for operational optimization”. In: *Simulation Modelling Practice and Theory* 16.8 (2008). ISSN: 1569190X. DOI: 10.1016/j.simpat.2008.03.006.

[21] M. Liu et al. “Voyage performance evaluation based on a digital twin model”. In: *IOP Conference Series: Materials Science and Engineering*. Vol. 929. 1. 2020. DOI: 10.1088/1757-899X/929/1/012027.

[22] NOAA. *National Data Bouy Center*.

[23] I. Ortigosa, R. López, and J. García. “Prediction of total resistance coefficients using neural networks”. In: *Journal of maritime research.* 6.3 (2009). ISSN: 1697-4840.

[24] Benjamin Pjedsted Pedersen and Jan Larsen. “Prediction of Full-Scale Propulsion Power using Artificial Neural Networks”. In: *8th International Conference on Computer and IT Applications in the Maritime Industries*. 2009.

[25] Fabian Pedregosa et al. “Scikit-learn: Machine learning in Python”. In: *Journal of machine learning research* 12.Oct (2011), pp. 2825–2830.

[26] F. Pérez Arribas. “Some methods to obtain the added resistance of a ship advancing in waves”. In: *Ocean Engineering* 34.7 (2007). ISSN: 00298018. DOI: 10.1016/j.oceaneng.2006.06.002.

[27] Joan P. Petersen, Ole Winther, and Daniel J. Jacobsen. “A machine-learning approach to predict main energy consumption under realistic operational conditions”. In: *Ship Technology Research* 59.1 (2012). ISSN: 09377255. DOI: 10.1179/str.2012.59.1.007.

[28] Derek A. Pisner and David M. Schnyer. “Support vector machine”. In: *Machine Learning: Methods and Applications to Brain Disorders*. 2019. DOI: 10.1016/B978-0-12-815739-8.00006-7.

[29] Johannes Schmidt-Hieber. “Nonparametric regression using deep neural networks with relu activation function”. In: *Annals of Statistics* 48.4 (2020). ISSN: 21688966. DOI: 10.1214/19-AOS1875.

[30] Alex J. Smola and Bernhard Schölkopf. *A tutorial on support vector regression*. 2004. DOI: 10.1023/B:STCO.0000035301.49549.88.

[31] Wieslaw Tarelko and Krzysztof Rudzki. *Applying artificial neural networks for modelling ship speed and fuel consumption*. 2020. DOI: 10.1007/s00521-020-05111-2.

[32] Tahsin Tezdogan et al. “Full-scale unsteady RANS CFD simulations of ship behaviour and performance in head seas due to slow steaming”. In: *Ocean Engineering* 97 (2015). ISSN: 00298018. DOI: 10.1016/j.oceaneng.2015.01.011.

[33] Tayfun Uyanık, Çağlar Karatuğ, and Yasin Arslanoğlu. “Machine learning approach to ship fuel consumption: A case of container vessel”. In: *Transportation Research Part D: Transport and Environment* 84 (2020). ISSN: 13619209. DOI: 10.1016/j.trd.2020.102389.

[34] Shengzheng Wang et al. “Predicting ship fuel consumption based on LASSO regression”. In: *Transportation Research Part D: Transport and Environment* 65 (2018). ISSN: 13619209. DOI: 10.1016/j.trd.2017.09.014.

[35] Wei Wang et al. “Experimental–numerical analysis of added resistance to container ships under presence of wind–wave loads”. In: *PLoS ONE* 14.8 (2019). ISSN: 19326203. DOI: 10.1371/journal.pone.0221453.

[36] Wenjian Wang and Zongben Xu. “A heuristic training for support vector regression”. In: *Neurocomputing* 61.1-4 (Oct. 2004), pp. 259–275. ISSN: 09252312. DOI: 10.1016/j.neucom.2003.11.012.

[37] Wenwei Xu. *What’s the difference between Linear Regression, Lasso, Ridge, and ElasticNet in sklearn?* Aug. 2019.

[38] Ran Yan, Shuaian Wang, and Yuquan Du. “Development of a two-stage ship fuel consumption prediction and reduction model for a dry bulk ship”. In: *Transportation Research Part E: Logistics and Transportation Review* 138 (2020). ISSN: 13665545. DOI: 10.1016/j.tre.2020.101930.

IV

Appendices

A

Filtration Plots

In this appendix different plots are presented of the same time interval in the data set. Each plot shows the result of an extra filtration which are discussed in chapter 6.3.

A.1. filtration plot after first filtration

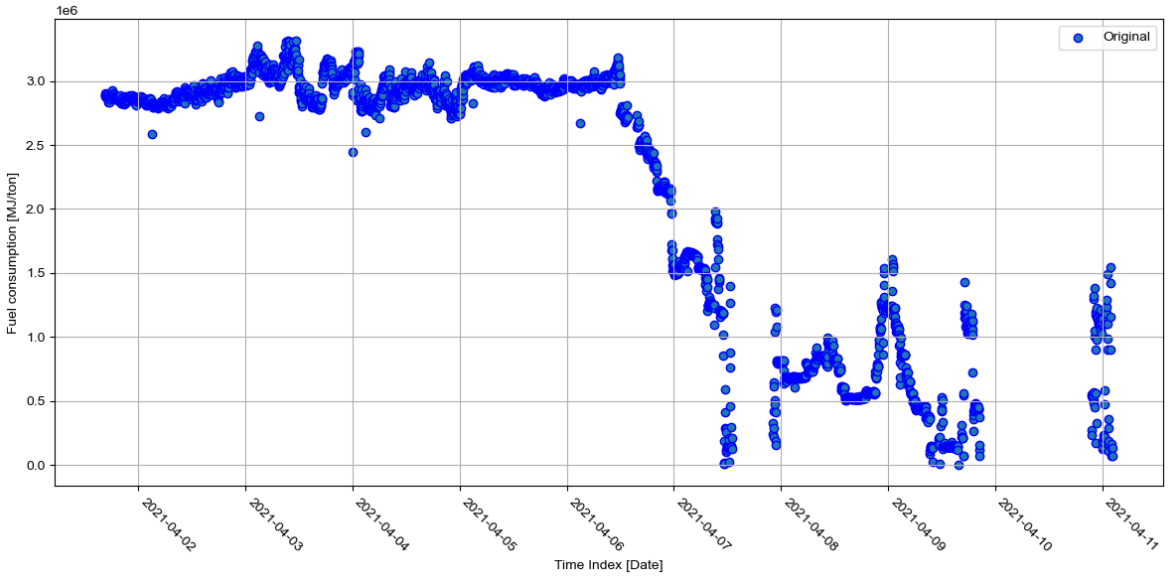

Figure A.1: Snapshot of data points after first filtration

A.2. filtration plot after filtration up to transverse velocity

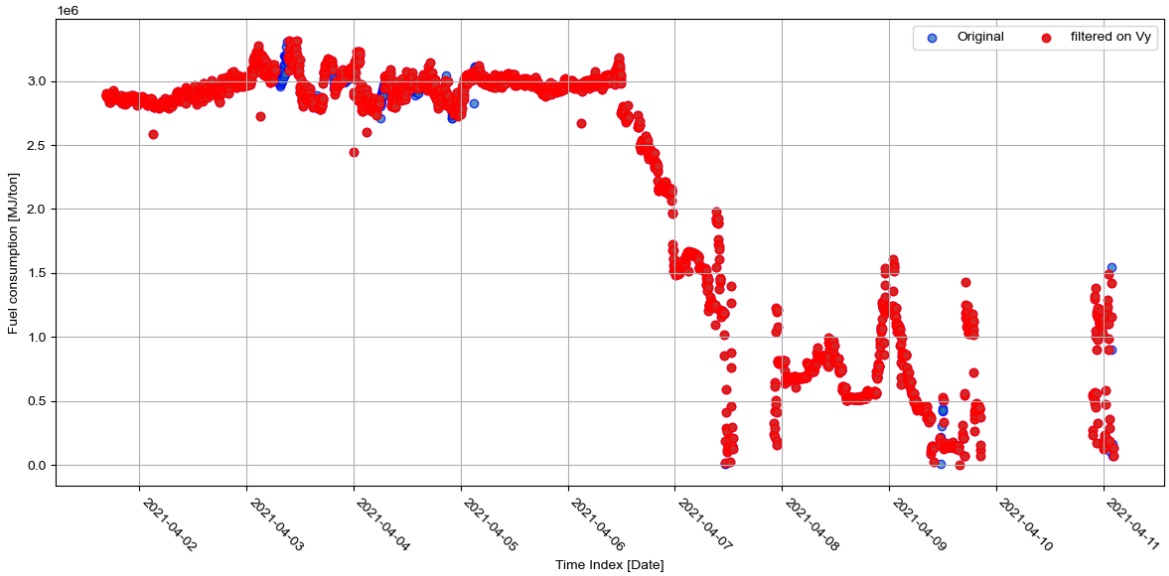

Figure A.2: Snapshot of data points after filtration up to transverse velocity

A.3. filtration plot after filtration up to longitudinal velocity

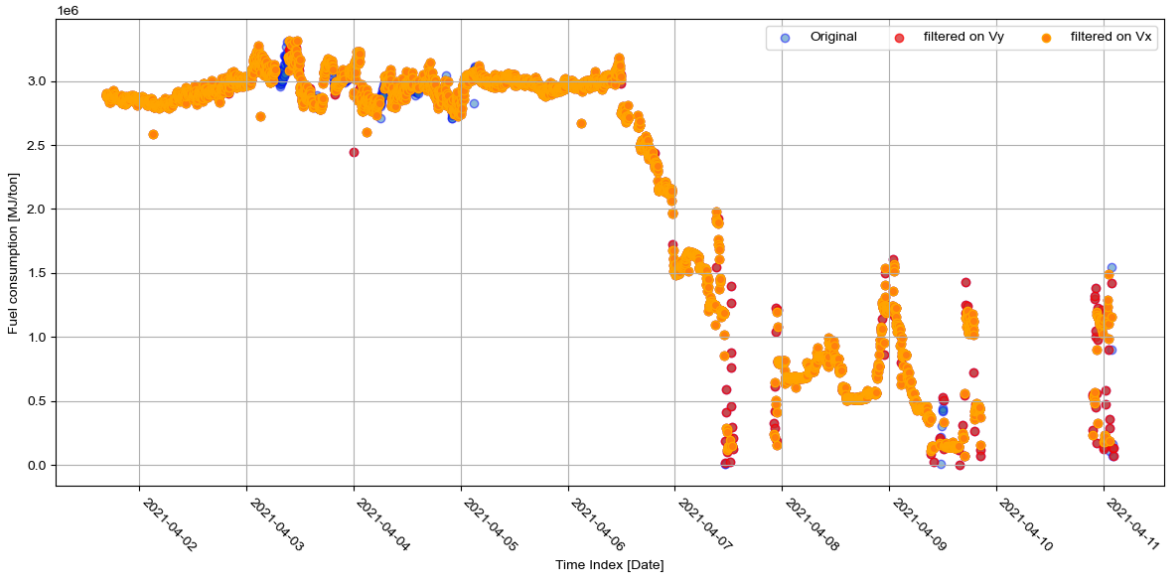

Figure A.3: Snapshot of data points after filtration up to longitudinal velocity

A.4. filtration plot after filtration up to rudder angle

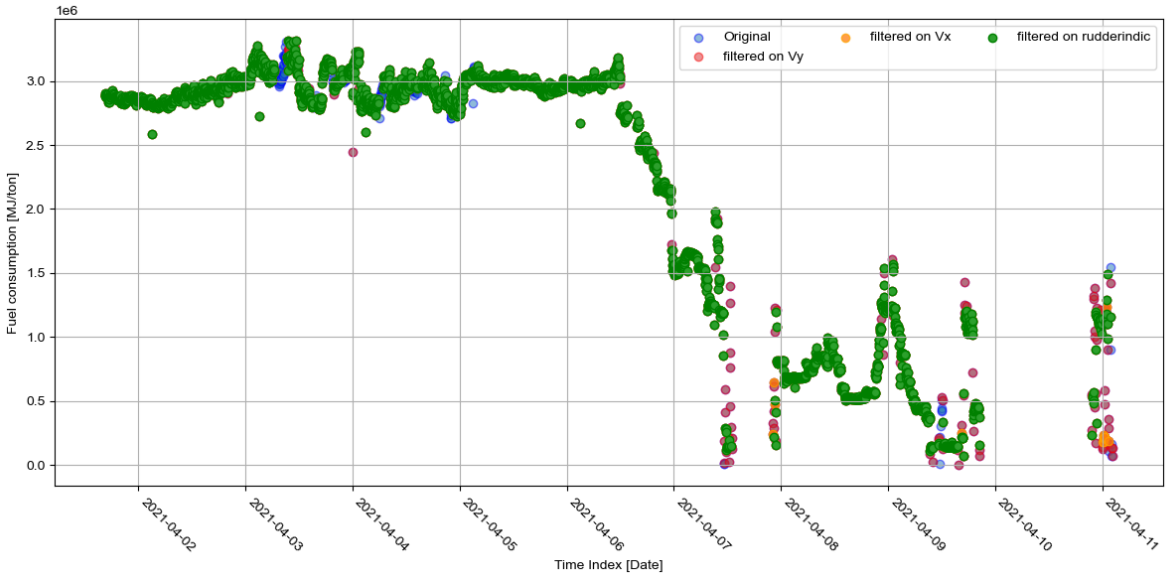

Figure A.4: Snapshot of data points after filtration up to rudder angle

A.5. filtration plot after filtration up to energy consumption

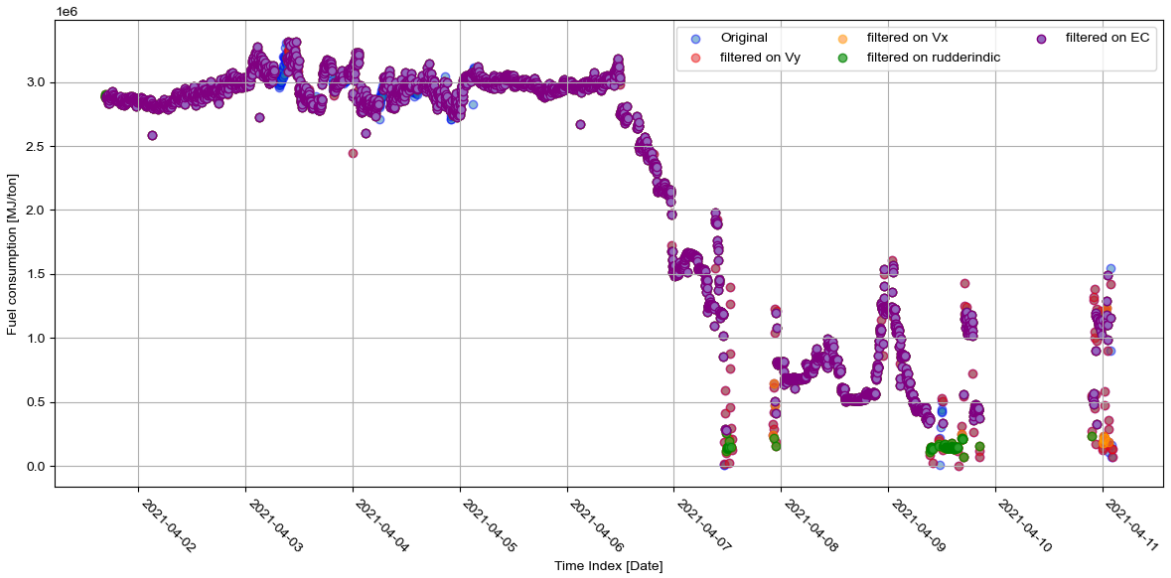

Figure A.5: Snapshot of data points after filtration up to energy consumption

B

Data Set Features

1. Mean rudder angle
2. Mean speed through water
3. Mean relative wind speed
4. Mean true wind speed
5. Mean real value of relative wind angle
6. Mean imaginary value of relative wind angle
7. Mean real value of true wind angle
8. Mean imaginary value of true wind angle
9. Mean real value of heading
10. Mean imaginary value of heading
11. Mean roll
12. Mean pitch
13. Mean angular velocity around longitudinal axis
14. Mean angular velocity around transverse axis
15. Mean angular velocity around vertical axis
16. Mean acceleration around longitudinal axis
17. Mean acceleration around transverse axis
18. Mean acceleration around vertical axis
19. Mean speed over ground longitudinal direction
20. Mean speed over ground transverse direction
21. Mean velocity in vertical direction
22. Mean velocity due to pitching
23. Mean velocity due to yaw
24. Real value of mean GPS-course
25. imaginary value of mean GPS-course
26. Standard deviation of rudder angle
27. Standard deviation of speed through water
28. Standard deviation of relative wind speed
29. Standard deviation of true wind speed
30. Standard deviation of real value of relative wind angle
31. Standard deviation of imaginary value of relative wind angle
32. Standard deviation of real value of true wind angle
33. Standard deviation of imaginary value of true wind angle
34. Standard deviation of real value of heading
35. Standard deviation of imaginary value of heading
36. Standard deviation of roll
37. Standard deviation of pitch
38. Standard deviation of angular velocity around longitudinal axis
39. Standard deviation of angular velocity around transverse axis
40. Standard deviation of angular velocity around vertical axis
41. Standard deviation of acceleration around longitudinal axis
42. Standard deviation of acceleration around transverse axis
43. Standard deviation of acceleration around vertical axis
44. Standard deviation of speed over ground longitudinal direction
45. Standard deviation of speed over ground transverse direction

46. Standard deviation of velocity in vertical direction
47. Standard deviation of velocity due to pitching
48. Standard deviation of velocity due to yaw
49. Standard deviation of real value of GPS-course
50. Standard deviation of imaginary value of GPS-course
51. Skew of rudder angle
52. Skew of speed through water
53. Skew of relative wind speed
54. Skew of true wind speed
55. Skew of real value of relative wind angle
56. Skew of imaginary value of relative wind angle
57. Skew of real value of true wind angle
58. Skew of imaginary value of true wind angle
59. Skew of real value of heading
60. Skew of imaginary value of heading
61. Skew of roll
62. Skew of pitch
63. Skew of angular velocity around longitudinal axis
64. Skew of angular velocity around transverse axis
65. Skew of angular velocity around vertical axis
66. Skew of acceleration around longitudinal axis
67. Skew of acceleration around transverse axis
68. Skew of acceleration around vertical axis
69. Skew of speed over ground longitudinal direction
70. Skew of speed over ground transverse direction
71. Skew of velocity in vertical direction
72. Skew of velocity due to pitching
73. Skew of velocity due to yaw
74. Skew of real value of GPS-course
75. Skew of imaginary value of GPS-course
76. Kurtosis of rudder angle
77. Kurtosis of speed through water
78. Kurtosis of relative wind speed
79. Kurtosis of true wind speed
80. Kurtosis of real value of relative wind angle
81. Kurtosis of imaginary value of relative wind angle
82. Kurtosis of real value of true wind angle
83. Kurtosis of imaginary value of true wind angle
84. Kurtosis of real value of heading
85. Kurtosis of imaginary value of heading
86. Kurtosis of roll
87. Kurtosis of pitch
88. Kurtosis of angular velocity around longitudinal axis
89. Kurtosis of angular velocity around transverse axis
90. Kurtosis of angular velocity around vertical axis
91. Kurtosis of acceleration around longitudinal axis
92. Kurtosis of acceleration around transverse axis
93. Kurtosis of acceleration around vertical axis
94. Kurtosis of speed over ground longitudinal direction
95. Kurtosis of speed over ground transverse direction
96. Kurtosis of velocity in vertical direction
97. Kurtosis of velocity due to pitching
98. Kurtosis of velocity due to yaw
99. Kurtosis of real value of GPS-course
100. Kurtosis of imaginary value of GPS-course
101. Mean derivative of rudder angle
102. Mean derivative of speed through water
103. Mean derivative of relative wind speed
104. Mean derivative of true wind speed
105. Mean derivative of real value of relative wind angle
106. Mean derivative of imaginary value of relative wind angle
107. Mean derivative of real value of true wind angle
108. Mean derivative of imaginary value of true wind angle
109. Mean derivative of real value of heading

110. Mean derivative of imaginary value of heading
111. Mean derivative of roll
112. Mean derivative of pitch
113. Mean derivative of angular velocity around longitudinal axis
114. Mean derivative of angular velocity around transverse axis
115. Mean derivative of angular velocity around vertical axis
116. Mean derivative of acceleration around longitudinal axis
117. Mean derivative of acceleration around transverse axis
118. Mean derivative of acceleration around vertical axis
119. Mean derivative of speed over ground longitudinal direction
120. Mean derivative of speed over ground transverse direction
121. Mean derivative of velocity in vertical direction
122. Mean derivative of velocity due to pitching
123. Mean derivative of velocity due to yaw
124. Mean derivative of of real value of GPS-course
125. Mean derivative of imaginary value of GPS-course
126. Standard deviation of derivative of rudder angle
127. Standard deviation of derivative of speed through water
128. Standard deviation of derivative of relative wind speed
129. Standard deviation of derivative of true wind speed
130. Standard deviation of derivative of real value of relative wind angle
131. Standard deviation of derivative of imaginary value of relative wind angle
132. Standard deviation of derivative of real value of true wind angle
133. Standard deviation of derivative of imaginary value of true wind angle
134. Standard deviation of derivative of real value of heading
135. Standard deviation of derivative of imaginary value of heading
136. standard deviation of derivative of roll
137. standard deviation of derivative of pitch
138. standard deviation of derivative of angular velocity around longitudinal axis
139. standard deviation of derivative of angular velocity around transverse axis
140. standard deviation of derivative of angular velocity around vertical axis
141. standard deviation of derivative of acceleration around longitudinal axis
142. standard deviation of derivative of acceleration around transverse axis
143. standard deviation of derivative of acceleration around vertical axis
144. standard deviation of derivative of speed over ground longitudinal direction
145. standard deviation of derivative of speed over ground transverse direction
146. standard deviation of derivative of velocity in vertical direction
147. standard deviation of derivative of velocity due to pitching
148. standard deviation of derivative of velocity due to yaw
149. Standard deviation of real value of GPS-course
150. Standard deviation of imaginary value of GPS-course
151. Skew of derivative of rudder angle
152. Skew of derivative of speed through water
153. Skew of derivative of relative wind speed
154. Skew of derivative of true wind speed
155. Skew of derivative of real value of relative wind angle
156. Skew of derivative of imaginary value of relative wind angle
157. Skew of derivative of real value of true wind angle
158. Skew of derivative of imaginary value of true wind angle
159. Skew of derivative of real value of heading
160. Skew of derivative of imaginary value of heading
161. Skew of derivative of roll
162. Skew of derivative of pitch

163. Skew of derivative of angular velocity around longitudinal axis
164. Skew of derivative of angular velocity around transverse axis
165. Skew of derivative of angular velocity around vertical axis
166. Skew of derivative of acceleration around longitudinal axis
167. Skew of derivative of acceleration around transverse axis
168. Skew of derivative of acceleration around vertical axis
169. Skew of derivative of speed over ground longitudinal direction
170. Skew of derivative of speed over ground transverse direction
171. Skew of derivative of velocity in vertical direction
172. Skew of derivative of velocity due to pitching
173. Skew of derivative of velocity due to yaw
174. Skew of derivative real value of GPS-course
175. Skew of derivative imaginary value of GPS-course
176. Kurtosis of derivative of rudder angle
177. Kurtosis of derivative of speed through water
178. Kurtosis of derivative of relative wind speed
179. Kurtosis of derivative of true wind speed
180. Kurtosis of derivative of real value of relative wind angle
181. Kurtosis of derivative of imaginary value of relative wind angle
182. Kurtosis of derivative of real value of true wind angle
183. Kurtosis of derivative of imaginary value of true wind angle
184. Kurtosis of derivative of real value of heading
185. Kurtosis of derivative of imaginary value of heading
186. Kurtosis of derivative of roll
187. Kurtosis of derivative of pitch
188. Kurtosis of derivative of angular velocity around longitudinal axis
189. Kurtosis of derivative of angular velocity around transverse axis
190. Kurtosis of derivative of angular velocity around vertical axis
191. Kurtosis of derivative of acceleration around longitudinal axis
192. Kurtosis of derivative of acceleration around transverse axis
193. Kurtosis of derivative of acceleration around vertical axis
194. Kurtosis of derivative of speed over ground longitudinal direction
195. Kurtosis of derivative of speed over ground transverse direction
196. Kurtosis of derivative of velocity in vertical direction
197. Kurtosis of derivative of velocity due to pitching
198. Kurtosis of derivative of velocity due to yaw
199. Kurtosis of derivative of real value of GPS-course
200. Kurtosis of derivative of imaginary value of GPS-course
201. 20 frequency bins of the spectrum describing the roll motion
202. 20 frequency bins of the spectrum describing the pitch motion
203. 20 frequency bins of the spectrum describing the acceleration around the longitudinal axis
204. 20 frequency bins of the spectrum describing the acceleration around the transverse axis
205. 20 frequency bins of the spectrum describing the acceleration around the vertical axis
206. 20 frequency bins of the spectrum describing the angular velocity around the longitudinal axis
207. 20 frequency bins of the spectrum describing the angular velocity around the transverse axis
208. 20 frequency bins of the spectrum describing the angular velocity around the vertical axis
209. Displacement
210. Distance between metacenter and center of gravity
211. Trim
212. Draft
213. Mean main engine energy consumption

C

Abbreviations used as labels in plots

- **EC_me_actual** : Energy Consumption
- **head_course_im** : Imaginary part of complex heading value
- **head_course_re** : Real part of complex heading value
- **kurt** : Kurtosis of non-derivative motion distribution
- **kurt_der** : Kurtosis of first derivative motion distribution
- **kurt_der2** : Kurtosis of second derivative motion distribution
- **kurt_der3** : Kurtosis of third derivative motion distribution
- **mean** : mean of non-derivative motion distribution
- **mean_accX** : mean of non-derivative distribution describing acceleration around longitudinal axis
- **mean_der** : mean of first derivative motion distribution
- **mean_der2** : mean of second derivative motion distribution
- **mean_der3** : mean of third derivative motion distribution
- **mean_gyrX** : mean of non-derivative distribution describing angular velocity around longitudinal axis
- **mean_pitch** : mean of non-derivative distribution describing pitch
- **mean_psi_im** : Mean of non-derivative distribution describing imaginary part of complex GPS-course heading value
- **mean_psi_re** : Mean of non-derivative distribution describing real part of complex GPS-course heading value
- **mean_roll** : mean of non-derivative distribution describing roll
- **rel_wind_im** : Imaginary part of complex relative wind direction value
- **rel_wind_re** : Real part of complex relative wind direction value
- **rel_wind_speed** : Relative wind speed
- **rudderindic** : Mean rudder angle
- **Skew** : Skewness of non-derivative motion distribution
- **Skew_der** : Skewness of first derivative motion distribution
- **Skew_der2** : Skewness of second derivative motion distribution
- **Skew_der3** : Skewness of third derivative motion distribution
- **slr_displacement** : Displacement of the vessel
- **slr_gm** : GM of vessel
- **Speed_tw** : Speed through water
- **std** : Standard deviation of non-derivative motion distribution
- **std_accX** : Standard deviation of non-derivative distribution describing acceleration around longitudinal axis

- **std_accY** : Standard deviation of non-derivative distribution describing acceleration around transverse axis
- **std_der** : Standard deviation of first derivative motion distribution
- **std_der_accX** : Standard deviation of first derivative distribution describing acceleration around longitudinal axis
- **std_der_accY** : Standard deviation of first derivative distribution describing acceleration around transverse axis
- **std_der_accZ** : Standard deviation of first derivative distribution describing acceleration around vertical axis
- **std_der_psi_re** : Standard deviation of first derivative distribution describing real part of complex GPS-course heading value
- **std_der_velocity_due_pitch** : Standard deviation of first derivative distribution describing motion-sensor velocity due to pitch
- **std_der_Vz** : Standard deviation of first derivative distribution describing vertical velocity
- **std_der2** : Standard deviation of second derivative motion distribution
- **std_der3** : Standard deviation of third derivative motion distribution
- **std_pitch** : Standard deviation of non-derivative distribution describing pitch
- **std_velocity_due_pitch** : Standard deviation of non-derivative distribution describing motion-sensor velocity due to pitch
- **TMD** : Draft of the vessel
- **Trim** : Trim of the vessel
- **true_wind_im** : Imaginary part of complex true wind direction value
- **true_wind_speed** : True wind speed
- **Vx_head_course** : Longitudinal speed
- **Vy_head_course** : Transverse speed