# Predicting offshore workability for platform supply vessels using IoT and machine learning

R.F. Dick
*SNSPOOL Fleet manager*
*Peterson Energy Logistics*
Den Helder, The Netherlands

Maritime Institute Willem Barentsz, University of Applied Science, Terschelling

***Abstract*—The North Sea offshore oil and gas production platforms need supply vessels regularly serving them for replenishment of materials (food), bulk like potable water and fuel. The weather and subsequently the metocean circumstances can create large disruptions in the sailing schedule of Platform Supply Vessels (PSV). These schedule anomalies result in additional operational costs due to idle time from waiting at the offshore location. Also, it may affect the safety and well-being of the crew being offshore in adverse weather waiting for a time window. The use of vessel motion prediction as a key indicator in PSV schedule planning has not been subject to many previous studies. One of the reasons for not using the vessel motion limits as an indicator is the costs of modelling various vessel types, and a large number of offshore locations and platform design variations. Traditionally, the predicted vessel motion is calculated by the use of the vessel hydrodynamic particulars namely the Response Amplitude Operator (RAO), which translates the wave excitation force into a vessel motion response. Within the Peterson SNSPOOL offshore supply fleet, vessels are equipped with many IoT sensors used for monitoring vessel operations, fuel optimization and electronic logging (voyage reports). This sensor technology can also be used to record motion and navigation parameters like roll, pitch, heave, heading, wind speed, and rate of turn. By using the operational parameters and combining these with the locally measured metocean data, the objective was to predict the motion and create a workability index for the vessel planners by use of Machine Learning (ML) models. Through a selection of available ML models and the comparison of Support Vector Regressors (SVR), Neural Networks (NN) and Gradient Boosting Tree (GBT) models a roll, pitch and heave prediction was achieved with 88%-91% accuracy, which could be validated against in-situ measurements and hydrodynamic models.**
**Keywords: Offshore Supply, Vessel Motion, Machine Learning, Crew Safety**

## I. Introduction

The SNSPOOL is a collaborative agreement between nine Oil and Gas operators who primarily operate on the Dutch Continental Shelf, with some minor activities in the UK, Denmark and Germany(fig:1). The SNSPOOL started in 2002 as a vessel-sharing concept. Later, it developed towards a total 4PL concept in which road and integrated logistics services were added. The SNSPOOL was initially set up to maximise efficiency and reduce operational costs by combining volumes and vessel capacities. The SNSPOOL collaboration entails the supply chain and logistics services between onshore and offshore facilities. Activities mainly consist of warehousing,


Fig. 1: Collaboration in the SNSPOOL (Peterson)

consolidation, quay activities, shipping activities and common administrative tasks such as cost allocation and invoicing, customs clearances and voyage administration. The shared activities in the SNSPOOL are facilitated by Peterson Den Helder B.V. (the facilitator), who manages the major cost components like vessel chartering, tank cleaning, fuel usage, port activities and optimizing idle time based on a participation level amongst the operators. The vessel activities, loading and discharging, and short-term storage occur on their quayside in Den Helder (The Netherlands). Having one primary base assures minimal delays and competitive port fee rates. Due to the sharing of vessel and storage capacities, the scaling leads to competitive pricing, which manifests in the purchase of fuel and vessel chartering. As a result, higher volumes and sharing costs lead to an average reduction of 30% for the operators while the availability and service levels are maintained higher than in a single-operator supply chain.

Like in any vessel operation, the costs for operating vessels can be split into two parts, the productive costs and the nonproductive costs (NPT). This paper focuses on reducing NPT. NPT can be further subdivided into the following components: waiting times (waiting in port, waiting on the weather (port and offshore), waiting on departure to fit the schedule, waiting on day-shift offshore due to platform closing times, waiting on handling offshore) and vessel breakdowns or scheduled maintenance. The total NPT for operating offshore supply vessels is about 30% of the total costs. When considering the Waiting On Weather Offshore (WOWO) component, the costs for waiting on weather are made up of the following components: additional fuel costs, scheduling costs from extra sailings, scheduling costs as additional chartering costs, port fees, labour costs in port (lines men & stevedores ), pilotage costs.

*WOWO consequences*

Sending an offshore supply vessel to an offshore location under adverse weather conditions can lead to several unwanted situations. When a vessel arrives at the platform safety zone (500m), the vessel needs to wait for the next suitable weather window. This results in additional fuel being used (hence extra emissions), voyage scheduling time becoming uncertain, the opportunity for maintenance time being missed and the crew being exposed to severe vessel motions (roll, pitch and heave) which will cause levels of fatigue.

Crew fatigue combined with platform personnel waiting for their platform replenishment may lead to lowering safety barriers when considering lifting operations alongside the platform, according to [1]–[3], exposure to high levels of vessel motion has a negative effect on the crew performance, hence minimizing crew to adverse weather conditions legitimates a study for better vessel motion prediction information to the vessel planners.

*Research objective*

This paper focuses on the logistic needs of the offshore wind and O&G industry on a daily operations planning time scale. The prevention of offshore waiting due to weather restrictions in these short time windows is a perspective that is little discussed in the existing literature. Within the vessel planning, the weather forecast information and the resulting vessel motion information are currently not presented as one single source of information to the personnel involved. During the analysis of the causes leading to weather waiting situations, it was observed that users will benefit from having the weather information and the resulting vessel motion combined as one. This paper focuses on the perspective of a logistic service provider serving a multitude of offshore locations, as opposed to most existing logistic models that often consider a single location. Additionally, the research provides a method to model vessel motion with the use of machine learning models with parameters from sensors located on the vessels.

## WOWO ANALYSIS

Analysing the vessel data from 2015 until 2019 (included), the WOWO incidents and distribution show a strong seasonal relation (fig:2). During the summer months, the percentage of incidents whereby WOWO occurred resolves around 2,5%, while in the winter months, the number of WOWO incidents increases to 14% of all voyages. If we consider the percentage of the total used charter time, the WOWO shows a similar pattern whereby in charter time, the summer months show a percentage of around 1%, while this increases to 6.5% at the winter times. These percentages are calculated as part of the total monthly charter time. However, if we estimate this percentage referencing the planned charter time (5 production vessels = 3600 charter hours monthly), then this percentage increases even further and must be multiplied with an average correction factor of 2.3.

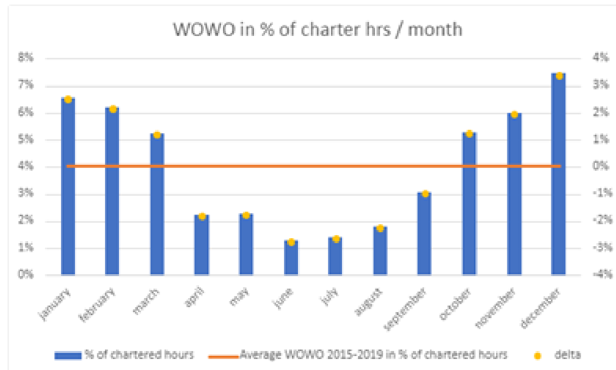



Fig. 2: Annual Number of WOWO occurrences

Considering the absolute hours (in charter days) and calculating the mean annual WOWO in five years, the average monthly loss varies from just above five days in the summer months to around 25 days in the winter months. The loss of charter days translates into a loss of €35000 to €175000 per month [4].

*WOWO distribution*

The SNSPOOL operators forecast their demand from the SNSPOOL based on the production runs and projects like accommodation support, decommissioning, maintenance etcetera. We use a statistical approach in the demand model according to the research conducted in the SNSPOOL covered in the research paper by Louise Peet. [5] The demands in the pool program are net values. To predict the gross values ( including the idle times), we need to calculate the likelihood of WOWO occurrence. This can be done with the use of data from the SNSPOOL using SPSS27 statistical analysis software. The WOWO frequency of both incidents and duration shows a non-normal distribution. The data is the best fit with a Gamma distribution with scale factor $\theta = 0.9$ and shape factor $k = 0.65$ (fig:3).

By using the Gamma distribution to obtain the probability of occurrence P(x), the likelihood of WOWO P(0.5 days/voyage) is 0.56. Thus, 60% (Cumulative Density Function, CDF) of all voyages show a WOWO of less than 0.5 days, while 20% show a WOWO of 1 day or more. Using the Gamma fit, we are able to add a portion of idle time to the net demand in the pool program. For the prediction and chartering of vessels, an 80% confidence level is achieved by adding 1 day to a production run. This is called flex time offshore[1].

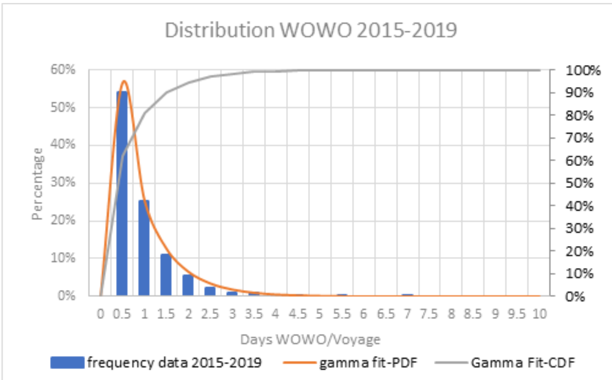



Fig. 3: WOWO Gamma Distribution

## II. Literature Review

During the literature review, a desktop study on existing research and novel solutions regarding vessel motion prediction solutions will be presented. The methods which are currently mostly used will be briefly explored. The emphasis on the use of data with machine learning models will be discussed with Neural Networks, Support Vector Regression and Gradient Boosting Tree models.

Planning vessel operations on the North Sea is equivalent to calculating the probabilities of successful operations. Three main procedures can be distinguished namely using (A) the vessel hydrodynamic response by using the vessel model and wave characteristics, (B) using a safety factor on the available time windows performing the marine operations and (C) calculating the vessel motion response using a (non-linear) regression technique using a statistical model.

### A. *hydrodynamic response*

In predicting the vessel's workability, one of the methods is to calculate the hydrodynamic behaviour based on the vessel's hull shape. Hydrodynamic behaviour is based on translation factors between (energy density) wave spectrum characteristics and vessel motions. These so-called Response Amplitude Operators (RAO's) translate the vessel's amplitude response and phase response with the wave energy spectrum to a vessel motion spectrum (fig:4).

The vessel dynamic response is described by the vessel motion equation:

$$(M + A)\ddot{\theta} + B\dot{\theta} + C\theta = F_{\mathrm{w}} \tag{1}$$

Where $\mathbf{M}$ and $\mathbf{A}$ are the structural and the added mass matrices respectively, $\mathbf{B}$ is the damping matrix, $\theta$ is the position vector, $\mathbf{C}$ is the restoring matrix and $\mathbf{F_W}$ is the total external (wave) force vector consisting of the Froude-Krylov and diffraction force, including first- and second-order wave forces. The mass, damping and force vectors are frequency ($\omega$) and direction ($\theta$) dependent.

[1]An extra day is added in the offshore schedule between sailings to create robustness

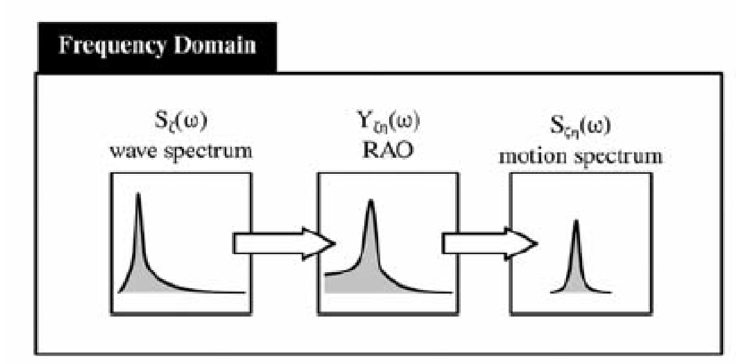



Fig. 4: RAO and wave spectrum

Nowadays, the vessel response forecast is mainly carried out using RAOs combined with the so-called wave spectrum (S). For the North Sea area, this spectrum is best described with a so-called JONSWAP (Joint Offshore North Sea Wave Analysis Program [6]), which in turn again is based on the Brettschneider spectrum for deep-water waves. [7]–[9]

When assuming linear behaviour, the vessel response operator $H_j(\omega, \theta)$ describes the translation between the vessel response output $S_{\mathrm{j}}(\omega, \theta)$ and the wave excitation input $\zeta_{\mathrm{j}}(\omega, \theta)$ depending on the circular wave frequency $\omega$ and the heading $\beta$ and the vessel speed.

$$\boldsymbol{H}_{\mathrm{j}}(\omega; \beta) = \frac{S_{\mathrm{j}}(\omega; \beta)}{\zeta(\omega)} \tag{2}$$

If the vessel hull shape is known, mass distribution and wave heading interaction (vessel speed, which in this case is zero being stationary alongside a platform), the transfer function of hull motion characterised by the RAO and phase angle can be determined. At the Norwegian institute Sintef, a so-called *Operational Robustness Index (ORI)* was developed by using the vessel's response characteristics with a 2D-wave spectrum and the user set criteria for the safe operation of the vessel [10]. Sintef created a web-based tool [11] to calculate standard-sized offshore vessels based on the JONSWAP spectrum [6] for various conditions. The simulation outcome shows the percentage of workability and ORI for a specific season, either on the North Sea or North Atlantic (using the Pierson-Moskowitch wave spectrum).

### B. *Alpha Factoring*

For marine operations which may be planned over a prolonged time, DNVGL introduced a method in which the uncertainties are factored using a so-called $\alpha$ factor to the operational window in which the marine operations are taking place. [12]. Gutmestad discussed in his paper *on waiting on weather windows* [13], the use of the DNVGL procedure on temporary marine operations by using the factoring out

of the reference time $T_{\text{ref}}$ based on the operational period $T_{\text{pop}}$ and the significant wave height $H_{\text{s}}$ from the weather forecast. Sarah Wilcken concluded in her thesis [14] that the method of using the $\alpha$ factoring in marine operations shows a considerable spread in deviations due to uncertainties in weather forecasts and by only taking the significant wave height into consideration, this would be forming a too small basis for planning marine operations with elevated weather uncertainties.
To overcome the lacking of the wave period uncertainties $T_{\text{p}}$ in the *alfa* factoring, Wu & Gao (2021) [15] proposed a new factor called the *revised alfa* factor $\alpha_r$. For the revised alpha factor the uncertainties of the weather forecast parameters like wave period but also other weather variables like wave direction $\text{D}_{\text{u}}$ or wind speed $\text{U}_{10}$ might be taken into the analysis of the alpha factor. Using the uncertainties of multiple weather variables with the dynamic response analysis of the vessel [15] showed that taking the wave spectrum peak period ($\text{T}_{\text{p}}$) into consideration in the alpha factor for marine planning, the unsafe marine lifting periods increased considerable, hence avoiding unsafe situations.

### C. Statistical modelling

Models may be constructed using machine learning methods using vessel motion measurements with metocean observation data. The strength of machine learning is that the algorithm does not need to be defined to calculate the results. Rather the opposite, the measured results are being fed to a model with the expected result. The machine learning model is then able to create the model from the learning data and verify this with the test data to present a model able to predict vessel motion based on historical data rather than having to program vessel motion algorithms and matrices. In this innovation, machine learning may dominate the final solution; hence, more understanding is needed to select the proper Machine Learning (ML) method. For motion forecasting, various regression methods can be used. Big data from vessel observations, sensors and high-precision observation measurements are combined with features from weather data and responses. This data can be used to model the particular ship's movement for that specific position and vessel heading, draft, loading conditions, etcetera. The fundamental of a linear regression model, which can be used in machine learning is represented by equation 3. The $\beta_0$ is the intercept and $\beta_1....\beta_n$ the slope coefficients of the different data features $x_i...x_n$

$$\phi = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ....\beta_n x_{in} + \epsilon \tag{3}$$

The intercept and slope coefficients are calculated by minimizing the residual sum of the loss function, which can be a squared loss (equation 4) or any other loss function which is continuous, converging and differentiable like Huber-Loss function [16]. In order to obtain both the slope and intercept minimums, the minimum loss is obtained by applying a gradient descent using a Taylor expansion (equation 5) towards the global minimum on both the slope and the intercept functions in equation 6.

$$g(\beta) = \underset{\beta}{argmin} \sum_{i=1}^{N} (f(x_i) - \Phi_i)^2 \tag{4}$$

With
$g(\beta)$ = the residual sum of the loss function
$f(x_i)$ = the value of the function of x
$\Phi_i$ = the mean value
Taylor approximation for minimizing the loss function l and obtaining the step size by using the $2^{nd}$derivative of the loss functions.

$$\begin{gathered} l(\phi + s) \approx l(\phi) + g(\phi)^T s \\ l(\phi + s) \approx l(\phi) + g(\phi)^T s + \frac{1}{2} s^T H(\phi) s \end{gathered} \tag{5}$$

With
$g(\phi)$ = $\bigtriangledown l(\phi)$ (is the gradient of l)
$H(\phi)$ = $\bigtriangledown^2 l(\phi)$ (is the $2^{nd}$ derivative Hessian Matrix function of l).

By using the step size from the step size found by the Taylor approximation, the coefficients are updated using gradient descent and the stepsize:

$$\beta^{i+1} = \beta^i \pm s \bigtriangledown g(\beta^{(i)}) \tag{6}$$

With
s = the step size obtained by the Taylor expansion,
or a fixed (small) number between 0 and 1.
$\bigtriangledown_g$ = the gradient of the loss function which approaches zero.

The basic data model structure is described by [17] and is shown in figure 5.
The use of machine learning (ML) in this perspective is not new. It has already been successfully applied in identifying and predicting financial markets, taxation anomalies, credit card credibility, and identifying food (wine, oils). The majority of the techniques used for building ML models are Kalman Filtering (which is not purely a machine learning model but a stochastic filtering algorithm), Neural Networks (NN), Support Vector Machines (SVM) ( in a vast set of varieties), Fuzzy Logic (FZ) to predict elevators positions in large buildings, Random Forest Decision Trees (RFDT), with boosting variants (BT) and Multiple Layer Perceptron as a feed-forward neural network. Li et al (2017) [18] identified the various types of machine learning with advantages and disadvantages which are summarized in table I.

#### *Support Vector Machines*

The Support Vector Regressor(SVR) [19], is one of the most used mathematical approaches for solving non-linear statistical regression problems. While SVR may be very effective due to low computational loads and the use of relatively few samples with regard to the number of features,

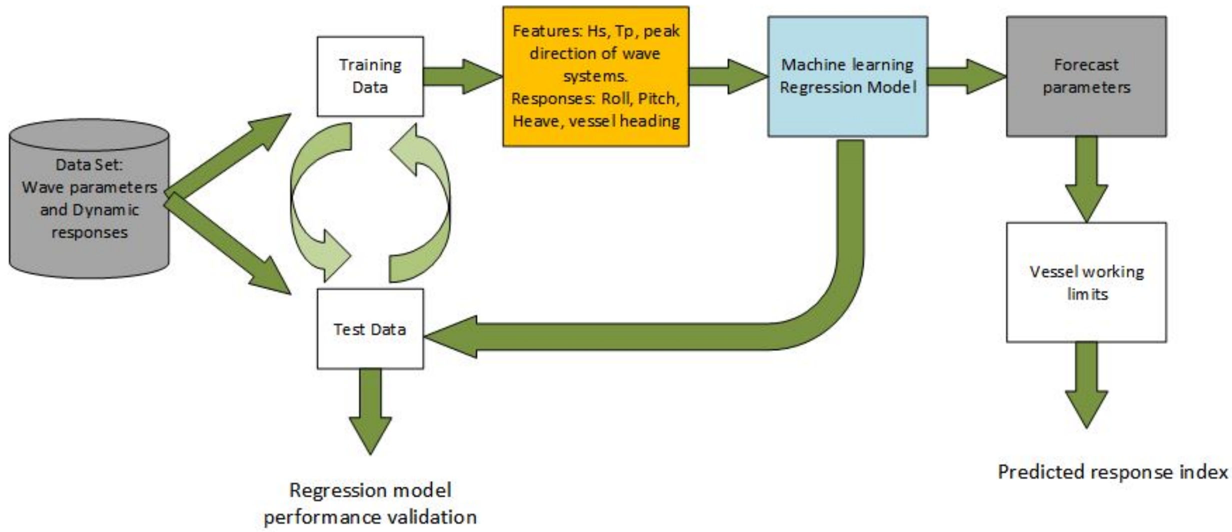



Fig. 5: Basic structure of using data in machine learning, adapted from [17]

it also has some disadvantages: easy to over-fit (the model is too strongly tuned and lacks generalisation) and probability scores and benchmarks are not directly calculated (so cross-validation is needed). Support Vector Machine models have been subject to many many studies concerning vessel motion predictions in the past. For example, non-linear motion predictions on moored FPSOs (Floating Production and Storage Object) were subject to a study by Xu & Zou [20], [21]. The SVR was chosen as the subject of the study over Artificial Neural Networks (ANN) and Multi-Layer Perceptron (MLP, a type of Neural Network) or Radial Base Feature (RBF) network because SVR can outperform the other proposed machine learning algorithms on contaminated data (especially outliers). Since measured data from real-time data is likely to be contaminated, the research focused on using Support Vector Regression with a (Gaussian) radial base function as the kernelized function (due to the non-linearity of the vessel damping characteristics). The SVR model tuning was performed using a training algorithm Sequential Minimal Optimization (SMO) [22]. The model parameters were set using Heuristics (trial and error method) for the level of generalisation penalty and kernel parameters. The verification and validation proposed by the researchers used Root Means Square Error (RMSE) and Square Correlation Coefficient ($R^2$) compared with the method results by using the motion equation (equation 1) of the vessel. The model prediction results are considered valuable if the RMSE values are near zero and the $R^2$ is near 1. The research showed vessel roll and pitch prediction results around RMSE of 0.1 and $R^2$ near 1. This verifies that the model is good enough for implementation. The results, however, were valid only for very short time windows (175 seconds) and rather not for the prediction horizons which we are aiming for (1-5 days ahead).

*Neural Networks*

Within offshore engineering, wave prediction using neural networks is a specific research field. Workability analysis for offshore applications has been the subject of studies using artificial neural networks. During the research with Heerema Marine Contractors, Haenen (2012) [23] studied the feasibility of using an ANN to calculate the vessel's workability and motions. From this study, it was proven that neural networks can replace conventional methods using hydrodynamic calculations with 2D wave spectra by using hind-cast data. In his Thesis, Haenen discriminates between frequency domain and time-domain approaches, using the frequency domain to predict long-term prediction, while the time-domain approach focuses on short-term prediction. The latter method uses the ship motion data as hindcast from Motion Reference Units (MRU). [23] Haenen uses two networks (time and frequency domain) to assess the validity of the model outcome versus RAO alternatives by providing wave inputs to predict vessel motions. The second model uses the hind-cast vessel motions to predict future vessel motions to verify if neural networks can establish correlations. The conclusion of his research, especially interesting using the parametric inputs wave height, period and direction ($H_s/T_p/\theta$), shows that the neural network is able to outperform the RAO methods (both for transit conditions and construction conditions) when considering the Mean Squared Error (MSE) as a quality benchmark. Furthermore, in the time domain, both the Pearson correlation and the Loss function show better results for the neural network than (wave) diffraction methods [7], the conclusion can be made that ANN may have better predictive capabilities than the conventional methods. The subject of prediction vessel motion RAOs using Neural Networks was also the topic of the thesis by Kaja Steffensen Bremer (2018) [24] This study used scaled models in a Sintef(Norway) laboratory. As a result of having the set-up in a laboratory, she was able to set all parameters like sampling frequencies and wave characteristics to an optimal

TABLE I: Various regression models according to [18])

| Prediction method | Advantages | Disadvantages |
|---|---|---|
| LR (linear regression) | Simple, easy to use and interpretable<br>Works well with small data sets<br>Assume linear approximation | No model generalisation is possible<br>Not suitable for complex correlation and non-linearity |
| SVM<br>(support vector machine) | Less over-fitting<br>No local minimum<br>Good in generalisation (in line with overfitting)<br>Good with Non-linearity | Expensive computation<br>Results are complicated and lack transparency<br>Selection of Kernel function |
| DT (Decision Tree) | Simple to understand, not a complex algorithm<br>Fast Construction<br>White box iso Black Box<br>Works well with non-homogeneous data<br>Fast prediction | Not suitable for online learning<br>computational complexity for uncertainty<br>Unable to extrapolate |
| KF (Kalman Filter) | Computational efficient | It does not work in considerable non-linearities<br>Works only with Gauss (white) process noise |
| MPC<br>(multiple party computation) | Systematic design<br>Explicit use of a model.<br>Stability guarantee | Limited model choices.<br>Large computation for non-linear and uncertain systems<br>. |
| FL (fuzzy logic) | Flexible, intuitive knowledge base design<br>A natural way of expressing uncertainty | Nontrivial and time-consuming to obtain rules.<br>Difficult for performance-robustness trade-off |
| NN (neural Networks) | Strong in generalisation ability.<br>Suitable for problems that are difficult-<br>to specify mathematically.<br>Efficient for online learning | Limited ability to explicitly -<br>identify possible causal relationships<br>Prone to overfitting<br>NN does not handle local minima very well |
| GBT[a]<br>(Gradient Boosting Tree) | Grey box iso black box.<br>Very fast computing<br>No data scaling necessary.<br>Handles weak learners very well<br>Very efficient with small data sets | Training error may not decrease exponentially<br>It can be sensitive to outliers<br>Prone to over-fitting<br>Model tuning may be slow<br>Model complexity and interpretability |

[a] adapted to Li et al

setting. Her research concluded that although the coefficient of determination ($R^2$) using neural networks resulted in .98, the $R^2$ is somewhat misleading, and the Root Mean Squared Error (RMSE) or Mean Absolute Percentage Error (MAPE) are a better indicator of the model's performance. Overall she concluded that machine learning models like neural networks could predict vessel motion within an error rate of below 10% and The RAO prediction within an error rate below 3%.

*Gradient Boosting Tree*

The Gradient Boosting Tree (GBT) model combines a decision tree model with a boosting algorithm (basically creating a strong feature by using a pool of weak (or base learners) and applying voting) with a gradient descent loss function [16]. One of the strong arguments for a GBT is that this model is more accessible to comprehend by being a "grey" box instead of many other ML models being black boxes. Not having to extensively engineer data before using them in a GBT model is another advantage of using a GBT (no scaling and normalisation of data are necessary before using the features in a GBT). A disadvantage is that a GBT cannot extrapolate, and the tuning of the model parameters (random grid search)may take considerable time.
To increase performance and accuracy on small data sets, [25] introduced a stochastic gradient boost, in which the use of small sub-samples was introduced. This type of hybrid bagging-boosting showed that even with an increase in variance on the sub-samples, the overall variance of the model decreased while the accuracy increased (fig:6). The result on small data sets suggests that the improvement in the variance is an important ingredient.
In 2022 Guachamin-Artero [17] and Portilla [26] published a

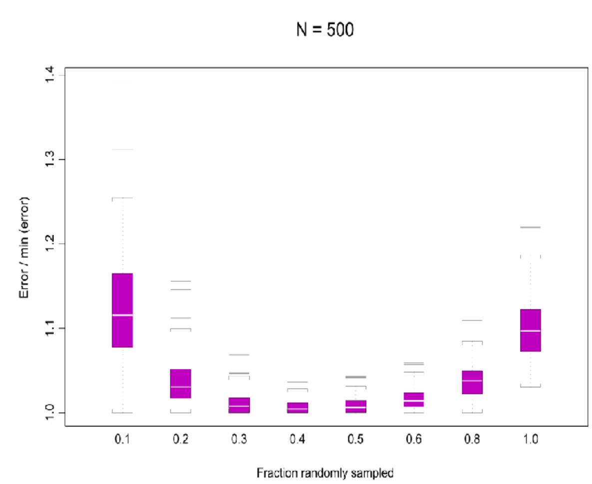



Fig. 6: use of introducing small sub-samples on GBT accuracy [25]

paper describing the prediction of dynamic vessel responses. The research aimed to answer whether it is possible to predict and plan marine operations using machine learning with the use of wave density spectra. The researchers used hind-cast data with wave spectra partitioning algorithms based on watershed methods [26] to obtain the different wave characteristics and used these wave systems with a GBT tuned with different features. Finally, a comparison was made using linear regression and GBT models. The research showed that a boosted tree model could accurately predict vessel roll motion using sufficient features for all wave systems in a wave spectrum. The research performed by [17] & [26] aimed to improve marine operations in the planning stage, hence this setup could be applicable to the method which will be applied in this research.
Abbas, [27] compared machine learning models in his thesis at the University of Liege (Belgium) to improve vessel performance analysis prediction using machine learning models. Although the research aim differs from this innovation theme, the outcome of the model comparison still shows valid arguments for the selection of models in a later stage for this innovation. The top three rankings all consisted of some form of a Tree model [27],pg51.

## III. Modelling Method

In this section, the method will be discussed based on figure 5. The data engineering part will be explained based on the data-driven design structure described in the Delft Design Guide [28]. Feature filtering, cleaning and data engineering will be explained before they can be used in a machine-learning model. The regression model selection will be further explored using different loss functions, tuning methods/grid search techniques and model generalization with an emphasis on the bias-variance trade-off. Finally, the forecast data is obtained using API requests from the forecaster[2] and used to predict the regression outcome for the three degrees of freedom (DOF), Roll, Pitch and Heave.

The SNSPOOL platform supply vessels, used for the support of the production platforms, are all equipped with sensor technology linked with a cloud server. This data link was used to interface the sensors needed for the experiment. The vessel used in the experiment is the *Dina Scout*, a vessel managed by the Norwegian owner Myklebusthaug [29]. The vessel details are summarized in table II vessel particulars. The vessel design is a UT755-LN designed by Ulstein[3](NO) and is typical for the platform supply vessels in the SNSPOOL. From the vessel's line plan and the stability booklet the hydrodynamic model was created and RAOs calculated[4]. This hydrodynamic model will be used for the validation of the experiment. The sensors which were used for the experiment are the vessel's own sensors used for the dynamic positioning and navigation of the vessel. Besides the vessel's own sensors, an additional inclinometer was installed in the midships on the bridge. Due to the commercial consequences of using $3^{rd}$ party motion reference units (MRU), it was decided to install a dedicated inclinometer from the vendor IRM. The inclinometer details are listed in table III.
The installed inclinometer measures 2 axes (roll & pitch). Also, the heave of the vessel needs to be predicted as part of the motion forecast, therefore, the vessel heave data will be measured using the (D)GPS antenna height, normalized for the tide differences and the position of the antenna on the vessel. The method used to calculate the movement of the GPS antenna to a vessel heave is taken from the method in [30].

TABLE II: Vessel particulars Dina Scout.

| Parameters | Descriptions | Values |
|---|---|---|
| L | Length overall | 76.60 m |
| $L_{pp}$ | Length between perpendiculars | 68.20 m |
| B | Breadth | 16.00 m |
| D | Depth | 7.0 m |
| T | Draft | 5.47 m (mean) |
| $\triangle$ | Displacement | 5077T |
| $\bar{GM}$ | Tranverse Metacentric Height | 1.66 m |
| LCB | Longitudinal Center of Buoyancy | 32.87 m from AP |
| LCF | Longitudinal Center of Flotation | 29.87 m from AP |
| | Offshore operational limits | values |
| roll Limit | | 10° |
| pitch limit | | 10° |
| heave limit | | 2m |

TABLE III: Inclinometer JN2201.

| Description | Value |
|---|---|
| Nr of axis | 2 (Roll, Pitch) |
| Maximum Angle | 45[°] |
| Sampling frequency range | 0.5..10 [Hz] |
| Accuracy | $\leq$ 0.001[°] |
| Resolution | 0.01[°] |

### *Data design*

The data structure is designed according to the data design guides and data-driven modelling [31], [32]. This process is briefly as follows (fig:7). Acquisition: Data collecting, the collecting of data depends on the platforms on which it is residing and the possibility to use (REST)API[5] to request the data. Wrangling: Data cleaning, the data needs to be filtered and cleaned or removed from the noise, trends etcetera. Exploration and Analysis: merged and synchronized to timestamps and sampling rate between the tables and finally. Reporting: features must be selected from the data that are to be used in the machine learning model.

[2]with Peterson energy Logistics the forecasts are being supplied by Infoplaza Business BV, https://www.infoplaza.com/nl/marine/marine-weather
[3]https://ulstein.com/
[4]The hydrodynamic model and RAOs were calculated by the company MO4

[5]representational state transfer

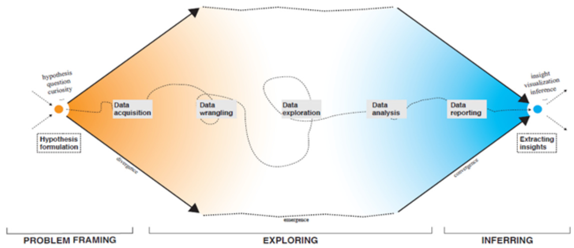



Fig. 7: data design structure, [31]

*Data acquisition:* The vessel sensor data used in the motion prediction model are:

- Vessel heading in degrees from Gyro North [°]
- Vessel Speed [kts]
- Vessel observed wind speed relative to vessel speed [kts]
- Vessel observed wind direction relative to heading [°]
- Vessel heave [m]
- Vessel pitch [°]
- Vessel roll [°]

Vessel sensor data is recorded and stored from the vessel sensors using an (IoT) cloud system from Onboard [33]. The data residing in the cloud servers can be accessed using API with embedded graphql queries [34][6]. The upload of data from the vessel site is achieved by using either the vessel satellite link (VSAT) or, preferably, a 4g link using a dedicated installed 4g connection with the use of a dedicated (peplink[7]) router.

Metocean observations in the North Sea are supplied (offline) by the Dutch Ministry of Infrastructure by means of Rijkswaterstaat [35]. The Dutch North Sea offshore platforms are equipped with meteorologic and oceanic observation sensors. Because the SNSPOOL vessel used in this experiment services many offshore locations, a selection of the following locations were used as data source for the model: Hollandse Kust Zuid (HKZ-A), platform L9-FF, platform K14-A and K13, platform J6-A, platform A12. The offshore observation data was also supplied using a (REST)API (online) from MeteoServer [36] with a time stamp of 10 minutes from the same offshore locations.

The data features from the observations, which are used to train the model, are the wave parameters from the swell wave system and wind waves ($H_s$,$T_p$ & $\theta$). The parameters are similar to the ones used by [17], however, the used parameters are only from 2 wave systems.

*Data Wrangling:* The aim of the experiment is to predict vessel motion during offshore handling activities. When a vessel is alongside an offshore platform, the vessel speed (over the ground, GPS speed) should be nearly zero. Therefore, the vessels data is filtered on GPS speed to be less than 0.5 knots measured on the(D)GPS speed.(some speed is allowed as the vessel crew will move the vessel very slowly occasionally, to position the vessel relative to the crane position).

The vessel's sensor data is sampled with a frequency of 1Hz, this is sufficient for the processing data points for a roll and pitch period. By using a Fast Fourier Transformation (FFT) (fig:8) on both roll and pitch data, it was found that the roll period of the vessel is around 5-7 seconds (roll), which proved that a sampling frequency of 1 Hz would suffice for this experiment. Data time series must be stationary before it can be used as learning (or test) data in any statistical analysis. This means that the data must have a constant mean, variance and covariance. Besides the stationarity, the vessel's motion data is influenced by operations on the vessel like delivering cargo, water and fuel or any bulk related to the gas production and transportation of gas, like methanol. Especially the delivery of fluids (bulk, water and fuel) or ballasting of the vessel, creates so-called data trending. Data trending shifts the observations like roll and pitch along a moving average. To test both trending and non-stationary data time series, the so-called Dicky-Füller and Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test were implemented. These tests and de-trending of the data are done using the Python module from statsmodel [37]. After de-trending the data a check is performed using a so-called auto-correlation function with a lag of 10 periods (fig:10a-10c).

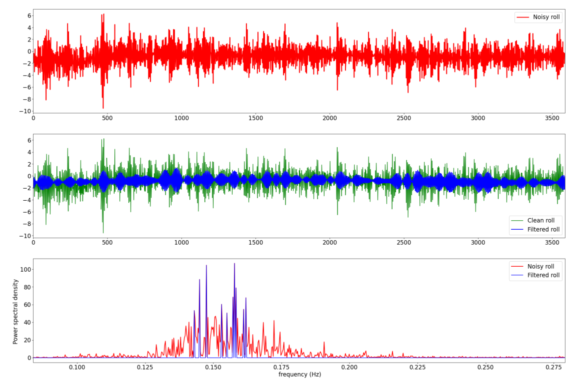



Fig. 8: Fast Fourier Analysis of roll data

Observation data from offshore locations was obtained with a sampling rate of 10 minutes. To synchronize the vessel data, and the observation data, the vessel data must be down-sampled from 1Hz to 10 minutes (600 samples at each data point). Using the Scikit-Sklearn [38] and Pandas [39] re-sample module, the vessel data is synchronized down-sampled and merged with the observation data from Rijkswaterstaat. Any missing values or discontinuity in the data are corrected using the K-Nearest Neighbor(KNN) strategy from a simpleImputer module. The latter ensures that any missing data is imputed as the most probable value in a time series instead of the mean value.

[6] in order to accommodate the vessel sensors on the cloud server, a dedicated node with measuring points was developed for this experiment by Onboard developers

[7] see: https://www.peplink.com/

*data reporting:* Once the featured data is filtered, cleaned, de-trended and made stationary, the data can be submitted to a machine learning model for learning, tuning and testing the models' parameters. To understand the feature importance in the three models (roll, pitch and heave), the data is tested for correlation, the mean decrease in impurity (MDI) and permutation importance (fig:9). The MDI is used to verify the GINI impurity[8] of the tree leaves, while the permutation importance can be seen as a sensitivity test on the (unseen) training features. The permutation feature importance is defined to be the decrease in a model score when a single feature value is randomly shuffled. By using the feature correlation mapping, the correlation can be analysed between the roll, pitch and heave and the features like wind speed and direction, significant wave height and direction, and wave periods. This gives an indication of how strongly the data is correlated with the expected results.

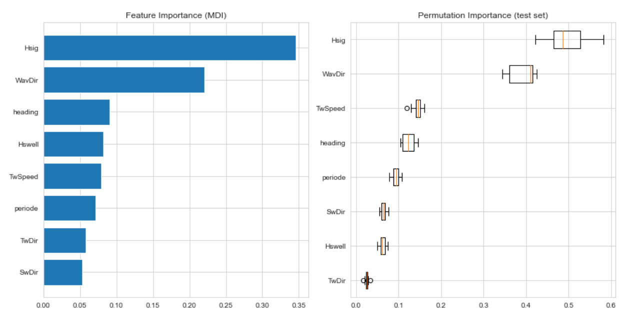



Fig. 9: MDI and Permutance tests on heave data

*Machine learning model*

Gradient Boosting Tree models played a significant role in the discovery of the Higgs_Boson at the Large Hadron Collider[9], but it is also widely applied in earth science questions evaluating sandstone reservoirs and search engines like Yahoo.

From the literature review in section II it can be noticed that the majority of the vessel motion modelling research covered either Neural Network solutions, Support Vector Regression or Tree models. While only the last solution focused on a planning problem with a longer time window, the Gradient Boosting Tree model was selected to be used for this experiment.

While decision trees (like k-nearest neighbour) were considered not very competitive in accuracy, they can become very strong in regression and classification by using either bagging (bootstrap aggregating) or boosting weak classifiers in a voting scheme and transfer in a strong classifier. Gradient Boosting Tree Regressors are doing exactly that. The weak learners are the individual decision trees. All the trees are connected in series and each tree tries to minimize the error of the previous tree. Due to this sequential connection, boosting algorithms are usually slow to learn (controllable by the developer using the learning rate parameter), but also highly accurate. Boosting is the remedy to a so-called high-bias problem. With data sets that consist of limited data and limited features, the use of boosting reduces the bias. The pseudo algorithm for gradient boosted tree is surprisingly small and presented by the following code in algorithm 1 (for a full explanation on Gradient Decision Trees, please refer to the book "The Elements of Statistical Learning" [40])

$F_0 = median\{y_i\}_1^n$

**for** *m=1:M* **do**

$$r_{m-1}(x_i) = y_i - F_{m-1}(x_i), i = 1, N$$

$$\sigma_m = quantile_\alpha\{|r_{m-1}(x_i)|\}_1^N$$

$$\tilde{y} = \begin{cases} r_{m-1}(x_i) |r_{m-1}(x_i)| \leq \sigma_m \\ \sigma_m.sign(r_{m-1}(x_i))|r_{m-1}(x_i)| > \sigma_m \end{cases}$$

$$\{R_{lm}\}_1^L = L - terminalnodetree(\{\tilde{y}_i, x_i\}_1^N)$$

$$\tilde{r}_{lm} = median_{x_i \in R_{lm}} r_{m-1}(x_i), l = 1, L$$

$$\tilde{\gamma}_{lm} = \tilde{r}_{lm} + \frac{1}{N_{lm}} \sum_{x_i \in R_{lm}} sign(r_{m-1}(x_i) - \tilde{r}_{lm}).min(\sigma_m, abs(r_{m-1}(x_i) - \tilde{r}_{lm}), \text{l=1,L}$$

$$F_m(x) = F_{m-1}(x) + \gamma_{lm} 1(x \in R_{lm})$$

**end**

**Algorithm 1:** Gradient Boosting in pseudo code

with

$M$ = number of iterations,

$N$ = number of features,

$L$ = number of trees (terminal nodes)

$\tilde{r}$ = pseudo residual

$\gamma$ = approximation based on minimization of the loss function

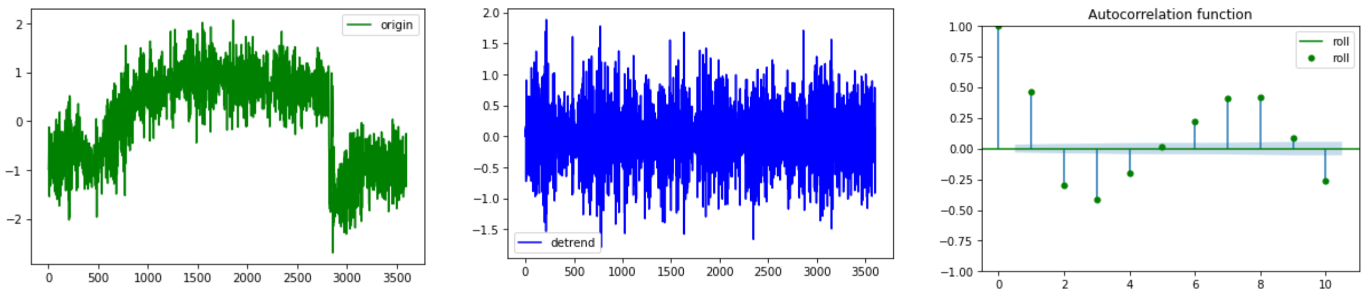
The value of the transition point $\sigma$ depends on the iteration number *m*. The $\sigma_m$ is the $\alpha$ quantile of the distribution of the pseudo residuals $(r_{m-1}(x_i)) = \{|y_i - F_{m-1}(x_i)|\}_1^N)$. $\alpha$ sets the breakpoint whereby the partition of observations can be changed without degrading the quality of the outcome. The value of $\alpha$ will be subject to the (random) grid search during the tuning process of the model.

The loss function $L$ can be any of the functions as presented in the table: IV.

In the vessel motion dataset, outliers may be present. The Huber loss function [16] is particularly good with assigning a smaller weight to outliers as it will behave as a squared function when the loss value is smaller than a preset factor $\sigma$ and behaves as an absolute value at data outliers with a high loss value. Therefore, the model will be using the Huber loss function. This is presented in the figure 11.

[8] see: https://en.wikipedia.org/wiki/Decision˙tree˙learning

[9] https://home.cern/science/accelerators/large-hadron-collider




(a) original roll data (b) de-trended roll data (c) autocorrelation test roll data

Fig. 10: detrending data

TABLE IV: Loss functions

| Setting | Loss Function | $-\sigma L \frac{(y_i, f(x_i))}{\sigma f(x_i)}$ |
|---|---|---|
| Regression | $\frac{1}{2}(y_i - f(x_1))^2$ | $y_i - f(x_i)$ |
| Regression | $\lvert y_i - f(x_i)\rvert$ | sign $\lvert y_i - f(x_i)\rvert$ |
| Regression | Huber | $(y_i - f_{m-1}(x_1))$, $\lvert y_i - f_{m-1}(x_i)\rvert \leq \sigma$<br>$\sigma_m sign\lvert y_i - f_{m-1}(x_i)\rvert$, $\lvert y_i - f_{m-1}(x_i)\rvert > \sigma$<br>where $\sigma_m$ = $\alpha^{th}$quantile $\{\lvert y_i - f(x_i)\rvert\}$ |

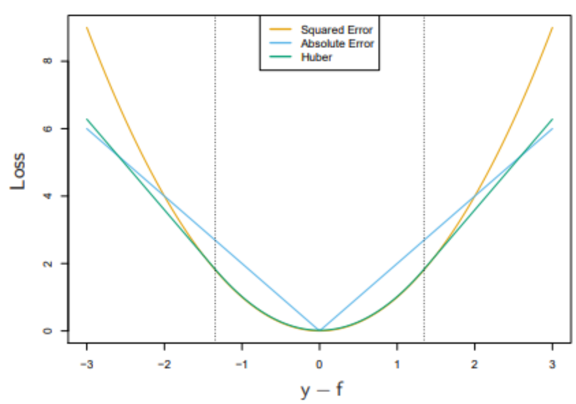



Fig. 11: A comparison of the Loss functions from table IV taken from [40]

*Validation strategy:* The data set is rather small (1510 data points). Therefore, the risk of having a high-variance problem is imminent. To overcome this risk and improve variance while bias remains unchanged, the training set is split up into 5-fold smaller data sets, the so-called KFOLD cross-validation strategy. To minimize the variance, many rounds of data set partitioning into split data sets are being performed and the training results are then combined (averaged). The original KFOLD validation strategy assumes that the data is coming from a single source and is having the same distribution and without referential dependency. The data set from the vessel data however is a time series data set and therefore does not fulfill this requirement. To overcome this limitation, a variation of KFOLD validation is being applied, called TimeSeriesSplit validation by the Scikit-Sklearn module in Python.

*Model tuning:* Tuning a Gradient Boosting Tree model can be time-consuming because the trees are sequentially connected and predict their predecessor pseudo residuals. The objective is to have a model tuned in less than 5 minutes. The parameters which must be tuned are:

- number of estimators
- learning rate
- maximum depth of a tree
- $\alpha$, the Huber quantile
- minimum samples needed per leave
- minimum samples split
- maximum amount of features (needed for bagging)

To prevent slowing down the training of the model and possible over-fitting, the depth of the trees should be limited between 4 to 8. Also for the learning rate care should be taken not to set this rate too low with the risk of over-fitting, or too high with a risk of too much generalization. The number of trees is represented in the model by the number of estimators. Again the higher the number of trees, the slower the model becomes in training. During the tuning of the model, early stopping is used to find the "sweet spot", which is the number of estimators whereby the training error and test error start to diverge (see fig:12). Early stopping is achieved during the training by observing the tolerance over the loss over 20 iterations, when the loss is steady over 20 iterations, the training will stop.
The maximum depth of the tree determines how deep the build tree can be. The more splits the tree has and the amount of information within the tree.

The search grid and the cross-validation which will be used for the tuning of the model are shown in the table V. The scoring metric for the cross-validation will be chosen as Mean Squared Error (MSE).

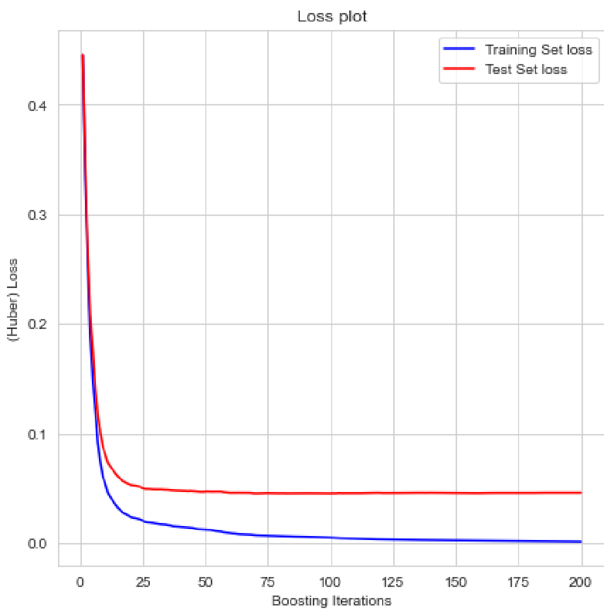



Fig. 12: training curve GBT model with early stopping

TABLE V: Hyper parameter Search Grid

| Parameter | Argument | Values |
|---|---|---|
| Nr of Trees | N-Estimators | [50, 200, 400] |
| Step size | learning˙rate | [0.2,0.1,0.01,0.001] |
| Tree Depth | max˙depth | [4,5,6] |
| Huber quantile | $\alpha$ | [0.95,0.5,0.05] |
| Samples per leave | min˙samples˙leaf | [5,7,10] |
| Samples per split | min˙samples˙split | [3,4,5] |
| Max nr of features | max˙feature | [1,2,4,8] |

*Forecast data*

The data used to create the vessel motion prediction is requested by a REST API call from the weather forecaster Infoplaza[10]. ( fig:5) . The submitted API data table is following the ECMWF model[11]. The data interval is one hour, therefore, the prediction results will be presented with the same interval. The following data points are being used:

- windDirection10m
- windSpeed10m
- significantWaveHeight
- meanWavePeriod
- meanDirectionSwell
- significantHeightSwell
- maximumWaveHeight
- peakWaveDirection
- meanPeriodSwell

The data from the API is converted to a JSON[12] object using Python, which is then further processed into an array and read into the machine learning model. The vessel motion prediction shall be valid for a selected number of locations (Dutch offshore blocks A-Q[13]) for the SNSPOOL, all locations will be requested by location key and grouped into an array. By requesting all locations, forecasts for the different locations can be presented by using only one mouse click. The model results (roll, pitch heave predictions) will be stored in a Pandas data frame with the columns being the vessel headings.

## Results

During the late spring and autumn of 2022, data was recorded from Dina Scout. The summer months of 2022 did not have sufficient valid observations on offshore weather waiting times. This resulted in a data set of 1 Million sample points after which 906000 were used to merge with the vessel observation records and resulted into 1510 feature points.
Once the data was cleaned and de-trended, the analysis took place on four features with the highest correlation scores, wind direction, vessel heading during offshore operations, swell direction and sea wave directions. From figure 15, it can be observed that the majority of observations are westerly oriented, whereby the swell direction is merely Northerly distributed. Most offshore platforms are orientated with respect to prevailing wind and wave direction, therefore, the majority of the observations are in either a Westerly or an Easterly direction. As a Gradient-Boosting Tree is not very strong in extrapolating, the heading selection will therefore be limited to the headings SW, NW and NE.
The tuning of all three models took place using a 2-step tuning, 1: alpha tuning with preset hyperparameters, 2: a randomized grid-search tuning with the grid from table V. The objective is to achieve full model tuning within 5 minutes with an $R^2 > 0.85$ and MSE $\leq 10\%$ of the maximum limits. The values achieved after tuning are summarized in table VI.

TABLE VI: Tuning results

| Parameter | Roll | Pitch | Heave |
|---|---|---|---|
| Nr of Trees | 50 | 50 | 50 |
| Step size | 0.1 | 0.15 | 0.15 |
| Tree Depth | 8 | 6 | 8 |
| Huber quantile | 0.95 | 0.95 | 0.95 |
| Samples per leave | 10 | 10 | 7 |
| Samples per split | 4 | 5 | 4 |
| Max nr of features | 8 | 4 | 4 |
| $^{a}R^2$ | 0.81/0.88 | 0.86/0.87 | 0.83/0.92 |
| Training accuracy score | 0.944 | 0.932 | 0.971 |
| MSE | 1.19° | 0.459° | 0.045 [m] |
| Tuning Time (s) | 00:10:10 | 00:04:31 | 00:04:35 |

[a] before and after model tuning

As can be noticed from the (tuning) table VI, the test square correlation coefficient $R^2$ is between 0.87 and 0.92, this is an improvement from before the grid search around 10%. The MSE observed for the roll data is around 1.19°, which is comparable with the earlier study by [17]. For a vessel motion

[10]https://www.infoplaza.com/nl/marine/marine-weather
[11]European Centre Medium Weather Forecasts
[12]Java Script Object Notation
[13]https://www.nlog.nl/olie-en-gaskaarten-van-nederland

prediction with a forecast on an hourly scale, this is sufficient to be used. The tuning takes approximately 4 minutes for the heave and pitch model, while the model tuning for the roll is somewhat more complex and takes around 10 minutes. The aim was to stay below 5 minutes, which could not be achieved without losing model quality for the roll model .

Validation of the test data is represented in figure 14. From all three models it can be observed that generally, the model prediction follows the actual test data but some outliers exist in the higher motion values.

The hydrodynamic model was used for the simulation of the vessel motion using the MO4 dashboard in the forecast application. The results were compared with the forecast weather data from Infoplaza and the machine learning prediction for the heading of 315°. In figure 15a, the Gradient Boosting Tree prediction is shown as the grey area below the $H_s$ warning line. The model predicts vessel motion workability below zero per cent (red dotted) up until the $26^{th}$ noon of February 2023. The hydrodynamic output shows a similar pattern for that date in figure 15b and figure 15d. Both figures are showing a similar time window for the unworkable period based on the same limitations for the vessel from table II(10° roll). Also, the warning issued by the weather forecast Infoplaza-based figure 15c on wave height limits is showing a similar result. Figures 16a and 16b represent the forecast roll most probable maximum prediction and the measured roll values on the $2^{nd}$ of May 2023. The vessel started its offshore handling at 15.40 LT until 16.30 LT (UTC +2). The model prediction shows a roll angle of 5.69°, while recorded values showed 2° and 6°, with the maximum values during the first 15 minutes of the offshore handling. After 16.30, the vessel moved out of the 500m zone and roll values were reduced. A similar pattern can be observed in figure 17 taken from the Onboard dashboard[14]: after 16.30 vessel leaves the offshore side and roll values are reduced from 6° to less than 2°.

## CONCLUSION

The objective is to use machine learning as a tool to predict the vessel's workability during offshore handling alongside production platforms. It was observed during the planning of the vessel schedules, that the information from weather forecasters is available to planners, however, the missing link is the resulting vessel motion due to the wave forces. With the use of hydrodynamic models, vessel motion can be calculated for a certain location at a certain time window. The latter is causing the effect that either information is fragmented (missing the overlay between weather forecast data and the resulting motion information) and it takes considerable time and effort to see this information for all offshore blocks in the Dutch offshore sector. By using vessel motion sensors, a motion model could be built using a Gradient Boosting Tree regression. Building sufficient data to achieve sufficient prediction accuracy took considerable time, more than 3 months were needed to build the final data set. However, building the dataset and tuning the models only need to be done once, thereafter the dataset continues increasing with valid observations. By developing a web-based prototype which not only presents the vessel motion but also the forecast parameters, the planners are able to see the complete set in a blink of an eye. By using the API from the weather forecaster and requesting all weather predictions for the competitive Dutch sector, the planners are able the switch predictions from one block to the next with only one mouse click while this took more than 12 hours in the previous dashboard set up. The accuracy of the machine learning model is sufficient for marine planning purposes, however, if a special lifting operation must take place with lower limits and margins, it is still recommended to simulate this using the vessel motion equation and diffraction methods for the sake of accuracy.

This experiment was conducted as part of a master's graduation at the Maritime Institute Willem Barentsz.

I like to express my gratitude to the following persons for their assistance and advice:

mr. Erwin Strik and mr. Florus Wilming, Onboard
mr. Olaf Sueters, Infoplaza Business BV
mr. Mark Paalvast, MO4-Mocean Forecast BV.
mr. Stephan Procee, Lecturer navigation at Willem Barentz maritime institute.

[14]the Peterson Onboard dashboard is built using graphana: https://grafana.com

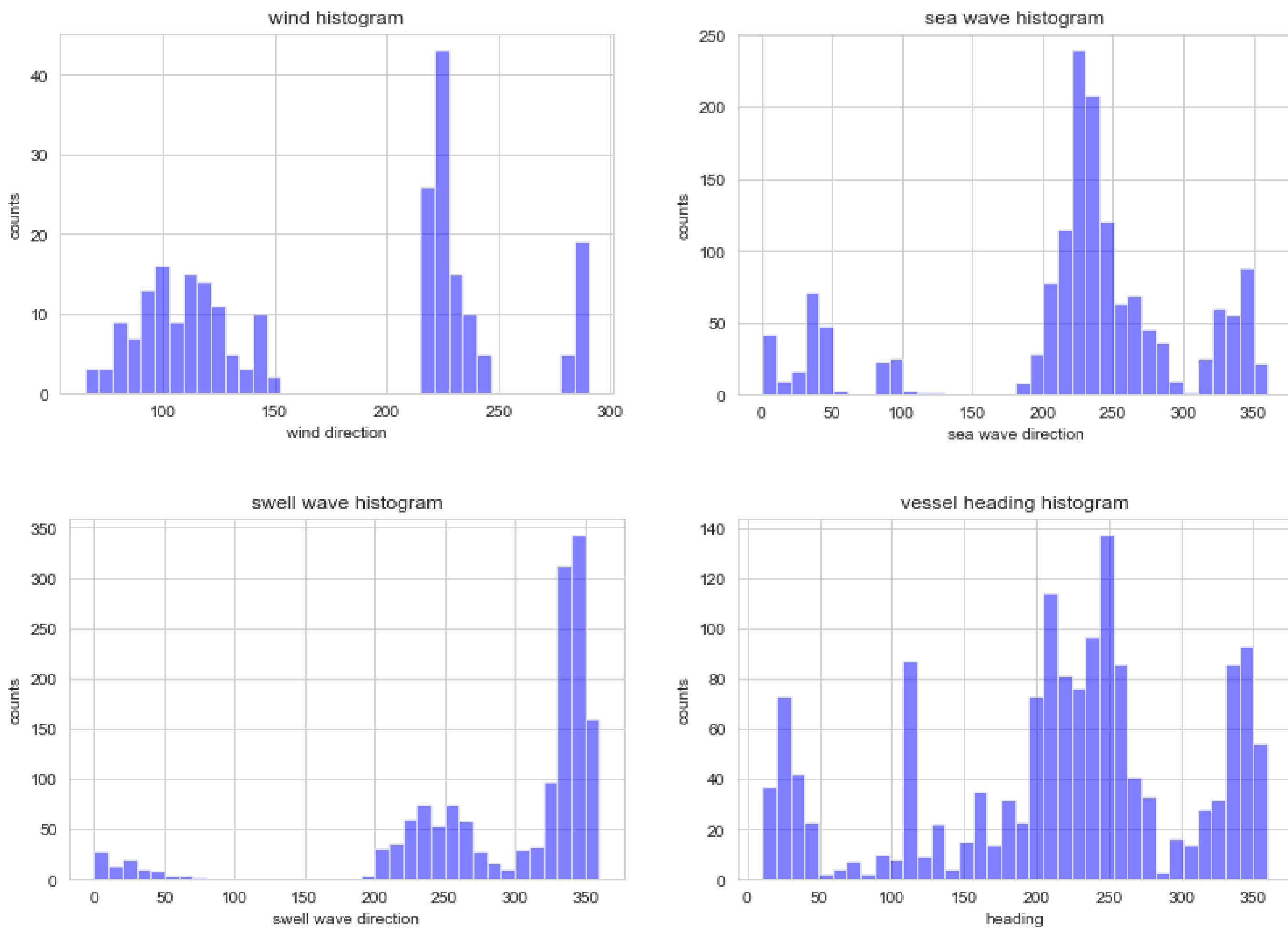
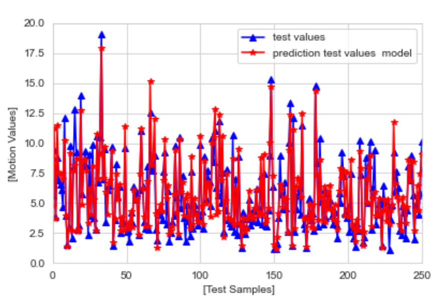
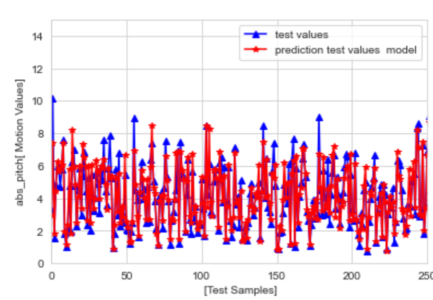


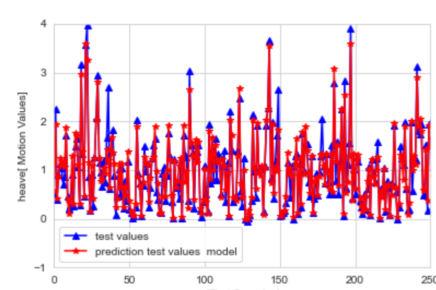

Fig. 13: Data distribution on the four features with the highest correlation values. The heading distribution is primarily determined by the orientation of the production platforms.




(a) validation roll data

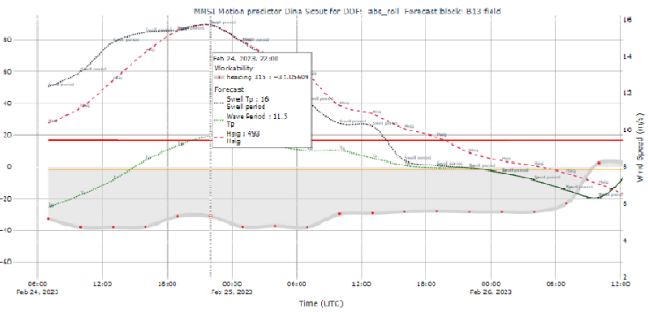



(b) validation pitch data

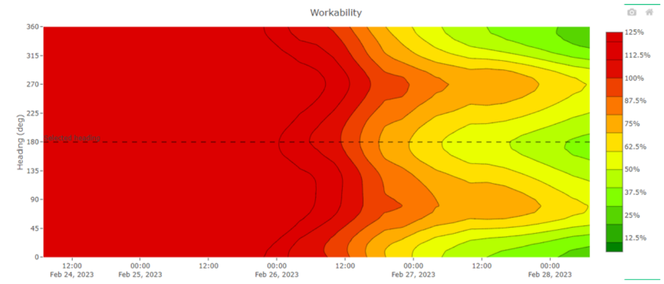



(c) validation heave data

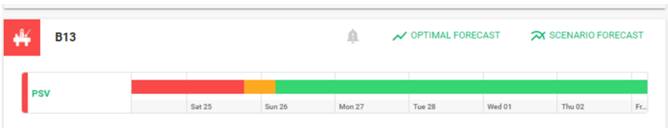
Fig. 14: Machine learning model outcome validation of test data for roll, pitch and heave motion



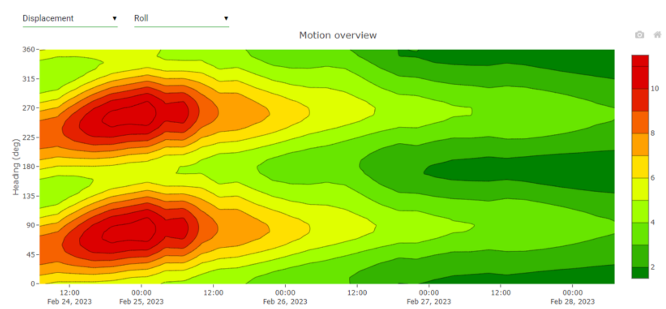

(a) Machine Learning roll model prediction




(b) hydrodynamic model prediction

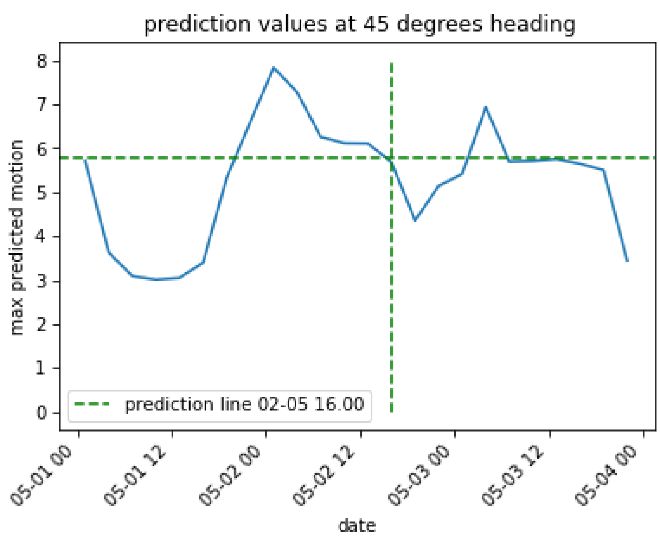



(c) Meteo forecast prediction

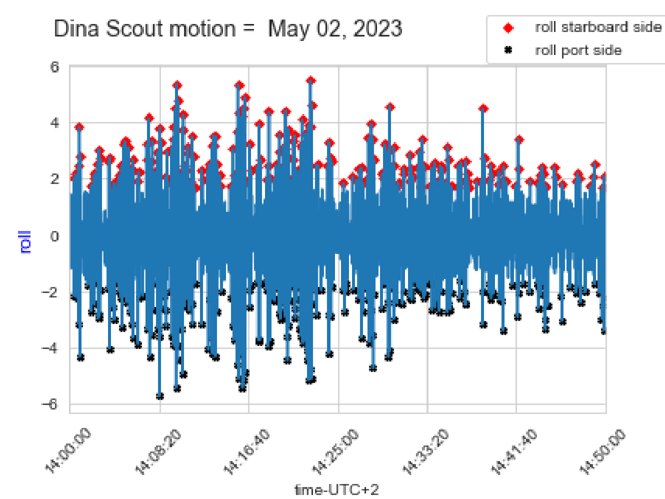



(d) hydrodynamic roll motion prediction

Fig. 15: The comparison between the machine learning model prediction left top with the hydrodynamic motion predictions and the forecast prediction by forecast data.




(a) Most probable maximum roll prediction




(b) Measured (detrended) roll motion from 14.00Z for 2 hours during offshore handling

Fig. 16: 16a the prediction results for most probable maximum roll motion on may $2^{nd}$ 2023, 16b: the actual measurement taken from the vessel motion sensor. The model prediction is 5.69°,the measurement shows a maximum motion of 5.5° starboard and 5.8° port.

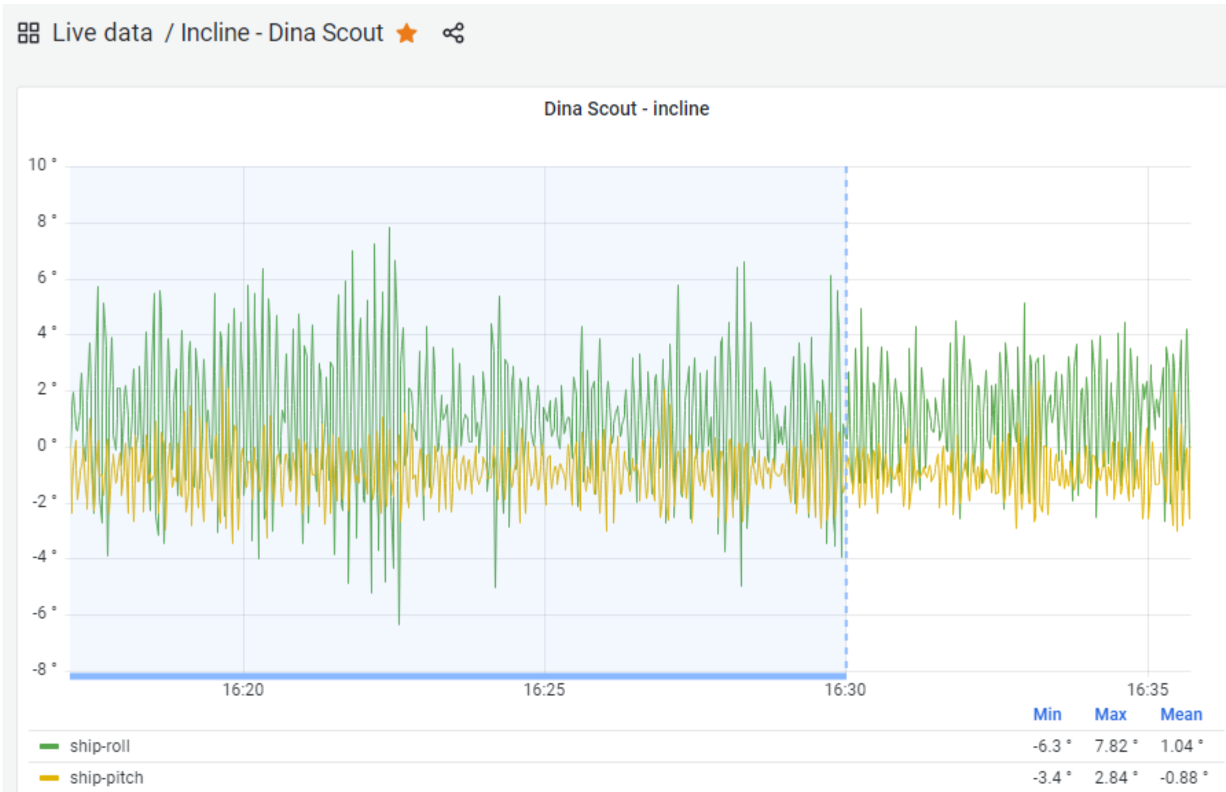


Fig. 17: Realtime data recorded during offshore handling time (blue shaded)

# References


[1] R. Islam, F. Khan, R. Abbassi, and V. Garaniya, "Human error assessment during maintenance operations of marine systems – What are the effective environmental factors?" *Safety Science*, vol. 107, no. April 2017, pp. 85–98, 2018. [Online]. Available: https://doi.org/10.1016/j.ssci.2018.04.011

[2] V. Rumawas, "Human Factors in Ship Design and Operation : Experiential Learning," Ph.D. dissertation, NTNU, 2016.

[3] A. Galieriková, "The human factor and maritime safety," *Transportation Research Procedia*, vol. 40, no. Transcom, pp. 1319–1326, 2019. [Online]. Available: https://doi.org/10.1016/j.trpro.2019.07.183

[4] J. D. D. Groot, "The effect of taking uncertainty into account on the robustness of an offshore scheduling problem," 2022.

[5] L. Peet, "Thesis, Assessing the opportunities of shared logistics between Oil Gas and Wind energy on the North Sea," 2021.

[6] D. Carter, "PREDICTION OF WAVE HEIGHT AND PERIOD FOR A CONSTANT WIND VELOCITY USING THE JONSWAP RESULTS," *Institute of Oceanographic Sciences*, vol. 9, no. l, pp. 17–33, 1982.

[7] J. Journee and W. Massie, *OFFSHORE HYDROMECHANICS*, 1st ed. Delft: Delft University of Technology, 2001, no. January. [Online]. Available: http://www.shipmotions.nl

[8] DNV GL, "DNV-RP-C205: Environmental Conditions and Environmental Loads - Recommended Practice," *Dnv Gl*, no. April, 2014.

[9] L. H. Holthuisen, *Waves in Oceanic and Coastal Waters*. Cambridge University Press, 2007. [Online]. Available: www.cambridge.org/9780521860284

[10] M. Gutsch, S. Steen, and F. Sprenger, "Operability robustness index as seakeeping performance criterion for offshore vessels," *Ocean Engineering*, vol. 217, no. March, p. 107931, 2020. [Online]. Available: https://doi.org/10.1016/j.oceaneng.2020.107931

[11] SINTEF, "SINTEF Vessel Response Tool," 2018. [Online]. Available: http://vrt.sintef.no

[12] DNV, "DNVGL-OS-H101 - Marine Operations, General," *Dnv*, no. October, 2011.

[13] O. T. Gudmestad, *Waiting on suitable weather to perform marine operations*. Springer Singapore, 2019, vol. 22. [Online]. Available: http://dx.doi.org/10.1007/978-981-13-3119-0˙1

[14] S. Wilcken, "Alpha factors for the calculation of forecasted operational limits for marine operations in the Barents Sea," Ph.D. dissertation, 2012.

[15] M. Wu and Z. Gao, "Methodology for developing a response-based correction factor ( alpha-factor ) for allowable sea state assessment of marine operations considering weather forecast uncertainty," vol. 79, no. June, 2021.

[16] J. H. Friedman, "Greedy Function Approximation : A Gradient Boosting Machine," no. 2, pp. 1–34, 1999.

[17] W. Guachamin-Acero and J. Portilla, "Prediction of dynamic responses for execution of marine operations using partitioning of multimodal directional wave spectra and machine learning regression models," *Ocean Engineering*, vol. 262, no. June, p. 112157, 2022. [Online]. Available: https://doi.org/10.1016/j.oceaneng.2022.112157

[18] G. Li, B. Kawan, H. Wang, and H. Zhang, "Neural-network-based modelling and analysis for time series prediction of ship motion," *Ship Technology Research*, vol. 64, no. 1, pp. 30–39, 2017. [Online]. Available: http://dx.doi.org/10.1080/09377255.2017.1309786

[19] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Leaming, 20, 273-297 (1995)*, vol. 7, no. 5, pp. 63–72, 1995.

[20] X. R. Hou and Z. J. Zou, "SVR-based identification of nonlinear roll motion equation for FPSOs in regular waves," *Ocean Engineering*, vol. 109, pp. 531–538, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.oceaneng.2015.08.068

[21] C. Z. Xu and Z. J. Zou, "Online Prediction of Ship Roll Motion in Waves Based on Auto-Moving Gird Search-Least Square Support Vector Machine," *Mathematical Problems in Engineering*, vol. 2021, 2021.

[22] G. W. Flake and S. Lawrence, "Efficient SVM Regression with SMO," *NEC Research Institute*, pp. 239–248, 2001.

[23] A. Haenen, "Neural Network for SSCV hydrodynamics," Ph.D. dissertation, TU Delft, 2012.

[24] K. S. Bremer, "Using Neural Networks to Predict the Response of a Floating Structure," Ph.D. dissertation, NTNU, 2018.

[25] J. H. Friedman, "Greedy Function Approximation : A Gradient Boosting Machine," no. 2, pp. 1–34, 1999.

[26] J. Portilla, F. Ocampo-Torres, and J. Mobaliu, "Spectral Partitioning and Identification of Wind Sea and Swell," pp. 107–122, 2009.

[27] F. Abbas, "Vessel Performance Analysis using High Frequency Operational Data," Ph.D. dissertation, Liege University, 2022. [Online]. Available: http://hdl.handle.net/2268.2/16491

[28] A. Boeien van, J. Daalhuizen, and J. Zijlstra, *Delft Design Guide*, 2020th ed., A. Boeien van, J. Daalhuizen, and J. Zijlstra, Eds. Delft: BIS Publishers, 2023.

[29] R. Myklebusthaug, "Myklebusthaug offshore vessels." [Online]. Available: https://www.mmred.no/index.php/our-fleet/offshore-vessels

[30] P. Kielland and J. Hagglund, "Using DGPS to measure the heave motion of hydrographic survey vessels," *Proceedings of the Annual Meeting - Institute of Navigation*, no. March, pp. 495–504, 1995.

[31] P. Kun, I. Mulder, A. De Götzen, and G. Kortuem, "Creative data work in the design process," *C and C 2019 - Proceedings of the 2019 Creativity and Cognition*, no. July, pp. 346–358, 2019.

[32] B. Kawan, H. Wang, G. Li, and K. Chhantyal, "Data-driven Modeling of Ship Motion Prediction Based on Support Vector Regression," *Proceedings of the 58th Conference on Simulation and Modelling (SIMS 58) Reykjavik, Iceland, September 25th – 27th, 2017*, vol. 138, pp. 350–354, 2017.

[33] E. Strik and F. Wilming, "Onboard." [Online]. Available: : https://onboard-platform.com/platform/

[34] Graphql, "Graphql." [Online]. Available: https://graphql.org/

[35] Rijkswaterstaat, "Water info portal," p. Ministry of Infrastructure and Water Management, 2020. [Online]. Available: https://waterinfo.rws.nl/

[36] M. Meulder de, "Meteo Server." [Online]. Available: https://www.meteoserver.nl/real-time-Zeeweer-API.php

[37] Statsmodel, "Statsmodel." [Online]. Available: https://www.statsmodels.org/stable/tsa.html

[38] Scikit, "Scikit Learn MAchine learning." [Online]. Available: https://scikit-learn.org/stable/modules/svm.htmlsvm-regression

[39] Pandas, "Python Pandas." [Online]. Available: https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.Series.resample.html

[40] T. et. all. Hastie, "Statistics The Elements of Statistical Learning," *The Mathematical Intelligencer*, vol. 27, no. 2, pp. 83–85, 2009. [Online]. Available: http://www.springerlink.com/index/D7X7KX6772HQ2135.pdf

## CONCEPT DESIGN

After the initial data design stage and before the prototype development, the next phase is concept design development. Concept design is experimenting with possible solutions or parts of solutions that may support the final innovative design. From the literature review, three possible models were identified as conceptual solutions for the regression model. Besides this core functionality, elaboration must take place on the presentation to the planning department, user interfaces and prototype development. Regarding the user interfaces, topics like the presentation of the graphics, and user sections such as vessel limits, offshore locations and model type may be varied using mock-ups. The various mock-ups were presented to the end users for their feedback on usability and ease of use. During the concept design phase, prominence must be given to the requirements as identified in the design brief. At the end of the concept phase, the best solution must be identified concerning the machine learning model based on ease of implementation, adaptability, process speed and accuracy. With regard to the implementation, the focus must be on the application development, accessibility, ease of updating the interface and the persistence of the models within the application. To identify the best model to be used in the innovation, three algorithms were programmed in Python 3.9 using Spyder IDE [15]. For all three models, the metrics were identified from the specifications documented in the design brief. These metrics are 1) the accuracy of the predictions for all three degrees of freedom, 2) the timings used for training the models and the final prediction timings, and 3) the ease of maintaining and implementation in the prototype application. The identified solutions from the literature research are Support Vector Machine regression (SVR), Neural Networks (NN) and Gradient Boosting Trees (GBT). For both SVR and GBT, the API coding from Scikit-Learn was used while for the NN the API from Keras [16] (Tensor flow) was used to program the models. The Keras API was selected as the preferred API for the NN model due to the extensive supporting documentation and the use of tuning dashboards (tensorboards) which increased the tuning process as a visual aid. The specifics and parameters used in the three models are presented in the tables:C1..C3.

TABLE C1: Support Vector Regression model.

| Parameters | Descriptions | Values |
|---|---|---|
| C | Regularizer | 3.76 |
| gamma | Kernel parameter | 1.0 |
| epsilon | noise error margin | 0.1 |
| Kernel | Non-Linear | RBF |

RandomGridSearch with cross-validation of 5 was applied to source the best parameters

[15]see spyder: https://www.spyder-ide.org/

[16]see tensor: https://www.tensorflow.org/

TABLE C2: Neural Network model.

| Parameters | Descriptions | Values |
|---|---|---|
| Layer 1 | normalizer | |
| Layer2 | RELU | 1024 |
| Layer 3 | TANH | 1024 |
| Layer 4 | Linear | 1 |
| L1 | Generalizer | 0.01 |
| L2 | Generalizer | 0.01 |
| Drop out | penalty | 0.2 |
| Loss | Huber | |
| Learning rate | step size | 0.01 |
| Optimizer | SGD | |

Keras Tensor board callbacks [17] was used to tune the parameters using a tuning matrix, early stopping is applied if the loss does not move and learning rate decreases to zero after 600 epochs

TABLE C3: Gradient Boosting Tree model.

| Parameters | Descriptions | Values |
|---|---|---|
| N˙estimators | Nr of Trees | 50 |
| Max˙depth | | 8 |
| Min Samples per leaf | | 10 |
| Min Samples per split | | 5 |
| Learning rate | | 0.2 |
| Loss Function | Huber | 0.95 |
| Maximum Features | | 8 |

RandomGRidSearch was used to source best parameters

The main objective of comparing the models is to determine the highest accuracy scores for the model to be used in the prototype. To determine the highest score, the $R^2$ scores and the residual errors are plotted for all three degrees of freedom for all three models. These plots are visualized in figure S1. From the results, it is obvious that the Gradient Boosting Tree is outperforming the other two models but the Support Vector Regression model is not doing much worse than the GBT. Neural Networks are in this case hard to tune and the accuracy is significantly lacking behind the other two models. As can be seen from the figures S1a and S1c, the SVR after optimization performs slightly better with a median absolute error (MedAE) and the $R^2$ score than the GBT model. The Neural Network only achieves a $R^2$ score of 0.712 and can therefore not be considered as a solution. Zooming in on the two other degrees of freedom (Heave and Pitch) for the two remaining models, SVR and GBT, the comparison reveals that the accuracies do not differ significantly. Figures S2a....S2d show the $R^2$ scores for heave and pitch for both GBT and SVR regression models. From these plots, we can conclude that either GBT or SVR would suffice as a model for the innovation, while the GBT model is slightly better performing for heave and pitch motion prediction.

The $2^{nd}$ metric on which the distinction is made, is the computational time. From both models the training time, validation time and prediction times were recorded with the same data set for the roll prediction. Table C4 show that

the GBT model cross-validation computation time is slightly larger than the SVR cross-validation computational times. Cross-validation however, only needs to be done before the model is used. The prediction timings on test set values are comparable.
The last performance benchmarking is a so-called SHAP value (SHapley Additive exPlanations) comparison. As the models are difficult to explain being either grey or black boxes, the SHAP values enables to determine the feature importance better without having to understand in depth about the model's algorithms. It is important to understand the selectivity and the influence of the features on the model prediction outcomes. The data which is obtained from either Rijkswaterstaat or the vessel sensors differ in sample density, therefore, we have a preference for feature importance from vessel sensors with a sampling frequency of 1 Hz. The plots that are being used to determine feature importance are so-called beeswarm plots. The beeswarm plots show the importance of the sorted features in the left Y-axis, while the colour distinction (red and blue) shows the sensitivity for both the low values and the high values on the positive and negative influence of the model prediction. In figures S3a & S3b these beeswarm plots are represented. Similarity can be observed between the two plots, except the features heading & TwSpeed are flipped between GBT and SVR. The beeswarm plot for the GBT is a little bit more pronounced (sharper) than the SVR plot, therefore, it can be concluded that the GBT would be the model to be used in this innovation.

## PROTOTYPE

The prototype is an application designed for PC usage and based on a web application. In a later development, also applications for handheld tools like mobile phones and or tablets may be developed. The development is based on plotly graphics library [18] called plotly dash and is programmed in Python with HTML (dynamic) callbacks. The reason for choosing Plotly is the high coupling with Python and the extensive literature and support which can be found on the internet (like Youtube explanations). The application is built according to some basic design principles for software development, using the "low coupling, high cohesion" principle. Systematically this is represented in figure S4.

The top layer represents the graphical user interface (GUI) and is basically what the user will look at on his PC. This layer's responsibility is only the presentation of information, between the logic layer and the GUI, the interface is created. The interface is built to communicate between the GUI and the logic layer by using the so-called dynamic callbacks (see fig: S6, a kind of listener with dependencies. The logic layer has the responsibility for calculations with the data from the API (forecast data) and the model data. The model data uses a "persistence storage", basically a physical storage (folder). It is important to build an application like this system in order to port the application to a client-server environment. In this case the complete design was built based on a GitHub repository with a virtual environment build in PyCharm. As the application can be called from outside the Peterson domain, security was developed using a login to protect the data from the public. This security resides on the server side.
As observed from the concept phase, the two models SVR and GBT are matching very closely. A decision was taken to use the GBT model for the prototype, however, in the near future, the SVR may be used also. To support the different model types within the Python coding, it was decided to call the models using a polymorphic design as a "vesselMotionModel" Class instance. By choosing polymorphic design we can change the model methods based on object inheritance.
The GUI is responsible for the user interaction. In the design a simple menu was selected for user selection on Offshore Location, the model type (Roll, Pitch, Heave) and the vessel motion limits. Besides the parameter settings, the user may refresh the weather forecast by calling the API. The motion type displacement, velocity and acceleration is a future feature and not yet available.
The final representation is shown in figure S5, the main figure is built from Graphic Objects with scattered traces with markers and lines. The user may select the traces in the right pane which are based on heading selections and forecast parameters. Zooming in on both date (X-axis) and values(Y-axis) is made possible to enhance visibility.

TABLE C4: Training and validation timings.

| | SVR | GBT |
|---|---|---|
| Training data | 260s | 54s |
| Testing data | 0.11s | 0.46s |
| Cross-validation | 834s | 1308s |

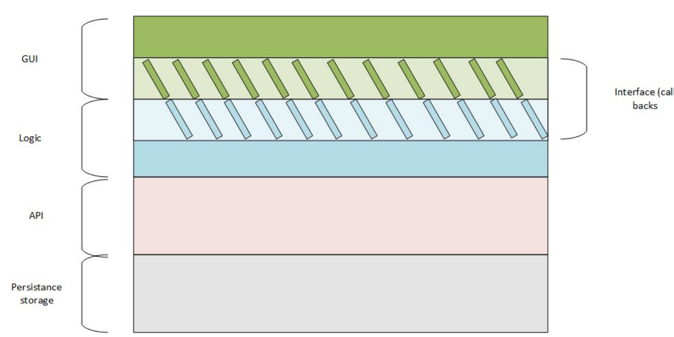
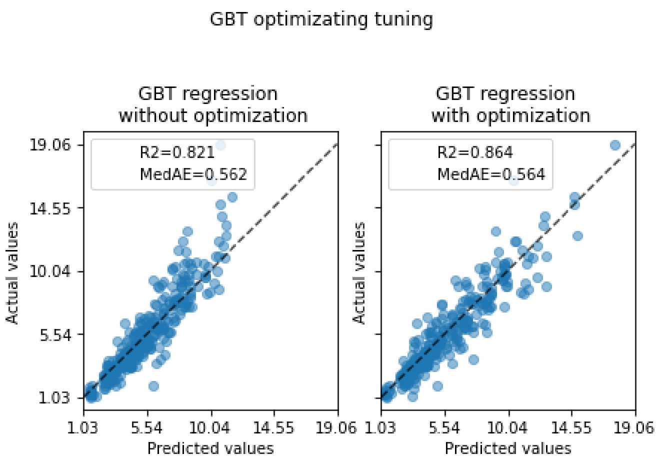
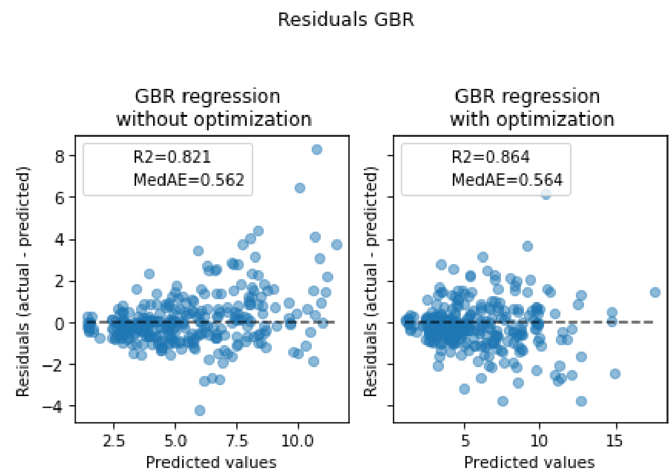


Fig. S4: application system layers

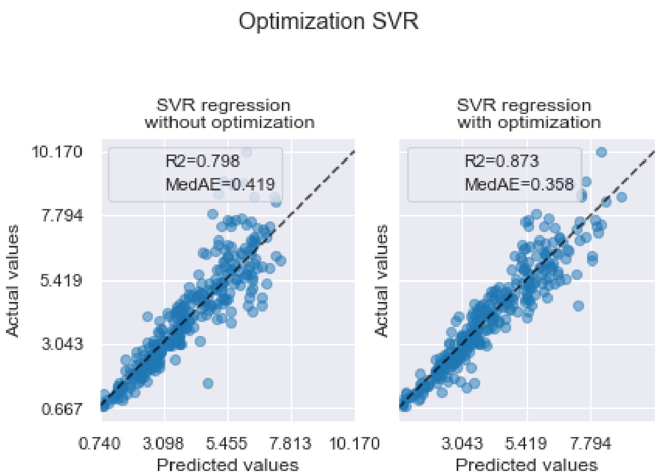
[18]see: https://plotly.com/python/



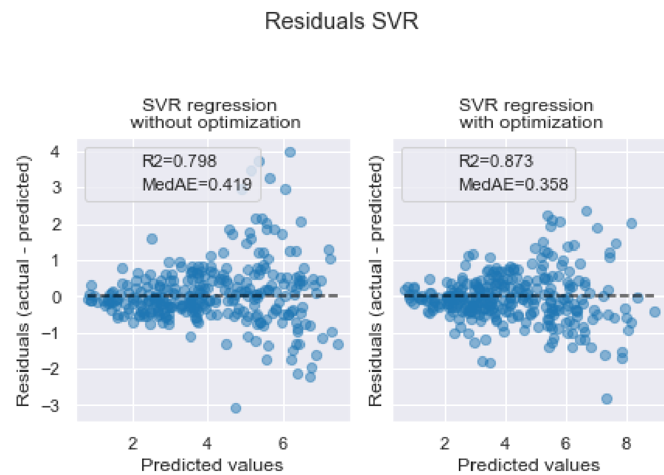

(a) GBT accuracy score



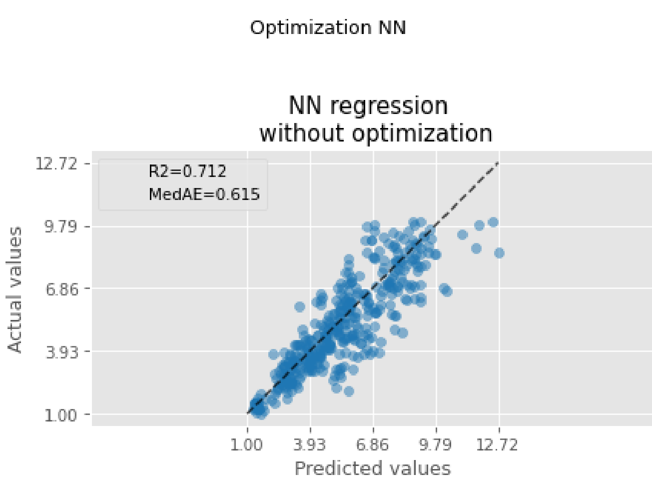

(b) GBT residuals plot



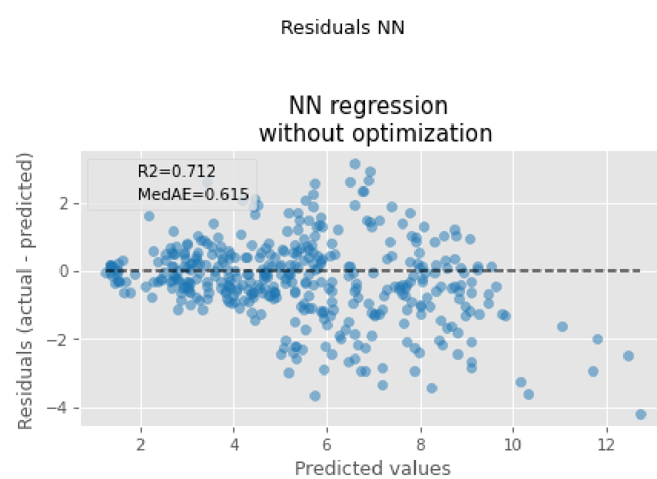

(c) SVR accuracy score




(d) SVR residuals plot

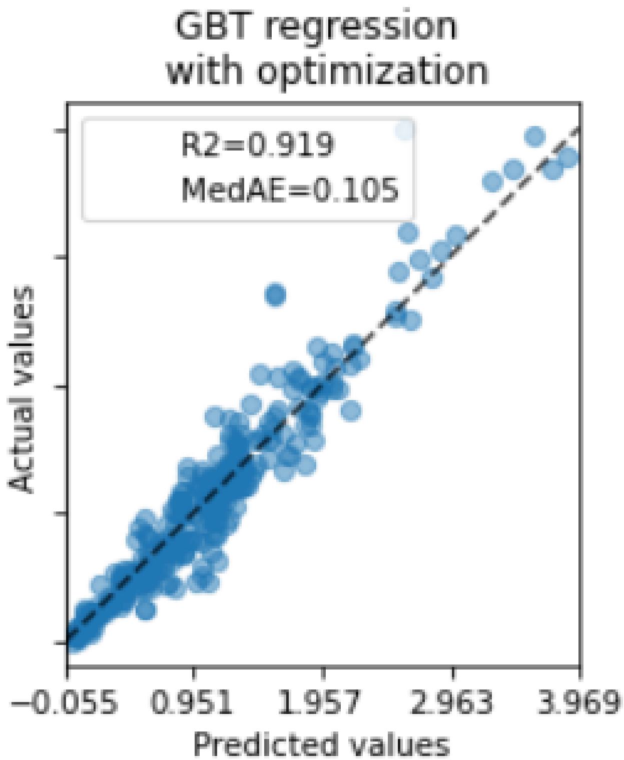



(e) NN accuracy score

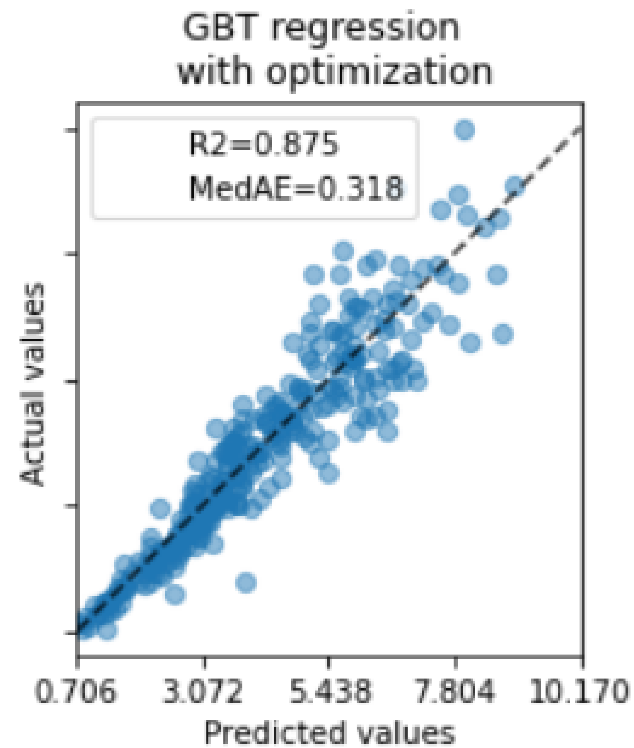



(f) NN residuals plot

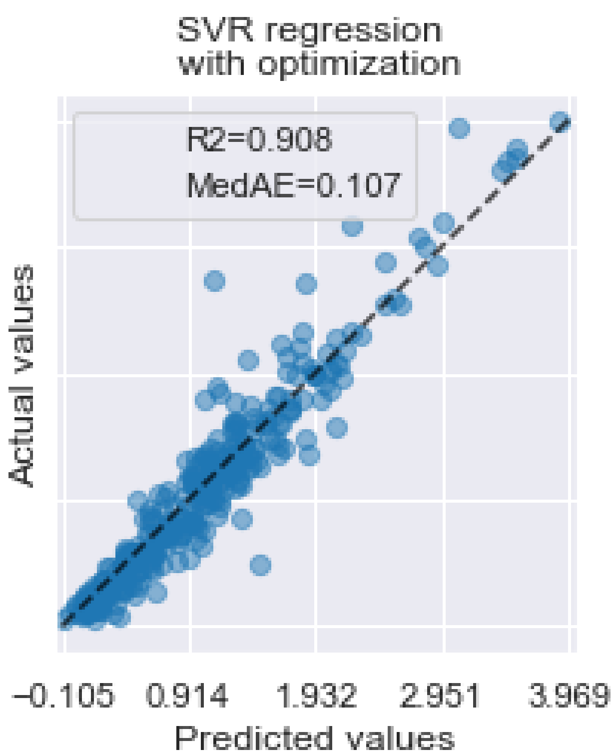
Fig. S1: The accuracy score comparison between the three machine learning models GBT, SVR and NN



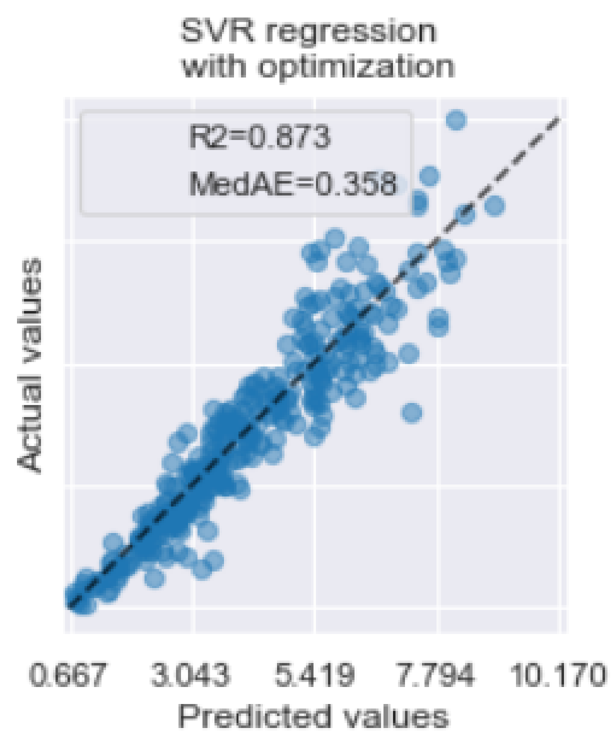

(a) GBT Heave accuracy plot




(b) GBT Pitch accuracy plot

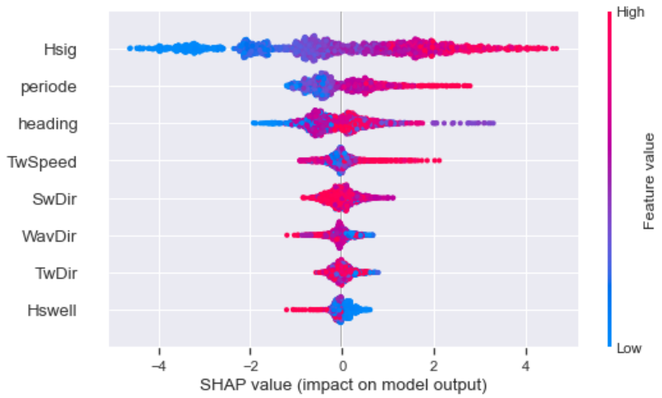



(c) SVR heave accuracy plot

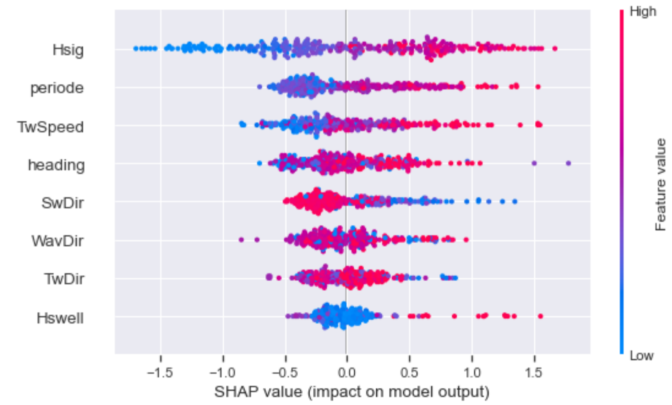



(d) SVR Pitch accuracy plot

Fig. S2: Accuracy scores for heave and pitch for SVR and GBT models




(a) GBT shapley beeswarm plot




(b) SVR shapley beeswarm plot

Fig. S3: shapely beeswarm plots for SVR and GBT models

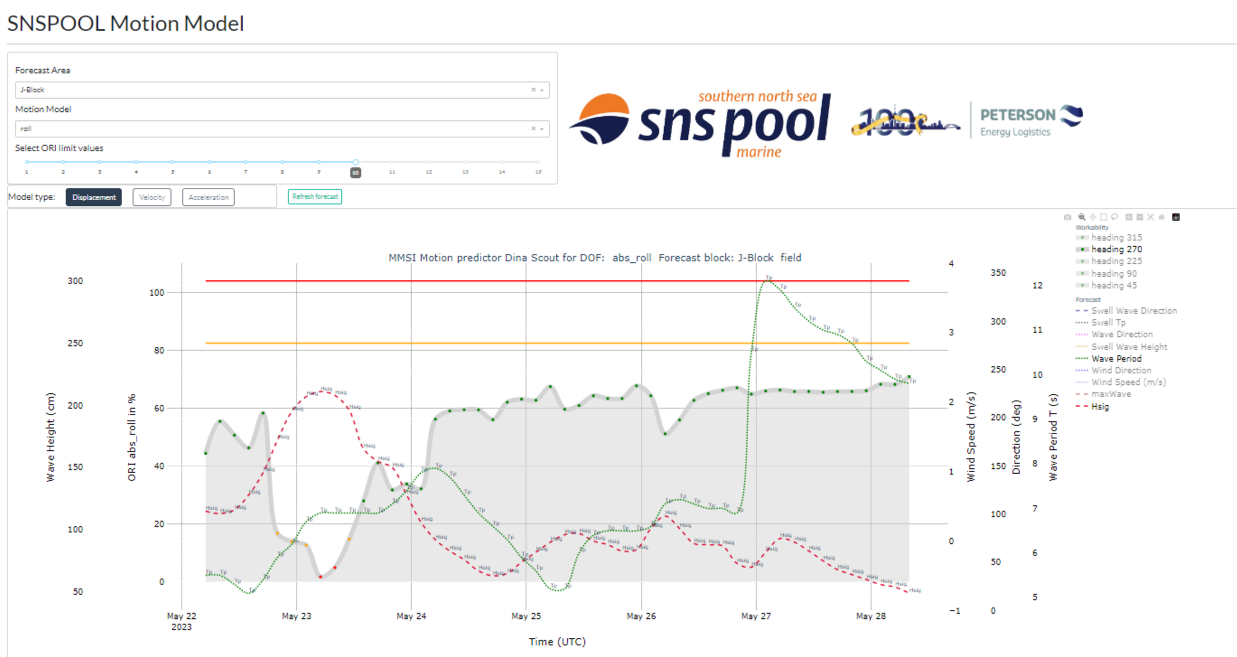



Fig. S5: prototype web application

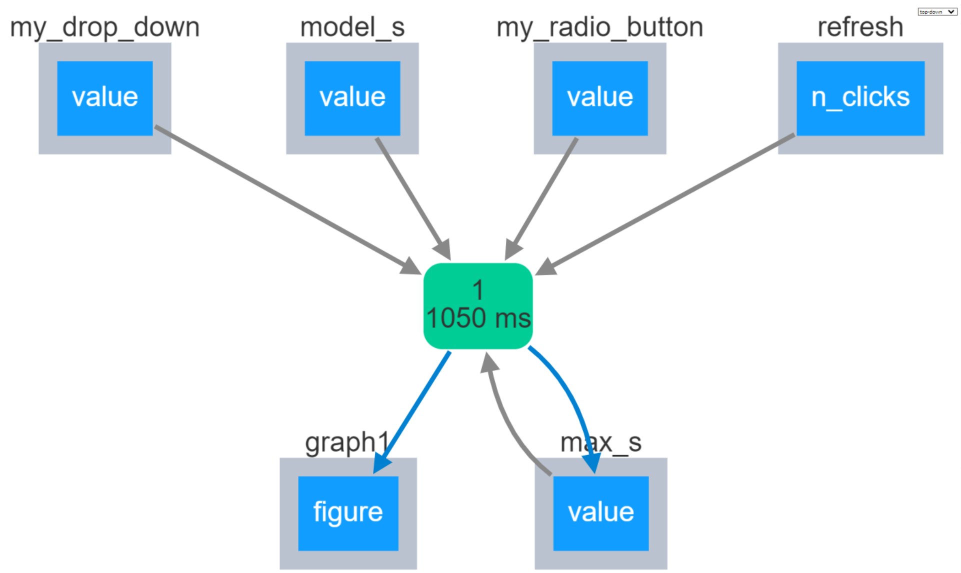



Fig. S6: call back dependencies