A data mining approach for predicting main-engine rotational speed from vessel-data measurements

Dimitrios Kaklis
Department of Informatics and Telematics, Harokopio University of Athens & NCSR Demokritos & Danaos Shipping Co.
Athens, Greece
dkaklis@iit.demokritos.gr

George Giannakopoulos
Institute of Informatics and Telecommunications, NCSR Demokritos
Athens, Greece
ggianna@iit.demokritos.gr

Iraklis Varlamis
Department of Informatics and Telematics, Harokopio University of Athens
Athens, Greece
varlamis@hua.gr

Constantine D. Spyropoulos
Institute of Informatics and Telecommunications, NCSR Demokritos
Athens, Greece
costass@iit.demokritos.gr

Takis J. Varelas
Danaos Shipping Co.
Piraeus, Greece
drc@danaos.gr

ABSTRACT

In this work we face the challenge of estimating a ship's main-engine rotational speed from vessel data series, in the context of sea vessel route optimization. To this end, we study the value of different vessel data types as predictors of the engine rotational speed. As a result, we utilize speed data under a time-series view and examine how extracting locally-aware prediction models affects the learning performance. We apply two different approaches: the first utilizes clustering as a pre-processing step to the creation of many local models; the second builds upon splines to predict the target value. Given the above, we show that clustering can improve performance and demonstrate how the number of clusters affects the outcome. We also show that splines perform in a promising manner, but do not clearly outperform other methods. On the other hand, we show that spline regression combined with a Delaunay partitioning offers most competitive results.

CCS CONCEPTS

• **Information systems** → **Clustering**; *Data mining*; • **Computing methodologies** → **Machine learning**; • **Theory of computation** → *Pattern matching*; *Unsupervised learning and clustering*; • **Applied computing** → *Multi-criterion optimization and decision-making*;

KEYWORDS

time-series forecasting, machine learning, Delaunay triangulation, splines, multivariate regression analysis
Topics Machine learning, Clustering, Time-series analysis

ACM Reference format:
Dimitrios Kaklis, George Giannakopoulos, Iraklis Varlamis, Constantine D. Spyropoulos, and Takis J. Varelas. 2019. A data mining approach for predicting main-engine rotational speed from vessel-data measurements . In *Proceedings of 23rd International Database Engineering and Applications Symposium, Athens, Greece, June 10–12, 2019 (IDEAS'19),* 10 pages.
https://doi.org/10.1145/3331076.3331123

IDEAS'19, June 10–12, 2019, Athens, Greece

ACM ISBN 978-1-4503-6249-8/19/06…$15.00
https://doi.org/10.1145/3331076.3331123

1 INTRODUCTION

Liner shipping companies can benefit significantly by improving ship scheduling and cost analysis in service route planning using computational methods. Furthermore, since there is a strong demand for ships to reduce their emissions, a number of current research activities focus on estimating shipping emissions and developing mitigating solutions to tackle the problem (e.g. [30]). In addition, volatility in fuel prices constitutes a major problem for shipping companies as fuel makes up for 60% of the overall ship operating cost [6]. As a result, modern ship management moves towards energy-efficient procedures and operations, aiming to reduce energy consumption for lowering management costs and thereby maintaining a competitive position in the market while reducing the corresponding environmental impact.

Routing optimization has been a major problem in shipping industry for over three decades and remains one of the research topics of primary interest for the maritime community. Especially nowadays, when new technologies and new concepts, such as *Big Data*, *Data Mining* and *Pattern Recognition* and new methods of data acquisition (AIS data), are overthrowing traditional ways of science exploration, data-driven maritime research is gaining in attention. The automatic identification system (AIS) is an automatic tracking and self-reporting system for identifying and locating vessels by electronically exchanging data among other nearby ships, AIS base

stations and satellites. The widespread use of AIS allowed vessel tracking and increased the availability of ship trajectory data.

The problem of optimal-route-planning takes into consideration the objectives of ship owners for energy consumption and on-time delivery of goods and the restrictions set by the regulatory framework (national regulations, IMO etc). Regardless the specific constraints, what makes the optimal-route-problem so challenging is the time-varying character of weather conditions during the voyage of the vessel. In this work the optimal-route-problem is mainly examined under the aspect of Fuel Oil Consumption (*FOC*) and an optimal-route is this that minimizes the vessel's FOC for a given destination.

As it is well known from ship-powering literature, *FOC* is closely related with the rotational speed (measured in revolutions per minute - *RPM*) of the main engine. In this connection, the optimal route problem could be significantly simplified if a good predictive model for *RPM* is available. To elaborate further along this path, this article summarizes the current status of our work to couple ship's velocity V with main-engine's *RPM* in the context of a non-convex regularized regression estimation problem and in conjunction with the fact that, as marine engineering points out, there is a strong correlation between these two factors. Coupling V with *RPM* will give ship operator the benefit of a tool that does not impose installation requirements on the ship, like sensors for gathering data, instead it can be readily used by only getting from satellites the position of the ship, calculating its speed and getting the weather conditions at each time interval. In the same time, it is a first step towards integrating input from more sources (including weather and sea condition data) and allowing the creation of data driven models (black box models) that are able to predict and optimize vessel consumption.

The rest of the paper is organized as follows: In section 2, a summary of the literature related to the routing optimization problem in maritime industry is provided and analyzed. Section 3 presents the motivation behind the work of this paper and summarizes our initial exploratory experiments. Section 4 describes a formulation of the problem at hand and gives an overview of the proposed algorithm. Section 5 depicts and interprets the experimental results, combined with statistical testing. Finally, Section 7 provides the main conclusions of this work and outlines the next steps.

2 RELATED WORK

Following strong regulatory and societal demand for ships to reduce their emissions, current research activities focus on estimating global shipping emissions and develop mitigating solutions to tackle the problem, e.g. [30]. In addition, the increase and volatility in fuel prices constitute a major problem for shipping companies as fuel contributes approximately 60% to the overall ship operating cost [6]. As a result, shipping companies move towards taking on board energy efficient procedures and operations for reducing energy consumption and thereby maintain their competitive position in the market as well as reduce the environmental impact. There is a plethora of theoretical papers related to ship route optimization, starting as early as 1960 [31] and evolving from using simple concepts, such as the so-called isochrone and isopone methods [7], to more elaborate and rigorous approaches, such as optimal control [29], dynamic programming [24], graph theory [25] and evolutionary algorithms [26].

Numerous studies in different disciplines have been undertaken to predict the fuel consumption by using ANN models [27] . ANN is found to be the domain for many successful applications involving prediction tasks, such as modelling and prediction of energy-engineering systems [22], prediction of the energy consumption of passive solar buildings [10], developing energy system and forecast of energy consumption [1], and analysis of emissions reduction [20]. There are also some relevant reports of ANN's being used for implementing decision-support systems in various subjects, such as solving the buffer allocation problem in reliable production [28], developing environmental emergency decision-support systems [24], risk assessment on prediction of terrorism insurgency [11] and modeling of simulation metamodel [2]. ANNs have been used to predict specific fuel consumption and exhaust temperature of a Diesel engine for various injection timings [21].

The optimization objectives in the ship routing problem are usually the minimization of the voyage time, fuel consumption and voyage risk. The approaches, which have appeared so-far in the pertinent literature, can be classified in two large categories: **(a)** *Vessel-based optimization*, where we optimize a given route with respect to vessel characteristics, e.g., vessel speed, main-engine rotational speed, trim, roll, heave and pitch motions, and **(b)** *Condition-based optimization*, where we optimize a given route by taking into account environmental data, e.g., wind (speed, direction), wave (height, frequency, direction), currents, etc. The aforementioned methods utilize techniques that can be separated into three main categories: **(a)** Analytical approaches trying to tackle the problem with the use of exact(NP-complete) and/or heuristic algorithms like label - setting algorithms , non-linear integer programming, simulated annealing [15]; **(b)** Data-oriented approaches that combine vessel-trajectory data, gathered from sensors or satellites (AIS data), with Machine- and Deep-Learning algorithms [23]; **(c)** Approaches where ML (machine learning) methods, e.g., Box Models: White, Black and Grey Box Models (WBM, BBM, GBM), are combined with analytical methods, e.g., the equations of motions of a freely floating body moving with constant forward speed (WBM), in order to increase the accuracy of a regression method in a ML model (BBM) [3].

Finally, methods that refine the voyage grid (map) in areas of critical interest involving, e.g., weather conditions, emission control areas (ECA, SECA: sulfur-oriented ECA's), high-risk zones (piracy), and choose from a set of optimal routes the best in terms of FOC and safety (PARETO optimal solutions, Genetic Algorithms) [12] must also be referenced.

3 MOTIVATION

The motivation for the current work came directly from a business need for the optimisation of the ship engine usage (RPM) in relation to FOC. Based on this requirement, we attempt first to perform an exploratory analysis on a real dataset in order to understand the nature of the FOC - RPM relation. Given the rich and composite feature set of the FOC prediction problem, before training any multi-parameter prediction model it is important to study the effect of each parameter separately. The exploratory analysis was performed

on a dataset comprising 10^6 observations from multiple ships and allowed us to determine which feature is most appropriate for the prediction of FOC (Fuel Oil consumption). Initial experiments on the complete dataset were performed using feature selection algorithms in order to rank the features by importance. The Random Forest regression was used for selecting the most informative features. The eight top ranked features on the basis of RF regression are depicted in Table 1.

Feature importance		
Feature Name	Importance	Description
RPM	0.98353	Main engine revolutions per minute.
STW	0.00365	Speed through water.
Speed Overground	0.00266	Speed of the ship with respect to the ground.
Apparent wind speed	0.00133	The relative speed, i.e., the speed experienced by an observer or a measuring instrument on the ship.
Port mid draft	0.00075	Draft amidships on the port side of the ship; port is the left-hand side of a vessel facing forward.
STBD mid draft	0.00042	Draft amidships on the starboard side of the ship; starboard is the right-hand side, facing forward.
Mid draft	0.00075	Draft amidships.
Apparent wind angle	0.0007	The relative angle, i.e., the angle experienced by an observer or a measuring instrument on the ship.

Table 1: The top ranked features by importance, using Random Forest regression.

It is clear from Table 1 that RPM plays a pivotal role in the prediction of FOC. Based on this finding, it seems reasonable to develop a predictive model for FOC using RPM only, since it has maximum importance and is much easier to measure than other features (e.g. wind speed or draft). By measuring the correlation between RPM and each of the remaining seven features of Table 1, using **PPMCC** (Pearson Product Moment Correlation Coefficient), showed an extremely high linear relation (0.92) between RPM and speed overground a result that is also aligned with Figure 1 which confirms a strong linear relationship between the two variables.

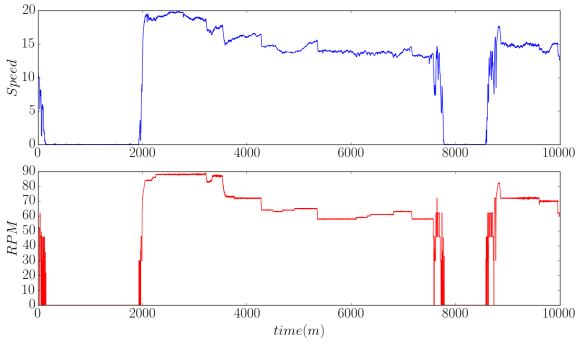

Figure 1: A sample plot of Main-Engine's rotational speed (RPM) and observed speed during a vessel's route (courtesy of DANAOS Shipping Co.)

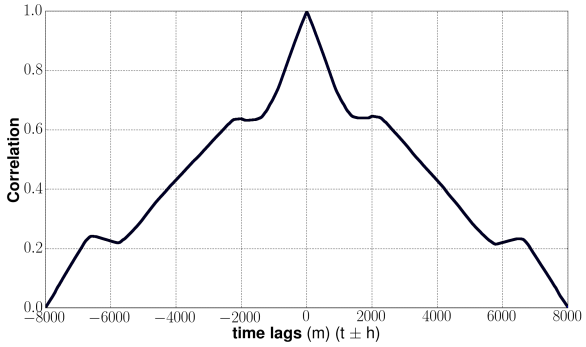

Figure 2: The correlogram of RPM and V during a vessel's route

A survey of the pertinent literature on Naval Architecture and Marine Engineering shows that there is no robust, low complexity, analytical relation between RPM and V. On the other hand, significant work has been done in complex, time-consuming methods that perform well, while taking into account various related factors, such as geometric and hydrodynamic ones [17]. Thus, our effort of finding a way to efficiently predict RPM from V utilizing data-driven based methods is well justified. From ship hydrodynamics it is well known that the , where Q is the torque absorbed by the propeller of the ship. Then, recalling standard resistance and propulsion theory of ships, we can say that, for a given ship, the torque Q is a function depending exclusively on the ratio V/RPM and, as a result, predicting RPM from V is a decisive step for predicting power and thus optimising fuel cost.

. This claim is also strengthened by recognizing the commercial potential value of a model like this, as velocity V is a feature that can be easily measured –even remotely from a satellite– and does not require further installments (e.g. sensors) on board.

In order to further study what happens before and after velocity changes, we plot the correlation coefficient for each lag variable (observations at previous time steps). This gives a quick idea of which lag variables may be good candidates for building a predictive model and how the relationship between the observations and their historic values changes over time.

The correlogram is a commonly used tool for checking randomness in a data set. In time series analysis, the correlogram, also known as an *autocorrelation plot*, is a plot of the sample autocorrelations r_h versus the time lag h. The correlogram of Figure 2 presents the lag number along the x-axis (time axis), with values varying between $-8*10^3$ and $8*10^3$ minutes and the correlation coefficient value (ranging from 0 to 1) along the y-axis. In random behaviors the auto-correlations should be nearly zero for all time-lag separations. In the opposite case, one or more of the auto-correlations should be significantly different than zero. This is the case of RPM in Figure 2, which reveals a strong correlation between RPM and V mainly for time steps $t\pm1, \ldots t\pm10^3$ (m) that can be utilized to select appropriate lag variables as extra features to our estimators.

4 PROPOSED METHOD

4.1 Problem formulation

A formal definition of the problem of predicting RPMs based on the monitored velocity (V) over ground can be defined as follows: *Given a vessel's speed for n consecutive periods, find a function* $f(V_1, ..., V_n) : R^n \rightarrow R^1$, *which estimates the engine's RPM at moment* t_{n+1}.

If we assume that the relationship between *RPM* and V is a partially linear function with non-linear segments over time, then it is possible to describe this specific problem as a *linear mixed-effects model (LMM)* [18]. A mixed model is a statistical model incorporating both fixed and random effects. A random-effects model is a kind of hierarchical linear model, which assumes that the data being analysed are drawn from a hierarchy of different populations, whose differences relate to that hierarchy. These models are useful in a wide variety of applications in physical, biological and social sciences. They are particularly useful in settings where repeated measurements are made on the same statistical units (longitudinal study), or where measurements are made on clusters of related statistical units.

The linear mixed-effects model (LMM) is a great way to model regression algorithms between clustered data and explore the heterogeneity between effects within and between groups of similar values [5]. The connection between *RPM* and V nicely fits the LMM setting, since in most cases there exists some degree of correlation between the two features which implies a linear dependency. Also, in specific moments very similar V values correspond to different values of *RPM* inducing a non-linear dependency. Analytically, an LMM can be described as:

$$y = T \cdot d + u + \epsilon, \tag{1}$$

where y is a vector containing the previously observed values of the feature we want to predict, T is a known matrix that relates the observations y to the unknown fixed-effect vector d, u is the unknown covariate vector for random effects and, finally, ϵ is the unknown vectors of random errors. Both u and ϵ share zero-mean normal distributions with $cov(u, \epsilon) = 0$.

A way to combine *RPM* and V through time (using discrete time slots) is to assume that the following model holds true:

$$\begin{aligned} &y_i = f(t_i) + \epsilon_i, \quad i = 1..., n, \quad t_i \in [0, T], \\ &y_i := RPM_i, \quad f(t_i) := g(V(t_i)) \end{aligned} \tag{2}$$

where RPM_i is ME rotational speed measured at time t_i, $V(t_i)$ is the ship's speed measured at time t_i and $g(V)$ is the sought-for underlying function that, when composed with the known function $V(t)$, gives, for $t = t_i$, the corresponding RPM_i with error ϵ_i. For continuous time t, the above equation can be written as

$$\begin{aligned} &y(t) = f(t) + \epsilon(t), \quad t \in [0, T], \\ &y(t) := RPM(t), \quad f(t) := g(V(t)) = (g \circ V)(t), \end{aligned} \tag{3}$$

Since the measurement of V and *RPM* usually results in many noisy observations a function learned from data can have the form of a smoothing spline that balances between goodness and smoothness of fit. A smoothing spline $\hat{\mathbf{f}}(t), t \in [0, 1]$ in the Sobolev space $\mathcal{H}^{m,2}$, consisting of L^2 functions whose weak derivatives of order up to m belong to L^2 as well, is a solution of the following minimisation problem

$$min_{f \in H^{m,2}} \Big[\frac{1}{n}(y - f)^T W (y - f) + \lambda \int_0^1 (f^m(t))^2 dt \Big], \tag{4}$$

where $y = (y_1, ..., y_n)^T$, $f = (f(t_1), ...f(t_n))^T$ and W is a given positive definite matrix accounting for the correlation between the components of the error vector ϵ. The parameter λ controls the trade-off between fidelity-to-the-data and smoothness of fit and is often referred to as the *smoothing parameter*. In [13, 14] it is shown that the solution of (4) can be expressed as:

$$\hat{f}(t) = \sum_{\nu=1}^{m} d_\nu \phi_\nu(t) + \sum_{i=1}^{n} c_i R^1(t, t_i), \tag{5}$$

where $\phi_\nu(t) = t^{\nu-1}/(\nu - 1)$, $\nu = 1, ..., m$ is a set of polynomials and $R^1(s, t) = \int_0^1 (s-u)_+^{m-1}(t-u)_+^{m-1} du/((m-1)!)^2$ with $x_+ = x$ if $x \geq 0$ and $x_+ = 0$ if $x < 0$ otherwise, is a polynomial spline of degree $2m - 1$, yielding the well-known cubic spline for $m = 2$. Denoting $\hat{T} = \{\phi_\nu(t_i)\}_{i=1,\nu=1}^{n,m}$, $\hat{\Sigma} = \{R^1(t_i, t_j\}_{i=1,j=1}^{n,n}$, one can prove that

$$(\hat{f}(t_1), ..., \hat{f}(t_n))^T = \hat{T}d + \hat{u}, \quad \hat{u} := \hat{\Sigma}c, \tag{6}$$

where $c = (c_1, ...c_n)^T$ and $d = (d_1, ..., d_n)^T$ are solutions to the so-called Henderson's Mixed Model Equation (MME) [8]. Finding the spline estimator for the Linear Mixed-Effects Model (LMM) described in Equation (1), can be done using the Best Linear Unbiased Estimators (BLUE) method [9]. As a result, using spline estimators as a model for revealing the underlying relationship between V and *RPM* seems to be rational and well grounded.

4.2 Partitioning the input space

In order to take advantage of the ability of **Splines** to fit to **LMMs** problems and adapt on the local (temporal) nature of this correlation between RPM and velocity, we build on this modeling theory. For this purpose, we introduce splines as a way of achieving higher accuracy in RPM prediction based on velocity history and we apply clustering algorithms on the vessel's trajectory data in order to find sub-trajectories that share similar velocity values.

For predicting *RPM* values from velocity (V) measurements using splines, we suggest to cluster the training space to regions with *similar* velocity patterns. Regions must have similar N previous values of V, on the basis that including velocity at N previous time steps, as an extra feature in the training phase, will lead to a higher accuracy when predicting *RPM* for time $t = t_{i+1}$ as shown in Section 3. Therefore, each cluster will represent subsets of similar distributions, in terms of standard deviation and mean value, with respect to the history of a value V_i at a given time t_i during a route. All velocity instances V that have similar N previous values are grouped in the same cluster in order to build and train different models that represent different distributions. The training data consists of a 2D vector of the form: $(V(t_i), \bar{V}_N(t_i))^T$ with $\bar{V}_N(t_i) = mean[V(t_{i-N}, ..., V(t_i)]$, and the corresponding $RPM(t)$ value.

At the final stage of the evaluation the problem converts to a problem of classification. Each instance $V(t_i)$ of the test (unseen) dataset must be classified to the most similar $cluster_{k(i)}$. From this point and on we predict the corresponding $RPM(t_i)$ value with the specified $model_{k(i)}$ trained on this particular group of data.

The idea behind the proposed piece-wise regression by clustering is that, as previously stated, the relationship between RPM and V is not linear at all times. As Figures 2 and 1 indicate there exists a strong linear relationship between the dependent (RPM) and independent (V) variables, nevertheless there are parts exhibiting a higher-order (non-linear) correlation. This remark guides us to the choice of building different models, each one corresponding to a different part of relation between our variables (RPM,V), in order to improve the overall accuracy.

Given a sample dataset comprising tuples of the form $\{(x_i, y_i), ..., (x_m, y_m)\}$, where $x_i := V(t_i)$ and $y_i := RPM(t_i)$, we assume that the relation between V and RPM, is described by a polynomial regression model of the form:

$$F(x_i) = b_0 + b_1 x_i + b_2 x_i^2 + \ldots + b_n x_i^n + \epsilon_i,\ i = 1, \ldots m$$
$$, or\ in\ matrix\ form:\ F = A(x_1, ..., x_m)b + \epsilon, \quad (7)$$

where $b = (b_0, b_1, ..., b_n)^T$ is the parameter vector, A is a Vandermonde matrix, also referred as the design matrix, and ϵ_i is the error vector. The problem of building K regression models in K different clusters $\mathcal{X}_1, \mathcal{X}_2, ..., \mathcal{X}_K$ can be formalized with the aid of (4.2) as follows:

$$F(x) = \begin{cases} A_1(x_{11}, ..., x_{1m_1})b_1, & x_1 = (x_{11}, ..., x_{1m_1})^T \in \mathcal{X}_1, \\ \vdots & \\ A_k(x_{K1} \ldots x_{Km_K})b_K, & x_K = (x_{K1}, ..., x_{Km_K})^T \in \mathcal{X}_k. \end{cases} \quad (8)$$

When choosing K one must take into account the trade-off between fitting the data and avoiding model complexity and overfitting, which may result in poor generalization on unseen data. This is related to one of the most crucial aspects in function learning, known as the trade-off between bias and variance. The value of K can be chosen through cross-validation, with a possible upper-bound dictated by the maximum tolerable complexity of the estimated model. Clustering the data-set and choosing the optimal K plays a crucial role in piece-wise regression analysis as the results of our experiments (see Section 5) point out.

A draft sketch of the algorithm that performs regression on different sections (clusters) of velocity values to predict RPM follows.

The algorithm begins with the set of (velocity, RPM) pairs D which is clustered into k clusters in a way that optimizes the bias-vs-variance trade-off. Then each instance $V(t_i)$ is classified to the "best" cluster D_b in terms of fitness (using the normalized distance d_{ij} from the centroid $\mathbf{C}_j$ of each cluster. The model that has been trained by the cluster that has minimum distance is used to predict the corresponding $RPM(t_i)$ value.

Algorithm 1 Piecewise regression algorithm with clustered data

Require: $D = \{(v_1, r_1), ..., (v_m, r_m)\}$: $v_i = V(t_i)$, $r_1 = RPM(t_i)$
Split D into k clusters $D_1, ..., D_k$
foreach $D_j, j \in [1, k]$ **do**
$M_j = train\_regression\_model(D_j)$
end
foreach $V(t_i)$ **do**
  foreach $D_j, j \in [1, k]$ **do**
    $C_j = centroid(D_j)$
    $d_{ij} = \frac{1}{1+\sqrt{\|(V(t_i), \bar{V}_N(t_i))^T - \mathbf{C}_j)\|^2}}$
    $\bar{V}_N(t_i) = mean[V(t_{i-N}, ..., V(t_i)]$
    $D_b = \arg\min(d, D_j)$
  end
end
$RPM(t_i) = M_j(V(t_i))$

5 EXPERIMENTAL EVALUATION

The aim of the experimental evaluation process is to test the applicability and the performance of the proposed methodology in predicting RPM from previous observations of the velocity (V). More specifically, the questions we examine within this experimental section are the following: **(a)** Does the clustering/partitioning of the input space when combined with models trained separately for each cluster affect the prediction performance? **(b)** Given a dataset containing (RMP, V) observations, is there an optimal number of clusters that maximizes prediction performance? **(c)** How does the number of clusters relate to the expected performance? **(d)** Do spline regression performs better than other established baselines? **(e)** How does the combination of spline regression and clustering of the input space perform?

In order to preserve the statistical independence of our results between different datasets, in all the experiments that follow we apply the two-sample Kolmogorov-Smirnov (K-S) test. The K-S test is a non-parametric test of the equality of continuous (or discontinuous), one-dimensional probability distributions that is used to compare one or more samples with a reference probability distribution. The size of train and test subsets for the experiments presented below is set to approximately $4 * 10^3$ and $3 * 10^3$ observations , respectively.

5.1 Regression methods

Apart from **Spline Regression** (Section 4.1), in our experiments we evaluated three more regression techniques namely Linear Regression, Random Forest Regression and Neural Networks as follows:

Linear regression is a classic regression technique, which models the output variables as linear combinations of the input variables. The regression coefficients of the input variables are usually estimated using least-squares error or least absolute-error approaches and the optimization problem is solved efficiently using either quadratic-programming or linear-programming. In order to accommodate non-linearity, when it exists, polynomial regression is an alternative to linear regression analysis.

Random-Forest regression is an ensemble technique used for classification and regression. It starts with constructing a set of decision trees at training time and then outputs the majority output value (in classification tasks) or the mean output value (in regression tasks) of all individual trees. The randomness principle is either covered by choosing a random subset of features or by choosing a random subset of observations to train each individual tree.

Neural Networks is another popular technique for regression and classification tasks. Using Python's Keras framework we defined a Neural Network with one input and four hidden layers, each one consisting of 10 neurons and one output layer. We used rectified linear unit RelU as the activation function of each layer. RelU is

defined as $y(x) = max(0, x)$, and is a function that –in contrast to other activation functions– back-propagates the larger percent of the error on the output to update the neuron weights. A stochastic gradient descent process, the AdaGrad-optimizer of the Keras framework- has been used to find the optimal set of weights for the neural network. Each optimization run for 10 epochs (full training cycles on the training set).

5.2 Clustering methods

The clustering techniques used in our experiments are K-means and a triangulation-based clustering algorithm and are briefly explained in the following. The techniques have been tested Î£n datasets of size 10^q ($q = 3, 4, 5$).

K-means clustering is a vector quantization method, with origins from the field of signal processing, that is widely used for data clustering. Its main aim is to partition the observations (vectors) into K clusters, so that each observation belongs to the cluster with the nearest centroid (representative vector of the cluster). As a result, the data space is partitioned into Voronoi cells.

Triangulation clustering (DC) [4] first partitions the training space in triangles using a triangulation-based method. Delaunay Triangulation (DT) was used in our experiments due to the fact that is intrinsically related to the Voronoi diagram being actually its dual graph. Another reason for opting in favour of DT among other triangulation techniques is its close connection with the so-called *Delaunay Configurations* that, as stated in [19], is closely related with a multivariate extension of the univariate B-Splines used in this paper for approximation.

By selecting a cut-off value p(a value that is used to determine the neighboring points from the adjacency list of each candidate vector $[\mathbf{V}(t_i), \bar{\mathbf{V}}_N(t_i)]\}$) we can find for each point in the training space its neighboring vertices in the resulting graph. By applying a Depth-First-Search (DFS) algorithm it is possible to find isolated subgraph components recursively as depicted in Figure 4, which shows the resulting clusters for the pointset in Fig. 3.

The basic idea behind clustering with triangulation is that it defines the cluster in a much broader manner, than, e.g., K-means, being able to cluster observations in non-spherical neighborhoods. Also K-means, in its general definition used here, doesn't seem to to detect outliers .In contrast with K-means DT based clustering, as depicted in 4 is able to detect and remove outliers from clusters resulting in more "reliable" clusters. Further research for improving this method has to focus on the search of optimal cut-off value p. Both clustering algorithms showed promising performance especially in conjunction with linear and spline regression, respectively.

5.3 The effect of clustering

Initial experiments were conducted for the previously described clustering methods with constant training size of approximately $3 * 10^3$ instances. Specifically, we utilized the algorithm proposed in Section 4.1 for the aforementioned regression methods, for different values of k (clusters) (for k-means clustering) and different cut-off values P (for the triangulation-based clustering). Indicative results are collected in Figures 5 and 6.

Table 2 summarizes the results of the experimental evaluation on five, statistically independent, samples of size $3 * 10^3$ instances using different combinations of clustering (K-means, Delaunay Triangulation (DT)) and regression (linear LR, splines-based SR, random forests RF and neural networks NN). The table reports the covariance of the input variables and the Mean Average Error (MAE) of the predicted values. Results show that *RF* with K-means and *Splines* with *DT* clustering have the best accuracy. However, the optimal number of clusters varies, depending on the time instance that the training sample was drown and therefore its distribution. The *DT*-based methods perform better with the splines model (*SR*) instead of *LR* and in some cases the overall accuracy achieved by the *SR*/*DT* combination is higher than that achieved by *LR* or *RF* combined with any of the clustering methods. Also, the DT clustering method produces better space partitioning than K-means when Spline regression is going to be used for RPM prediction. The results of our experiments are aligned with the theory that K-means is locally isotropic in contrast to *DT* clustering that is moving in the search space for finding neighboring points by using the weighted edges of the Delaunay Triangulation. On the other hand Neural Networks do not seem to work well after clustering as the results of Table 2 indicate.

Experimental Results				
Algorithm	variance	clusterer	MAE	opt #clusters if ≠ 1
SR	63.892	K-means	1.595	19
SR	66.497	K-means	2.527	6
SR	64.693	K-means	1.880	26
SR	63.892	DC	1.405	19
SR	66.497	DC	1.527	6
LR	63.892	K-means	1.58	19
LR	66.497	–	2.474	1
LR	64.693	K-means	1.761	30
LR	63.892	DC	1.580	19
LR	66.497	–	2.474	1
LR	64.693	–	2.340	1
RF	63.892	K-means	1.550	19
RF	66.497	K-means	2.055	5
RF	64.693	–	21.54	1
RF	63.892	DC	1.550	19
RF	66.497	DC	2.202	5
RF	64.693	–	1.880	1
NN	63.892	–	2.2021	1
NN	66.497	K-means	2.055	5
NN	64.693	–	2.141	1
NN	63.892	DC	12.345	19
NN	66.497	DC	9.234	5
NN	64.693	–	4.412	1

Table 2: Results of the experimental evaluation.

The experimental results in Figures 5 and 6 and Table 2 and further results for five different statistically independent subsets of approximately $4 * 10^3$ observations are summarized in Figure 7 that illustrates the mean difference (in error rate) between clustered and non-clustered data for different regression methods.

Results show that clustering improves the regression algorithms performance especially concerning the first three algorithms (i.e. LR, RF, SR). On the other hand, the NN algorithm has worse performance when combined with clustering, which is also obvious from the last rows of Table 2. Another outcome is that Spline regression (SR) exhibits the largest improvement in terms of prediction error compared to the other three regression methods, when we compare performance between the application in the original and the clustered dataset. This result must be further examined in order to search for a connection between the knots of the spline estimator and the clustered input values. Finally, based on the experimental

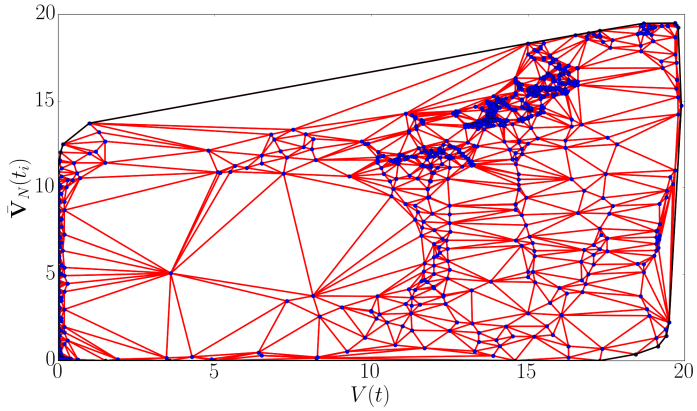

Figure 3: Convex hull and Delaunay Triangulation of a planar training pointset $\{[\mathbf{V}(t_i), \bar{\mathbf{V}}_N(t_i)]\}_{i=1}^{10^3}$

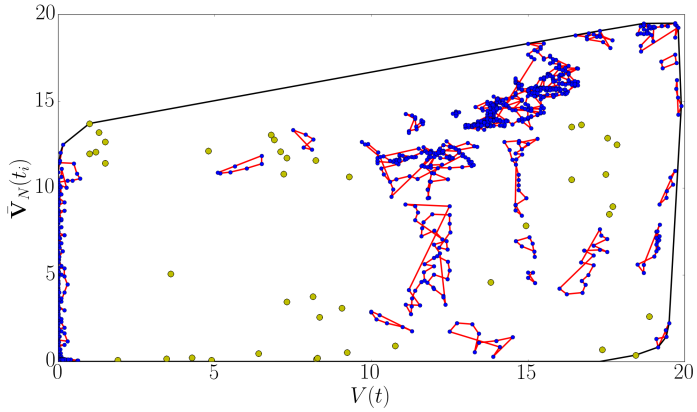

Figure 4: Clustering outcome after applying DFS on the DT of Fig. 3; points belonging to the same cluster are connected with red linear segments.Green points indicate outliers.

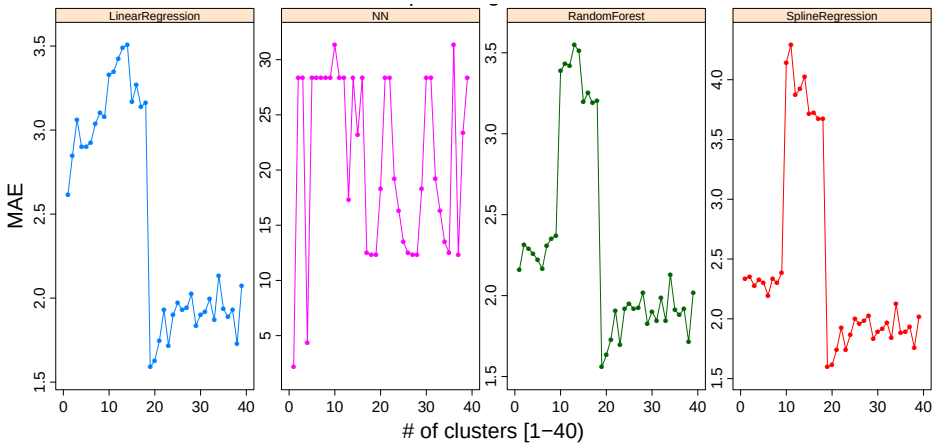

Figure 5: Error convergence with K-means for varying k values

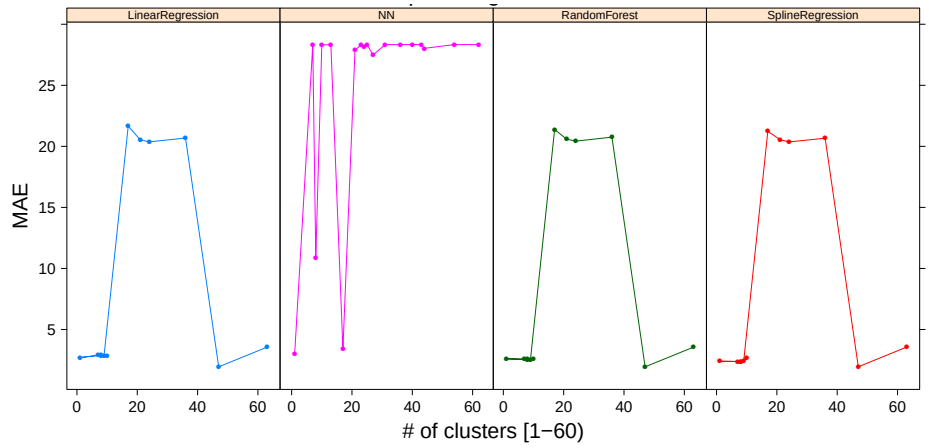

Figure 6: Error convergence with DT clustering for a varying number of clusters.

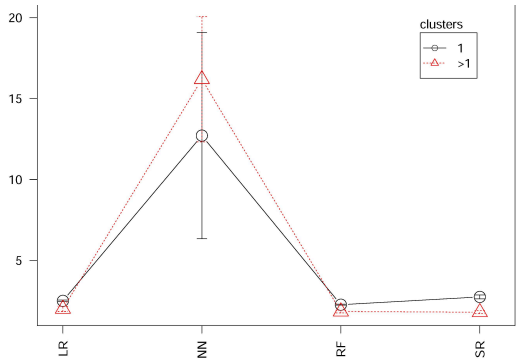

Figure 7: Mean error rate difference of regression models (clustered vs non-clustered samples)

results we can conclude that for all 3 regression algorithms there exists an optimal number of clusters for which they achieve the highest accuracy.

5.4 Finding the optimal number of clusters

The number of clusters of the input variables is proven to be a critical parameter that affects the accuracy of the regression algorithms. In order to statistically prove that the number of clusters plays a significant role in the process of estimating RPM, we perform the Kruskal-Wallis statistical test [16]. This test is a non-parametric approach to the one-way Analysis of Variance (ANOVA) and is used to compare three or more groups on a dependent variable that is measured on at least an ordinal level. The significant result in a Kruskal-Wallis test indicates that there are group differences, but needs a post-hoc procedure to determine which groups are significantly different from each other.

The Kruskal-Wallis test in our case examines the statistical significance between groups of initial parameters, and more specifically between: i) the number of clusters, ii) the clustering method, and iii) the Regression method. The null-hypothesis tested is that there are no significant differences between our groups of features that affect the error rate.

The results indicate that we can accept the null hypothesis for the regression methods (p-value>0.05), while we can reject it for the rest of the groups (clustering method, number of clusters), because their p-value is smaller than the predefined threshold (0.05). As a consequence we can safely claim that the clustering method and the number of clusters have a significant impact on the error rate.

The above results justify the initial idea of applying piece-wise regression on clusters of input values and are indicative of an underlying strong relationship between clustering and regression analysis that must be further examined. They also show that future work must examine many more hyper-parameters and their impact on RPM estimation. For example, the number N of previous time steps involved for building the training vector $(\mathbf{V}(t_i), \bar{\mathbf{V}}_N(t_i))^T$, the variance of the training set, the order and smoothness of the basis functions used in the adopted splines, etc.

5.5 Splines Regression vs other regression methods

The experiments so far showed that clustering of the input space results to higher accuracy for at least 3 out of the 4 proposed regression models. The aim of the next experiment is to test whether the combination of Linear Mixed Models and Spline Regression, as presented in Section 4.1, stands in practice, i.e., test whether, under some constraints, splines perform better than the other two regression methods. As a first step towards this direction, Table 3 presents the results for the optimal number of clusters between 10 statistically independent subsets. These results are associated with the red-dotted line of Figure 7, however, they provide more analytic information.

The results of Table 3 are depicted in the boxplot of Figure 8 below, where the performance of each regression method is compared in terms of absolute value and standard deviation from the median. We easily observe in this plot that Splines (*SR*) and Random Forest (*RF*) perform better than Linear Regression (*LR*) both in terms of accuracy and variance. As far as the comparison between *SR* and

sample	SR	LR	RF
1	1.595/ ***19***	1.588 / ***19***	1.557 /***19***
2	2.027/***6***	2.794 /***2***	2.255 / ***5***
3	2.075 /***2***	2.903/***2***	2.589/***3***
4	1.889 /***26***	1.761/***30***	1.831 / ***39***
5	2.013 / ***31***	2.696/***2***	2.381/***4***
6	2.034 / ***27***	1.5844/***17***	1.667 / ***27***
7	1.4056 / ***23***	2.08 /41	1.574/***23***
8	1.411 /***22***	1.8843 /***52***	1.623/***17***
9	1.436 /***28***	1.588/***27***	1.856 /***44***
10	1.573 /***44***	1.582 /***57***	1.937 /***21***

Table 3: MAE / optimal # of clusters of the three top, in terms of performance, regression methods for the optimal number of clusters being > 1.

RF is concerned, while *SR* appears to perform better than *RF*, the latter exhibits lower variance in error rate than that by Splines.

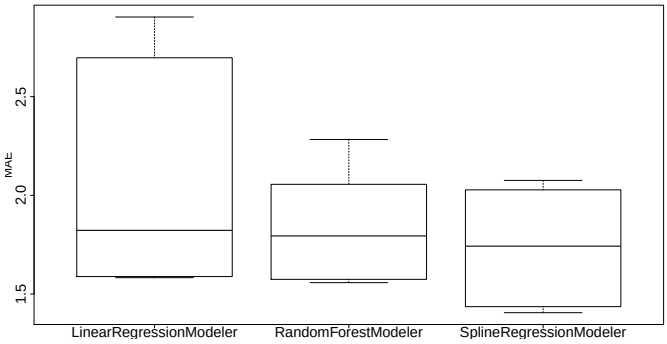

Figure 8: BoxPlot of regression methods compared to error rate

In order to determine which regression method performs better, we apply statistical testing with the results from the Table 3 . Because of our relatively small sample size of 10 samples < 30, we assume non-normality to our dependent variable (the error measured) so we decide to conduct a Wilcoxon signed-rank test, which is the non parametric equivalent of a Paired T-test. Both test are used extensively to compare two groups of dependent (i.e., paired) quantitative data. Wilcoxon can be used in order to determine which algorithm is significantly different than others in terms of accuracy. The null hypothesis tested here is that the true mean error difference between the two regression methods evaluated each time is greater than zero.

The results from the three separate Wilcoxon paired ranked tests indicate that: a) Comparing *RF* with *LR* we get a p-value of $\simeq 0.18 > 0.05$, meaning we can accept the null hypothesis stating that the true mean error difference between the two pairs tested is greater than zero and conclude that *LR* performs better than *RF*. b) Comparing *SR* with *LR* we get a p-value $\simeq 0.03$, which is less than 0.05. Thus, we can reject the null hypothesis. This means that with a confidence level around 95% *SR* performs better than *LR*. c) The same can be stated also for *SR* and *RF*, as the p-value $\simeq 0.04$, is close but below the predefined threshold of 0.05 indicating that *SR* performs significantly better than *RF*.

From the statistical tests conducted above, we can conclude that, while it is safe to assume that Splines perform better, with statistical significance, than the other regression methods, the overall performance from the regressors, except NN, combined with clustering was relatively good. In light of these findings one of the next steps of our work would be to study further Spline approximation theory and how clustering affects their accuracy but also to conduct experiments of larger scale with RF and LR.

5.6 Combining splines with clustering

Our experiments so far showed that a strong connection exists between partitioning the input space and the performance of spline regression. Fig. 9 attempts to validate this statement by depicting the mean error difference between the two clustering techniques (K-means and DT-based clustering) for the optimal number of clusters against 5 statistically independent samples consisting of $\approx 4 * 10^2$ observations.

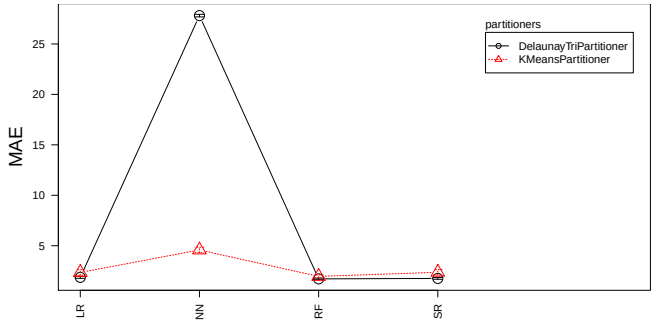

Figure 9: Plot of mean error difference between the two clustering methods for the optimal number of clusters.

More specifically, Figure 9 shows that, while both clustering techniques perform well, DT clustering seems to perform better when it is combined with 3 of the 4 regressors (except from neural networks). Looking at the plot we can also state that spline regression combined with *DT* clustering presented marginally the largest improvement in terms of accuracy compared to the other two regression techniques. This experimental result agrees with pertinent literature, which states a connection between Delaunay partitions and polynomial splines; see Section 4.

6 CONCLUSIONS AND NEXT STEPS

The motivation problem of our work is that of vessel optimal routing by minimizing its FOC (Fuel-Oil Consumption). Via reviewing the pertinent literature on the subject and conducting initial experimentation we concluded that the problem could be handled efficiently if a good predictive model for the *RPM* (revolutionary speed) of the main engine of a vessel moving with known speed Vwere available. Furthermore, access to real industrial data, taken from measurements on-board ships, indicate a strong correlation between *RPM* and V on specific time instances during a voyage while others suggested a non-linear relationship between them.

On the basis of the above, we have been led to the idea of developing an *RPM* predictive model that separates the domain in

correlated subdomains with respect to velocity V. In this connection, we opted for Spline regression (SR) in order to approximate the underlying function $RPM(V)$ on each subdomain as splines are by their nature continuous piecewise polynomials appropriate for approximation in partitioned domains. The regressor team also included Linear regression (LR), Random Forest (RF) and a baseline Neural Network (NN).

Summarising the results of the approach assembled so far, it seems that spline (SR) and RF (Random-Forrest) regression alongside with partitioning the input space either with K-means or DT-based (Delaunauy Triangulation) algorithm perform better and tend to achieve higher accuracy compared to LR NN. It is also worthnoticing that by enhancing our feature vector with the mean of velocity at N previous time - steps we managed to improve further the accuracy of our predictive scheme.

Besides expanding the scale and variability of our experiments, our short-term future objectives will focus on investigating the effect of several hyper-parameters related to clustering and Spline regression model such as: (i) the optimal cut off value for the DT (triangulation) clustering algorithm and generally the optimal number of clusters for either of the two proposed clustering methods, (ii) the distance metric used for in this paper only Euclidean distance has been tested, (iii) the population and the exact placement of the knots used to approximate the underlying function on each partition and (iv) the order of the spline estimator used to interpolate the data on each partition.

Also further research as far as the tuning process of the hyper-parameter N that control the previous time steps needs to be conducted. Finally, another issue that can be investigated is the practical utilization of the time instance t_i that samples were drawn. A way to achieve this is to include weather conditions at time t_i in our feature space. Wind speed-direction, swell, wave height etc, are some of the parameters that can easily be fed to our existing models or to larger scale - to be built - models like NN that incorporate the existing setting of our proposed algorithm, in order to achieve higher accuracy.

7 ACKNOWLEDGEMENTS

The first author is financially supported by Stavros Niarchos foundation (SNF) . Access to industrial data has been provided by Danaos Shipping Co.

REFERENCES

[1] K. Amirnekooei, M. M. Ardehali, and A. Sadri. 2012. Integrated resource planning for Iran: Development of reference energy system, forecast, and long-term energy-environment plan. *Energy* 46 (2012), 374–385.
[2] B. Can and C. Heavey. 2012. A comparison of genetic programming and artificial neural networks in metamodeling of discrete-event simulation models. *Computers Operations Research* 39 (2012), 424–436.
[3] Andrea Coraddu, Luca Oneto, Francesco Baldi, and Davide Anguita. 2017. Vessels fuel consumption forecast and trim optimisation: a data analytics perspective. *Ocean Engineering* 130 (2017), 351–370.
[4] C Eldershaw and Markus Hegland. 1997. Cluster analysis using triangulation. *Computational Techniques and Applications* (1997), 201–208.
[5] John Fox. 2015. *Applied regression analysis and generalized linear models*. Sage Publications.
[6] Mihalis M Golias, Georgios K Saharidis, Maria Boile, Sotirios Theofanis, and Marianthi G Ierapetritou. 2009. The berth allocation problem: Optimizing vessel arrival time. *Maritime Economics & Logistics* 11, 4 (2009), 358–377.
[7] H Hagiwara and JA Spaans. 1987. Practical weather routing of sail-assisted motor vessels. *The Journal of Navigation* 40, 1 (1987), 96–119.
[8] CR Henderson. 1974. General flexibility of linear model techniques for sire evaluation. *Journal of Dairy Science* 57, 8 (1974), 963–972.
[9] Charles R Henderson. 1975. Best linear unbiased estimation and prediction under a selection model. *Biometrics* (1975), 423–447.
[10] S. A. Kalogirou and M. Bojic. 2000. Artificial neural-networks for the prediction of the energy consumption of a passive solar-building. *Energy* 25 (2000), 479–91.
[11] Neungrit P. Kengpol A. 2014. A decision support methodology with risk assessment on prediction of terrorism insurgency distribution range radius and elapsing time: An empirical case study in Thailand. *Computers Industrial Engineering* (2014), 55–67.
[12] Boram Kim and Tae-Wan Kim. 2017. Weather routing for offshore transportation using genetic algorithm. *Applied Ocean Research* 63 (2017), 262–275.
[13] G.S. Kimeldorf and G. Wahba. 1971. Some Results on Tchebycheffian Spline Functions. *J. Math. Anal. Appl.* 33 (1971), 82–94.
[14] George S Kimeldorf and Grace Wahba. 1970. Spline functions and stochastic processes. *Sankhyā: The Indian Journal of Statistics, Series A* (1970), 173–180.
[15] Odysseas Kosmas and Dimitrios Vlachos. 2012. Simulated annealing for optimal ship routing. *Computers & Operations Research* 39, 3 (2012), 576–581.
[16] William H Kruskal and W Allen Wallis. 1952. Use of ranks in one-criterion variance analysis. *Journal of the American statistical Association* 47, 260 (1952), 583–621.
[17] Edward V Lewis. 1988. Principles of naval architecture, Second revision. *Jersey: SNAME* 2 (1988).
[18] P McCullagh and J. A. Nelder. 1989. *Generalized Linear Models*. Vol. 2nd ed. Chapman and Hall/CRC Press.
[19] Marian Neamtu. 2007. Delaunay configurations and multivariate splines: A generalization of a result of BN Delaunay. *Trans. Amer. Math. Soc.* 359, 7 (2007), 2993–3004.
[20] B. Ozer and E. Gorgun. 2013. ÎŹncecik, S., The scenario analysis on CO2 emission mitigation potential in the Turkish electricity sector: 2006–2030. *Energy* 49 (2013), 395–403.
[21] Adnan Parlak, Yasar Islamoglu, Halit Yasar, and Aysun Egrisogut. 2006. Application of artificial neural network to predict specific fuel consumption and exhaust temperature for a diesel engine. *Applied Thermal Engineering* 26, 8-9 (2006), 824–828.
[22] Kalogirou Sa. 2000. Applications of artificial neural-networks for energy systems. *Applied Energy* 67 (2000), 17–35.
[23] R Savitha, Abdullah Al Mamun, et al. 2017. Regional ocean wave height prediction using sequential learning neural networks. *Ocean Engineering* 129 (2017), 605–612.
[24] Wei Shao, Peilin Zhou, and Sew Kait Thong. 2012. Development of a novel forward dynamic programming method for weather routing. *Journal of marine science and technology* 17, 2 (2012), 239–251.
[25] George Stergiopoulos, Evangelos Valvis, Dimitris Mitrodimas, Dimitrios Lekkas, and Dimitris Gritzalis. 2018. Analyzing Congestion Interdependencies of Ports and Container Ship Routes in the Maritime Network Infrastructure. *IEEE Access* 6 (2018), 63823–63832.
[26] Rafal Szlapczynski and Joanna Szlapczynska. 2012. On evolutionary computing in multi-ship trajectory planning. *Applied Intelligence* 37, 2 (2012), 155–174.
[27] N. K. Togun and S. Baysec. 2010. Prediction of torque and specific fuel consumption of a gasoline engine by using artificial neural Networks. *Applied Energy* 87 (2010), 349–355.
[28] Athanasios K Tsadiras, CT Papadopoulos, and Michael EJ OăĂŹKelly. 2013. An artificial neural network based decision support system for solving the buffer allocation problem in reliable production lines. *Computers & industrial engineering* 66, 4 (2013), 1150–1162.
[29] DS Vlachos. 2004. Optimal ship routing based on wind and wave forecasts. *Applied Numerical Analysis and Computational Mathematics* 1, 2 (2004), 547–551.
[30] Conor Walsh and Alice Bows. 2012. Size matters: exploring the importance of vessel characteristics to inform estimates of shipping emissions. *Applied Energy* 98 (2012), 128–137.
[31] Laura Walther, Anisa Rizvanolli, Mareike Wendebourg, and Carlos Jahn. 2016. Modeling and optimization algorithms in ship weather routing. *International Journal of e-Navigation and Maritime Economy* 4 (2016), 31–45.