Electronic Research Archive

ERA, 31(8): 4753–4772.
DOI: 10.3934/era.2023243
Received date: 20 March 2023
Revised date: 05 June 2023
Accepted date: 20 June 2023
Published date: 06 July 2023

http://www.aimspress.com/journal/ERA

Research article

A fair evaluation of the potential of machine learning in maritime transportation

Xi Luo[1,†], Ran Yan[2,*,†], Shuaian Wang[1,†] and Lu Zhen[3,†]

[1] Department of Logistics and Maritime Studies, Faculty of Business, The Hong Kong Polytechnic University, Hung Hom, Hong Kong
[2] School of Civil and Environmental Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore
[3] School of Management, Shanghai University, Shanghai 200444, China

* **Correspondence:** Email: angel-ran.yan@connect.polyu.hk.

† All authors contributed equally and are co-first authors.

Abstract: Machine learning (ML) techniques are extensively applied to practical maritime transportation issues. Due to the difficulty and high cost of collecting large volumes of data in the maritime industry, in many maritime studies, ML models are trained with small training datasets. The relative predictive performances of these trained ML models are then compared with each other and with the conventional model using the same test set. The ML model that performs the best out of the ML models and better than the conventional model on the test set is regarded as the most effective in terms of this prediction task. However, in scenarios with small datasets, this common process may lead to an unfair comparison between the ML and the conventional model. Therefore, we propose a novel process to fairly compare multiple ML models and the conventional model. We first select the best ML model in terms of predictive performance for the validation set. Then, we combine the training and the validation sets to retrain the best ML model and compare it with the conventional model on the same test set. Based on historical port state control (PSC) inspection data, we examine both the common process and the novel process in terms of their ability to fairly compare ML models and the conventional model. The results show that the novel process is more effective at fairly comparing the ML models with the conventional model on different test sets. Therefore, the novel process enables a fair assessment of ML models' ability to predict key performance indicators in the context of limited data availability in the maritime industry, such as predicting the ship fuel consumption and port traffic

volume, thereby enhancing their reliability for real-world applications.

Keywords: fair evaluation of prediction models; machine learning; small dataset; maritime transportation

1. Introduction

ML has advanced significantly in recent years, and the application of this data-driven method has been extended to various fields, such as computer vision, smart cities, biometrics and agriculture (see [1–4]). ML has recently received much attention in terms of solving maritime transportation problems [5,6]. ML techniques have been applied to ship energy efficiency management [5,7], ship risk prediction and safety management [8,9], ship inspection planning [10,11], ocean freight market condition prediction [12,13] and other areas.

ML is a branch of artificial intelligence in which computers learn by improving their task performance through fitting to historical data rather than being literally programmed [2,14]. The ML algorithm is generally trained on a proportion of the historical data and the remainder of these data are regarded as the test set, which validates the generalization ability of the trained model. The construction of ML models usually requires a large volume of data, so decisions concerning how many data samples to use have a significant impact on their performance. Faber et al. [15] report that the accuracy of an ML model in predicting the energy required for the formation of Elpasolite crystals improves significantly as the training data size increases. Ng et al. [16] find that the prediction accuracy of three ML models in terms of soil properties improves as the training sample size increases by examining the learning curves. These studies unambiguously demonstrate that the predictive performance of ML models can be improved by increasing the size of the training dataset. However, the improvement brought about by data accumulation is likely to be limited by the noise and errors in the training data. The amount of data required for training ML models depends on many factors, such as the complexity of the ML algorithm, the expected prediction accuracy and the number of features considered. In terms of complexity, the more complex the ML model, the more training data are required to train the algorithm and thus guarantee the generalization ability of the prediction model.

In practice, learning curves are widely applied to identify the extent to which an ML model benefits from adding more training data and whether the model suffers from underfitting or overfitting. A learning curve graph typically displays two curves: one depicting the relationship between the number of training samples and the predictive accuracy on the training set and the other depicting the relationship between the number of training samples and the predictive accuracy on the validation set. When the prediction accuracies for both the training and validation datasets converge to a value as the training data size increases, the size corresponding to the value represents the minimum requirement for training an ML model to obtain satisfactory performance. Thus, a training dataset with a size below the minimum is too small to guarantee the satisfactory performance of the model. In addition, some technologies, such as semi-supervised learning [17,18], have been developed to train ML models on relatively small training dataset. For example, Wu and Prasad [19] propose a semi-supervised deep learning approach for hyperspectral image classification, which uses both limited labeled data and abundant unlabeled data to train a deep neural network.

Although the ML models emerging in recent years demonstrate competitive performance in multiple prediction tasks, conventional models which are usually based on expert knowledge or simple

rules such as linear regression (LR) also play an important role in the maritime industry. To determine whether ML models or the conventional model is more accurate in terms of a prediction task, researchers usually train multiple ML models and the conventional model on the same training set simultaneously. If one ML model is more accurate than the conventional model on the test set, we can conclude that ML methods can better fit the training data and offer a higher prediction accuracy for unseen data than the conventional model [20]. The above comparison process is referred to as the common process in this study. For example, in ship fuel consumption prediction research, statistical regression methods are regarded as the conventional approach for ship speed–fuel consumption modeling (see [21–23]). Gkerekos et al. [24] collect 745 ship sailing records from a noon report and compare LR methods and several ML algorithms such as random forest (RF), the extra tree model, the support vector machine (SVM), the boosting model, the bagging model and artificial neural networks (ANNs) in terms of their abilities to predict ship fuel consumption. They conclude that SVM is the most accurate according to the coefficient of determination (R^2) on the same test set. Uyanık et al. [25] also compare the performance of ML and statistical regression models in terms of predicting ship fuel consumption using 724 samples from the noon report, and find that deep neural networks perform best on the test data. Li et al. [26] combine the noon report data and meteorological data according to the estimated geographical positions derived from the great circle route. Based on the fusion dataset of no more than 500 samples, they conduct a study to evaluate the performance of statistical regression models, including ridge regression and least absolute shrinkage and selection operator (LASSO) regression, and nine ML models, such as RF, SVM, ANNs and extremely randomized trees (ERTs), in predicting ship fuel consumption rate. They conclude that ERTs present the best fit and generalization performances among all models. Different from Li et al. [26], Du et al. [27] combine the noon report data and the meteorological data according to the actual geographical coordinates provided by the AIS data. They also compare the performance of statistical regression models, including ridge regression and LASSO regression, with nine ML models such as RF, SVM, ANNs and ERTs to predict ship fuel consumption rate. Their findings also show that ERTs exhibit the most promising predictive performance among all the models.

Due to the difficulty and huge expense of collecting large quantities of data, most of the aforementioned studies use no more than 1000 records to train and test ML models. However, these training datasets may be inadequate, particularly for ML models with a high degree of complexity. Training ML models using a small quantity of data may lead to two problems. First, inadequate training data not only make pattern recognition more difficult but also reduce the generalization ability of the models [3]. Although ML models perform better than statistical regression models on small test sets, this improvement cannot be guaranteed for a larger and unseen dataset. In addition, with a small test set and the exponential number of trained ML models required to cover the whole hypothesis space, an ML model that can make predictions with 100% accuracy on the test set may exist (i.e., provide the ground truth). For example, if each sample in the test set takes the value of 0 or 1 and the test set contains n samples, then the values of all samples have 2^n cases, and the true value for all samples is one of the 2^n cases. If we have 2^n ML models and each randomly predicts the targets, one of the models must have 100% prediction accuracy. However, this does not indicate that the ML model outperforms the conventional model in terms of predictive ability. Thus, the common process of comparing multiple ML models and the conventional model on the same training set leads to unfair comparisons.

We design a novel process to compare the performance of ML models and the conventional model

for a small dataset. First, we train multiple ML models on the same dataset, and the model with the best performance is selected using the validation set. The best ML model is then retrained on a new training dataset, which is comprised of the original training set and the validation set. This new training dataset is then used to train the conventional model. Finally, the prediction accuracies of the retrained ML model and the conventional model are compared on the same test data. This novel process guarantees a fair comparison between the ML models and the conventional model. It is worth noting that this novel process is used to compare the predictive performance of ML models and conventional models rather than to compare the performance of multiple ML models.

In this study, we regard an ML model as having consistent performance if its superiority over the conventional model is consistent across different test sets. For example, the novel process or the common process can select the best ML model A^* from multiple ML models and A^* then performs better/worse than the conventional model on different test sets in terms of a certain metric. Model A^* thus has consistent performance. To verify that the novel process is more likely to select an ML model with consistent performance, two processes are executed simultaneously many times to calculate the probability that each process selects an ML model with consistent performance. LR is the most basic regression method and ANN is a popular ML method, so numerous ANN models with different structures are randomly generated and two processes are used to select the ANN model with the best predictive performance in each execution. Based on the selection, the predictive performance of the best ANN model is compared twice with that of the LR model on two test sets. When the preset number of executions is reached, we can calculate and compare the probabilities of selecting ML models with consistent performance for the two processes. The results show that the proposed process has a higher probability of selecting the ML model with consistent performance under a small dataset.

The proposed novel process for comparing the performance of ML models and conventional models on small datasets has important implications for the maritime industry. By enabling a fair comparison between ML and conventional models in the context of limited data availability in the maritime industry, the proposed process provides a more accurate assessment of the ability of ML models to predict key performance indicators such as ship fuel consumption. This, in turn, can enhance the reliability and robustness of ML models used in the maritime industry, making them more suitable for real-world applications.

2. Data description

Our data are derived from two sources: the Asia Pacific Computerized Information System (APCIS) operated by the Memorandum of Understanding on Port State Control in the Asia-Pacific (Tokyo MoU)[1] and the World Register of Ships (WRS)[2]. The APCIS is a freely accessible database of historical PSC inspection data from the member authorities of the Tokyo MOU. These data mainly include information on deficiencies and detentions of ships that have been identified during PSC inspections. Many studies use PSC inspection data derived from this database to train different ML models for predicting different types of ship risk, such as the ship detention probability and the number of ship deficiencies, thereby achieving accurate identification of high-risk foreign visiting ships (see [28–30]). The WRS is a comprehensive database maintained by the International Maritime Organization (IMO) that contains information on ships of all types and sizes around the world [31]. It

[1] https://www.tokyo-mou.org/inspections_detentions/psc_database.php.
[2] https://world-ships.com/.

provides information on ships' ownership, classification, construction and operational status, as well as details of their flag state, port of registry and other relevant information. PSC plays an important role in ensuring maritime safety and control ship pollutants in maritime transportation (see [32–39]).

Table 1. Feature description.

Output target	Description	Min value	Max value	Mean value
The number of deficiencies	This output target represents overall number of ship deficiencies found in the PSC inspection.	0	51	4.10
Input features	**Description**	**Min value**	**Max value**	**Mean value**
Ship age	This parameter is the years between the date of keel laid and the date of PSC inspection.	0.36	48.94	11.63
Ship type	Bulk carrier, general cargo/multiple purpose, container ship, passenger ship, tanker, other.	/	/	/
Ship gross tonnage	This parameter measures the overall internal volume of ships in unit 100 cubic feet.	299.00	228,283.00	44031.12
Ship length	This parameter is the maximum length of a ship in unit of meter.	40.75	400.00	214.44
Ship beam	The parameter is the width of the hull in unit meter.	7.80	63.10	31.84
Ship depth	The parameter measures the perpendicular distance between the top of the keel and the bottom of the upper deck in unit meter.	3.30	38.00	17.73
Total detention times	The parameter presents the sum of the detentions in all historical PSC inspections around the world.	0	18	0.57
Last inspection time	This parameter calculates the number of months between the last PSC inspection and the current PSC inspection.	0.03	180.67	10.01
The number of deficiencies in the last inspection	The parameter presents the ship deficiencies number detected in the previous PSC inspection.	0	55	2.43
State of last inspection	1, if the ship was detained in the last inspection; 0, otherwise; none, if the ship was inspected for the first time in the Tokyo MoU.	/	/	/
The number of times changing the flag	The parameter represents the number of times changing a ship's flag.	0	8	0.68
Ship flag performance	This parameter is provided by the flag Black-Grey-White list [43], whose state can be ranked by "white", "grey", "black". The state of this parameter is "not listed" if the flag is not displayed on the list.	/	/	/
Ship company performance	This parameter is calculated according to historical detention and deficiency records of all vessels in a company's fleet in the last running 36 months [44]. The performance of ship companies can be ranked by "high", "medium", "low" and "very low". The state of "not listed" is used to represent that the performance of the companies is not displayed on the list.	/	/	/
Ship recognized organization (RO) performance	This parameter is calculated according to the RO performance list [45], whose state can be ranked by "high", "medium", "low" and "very low". The state of this parameter is "not listed" if the performance of the RO is not displayed on the list.	/	/	/
Casualties in the last five years	The parameter takes a value of 1 if the ship suffered casualties in the past five years, otherwise it takes a value of 0.	0	3	0.08

Note: The flag Black-Grey-White list and the RO performance list are provided by the Tokyo MoU.

We use the conventional model (LR model) and the ML model (ANN) to make predictions about ship deficiencies. After reviewing the existing literature (see [40–42]), we choose 15 input features from the APCIS database and WRS database that are considered to be useful to predict the number of deficiencies of ships in PSC inspections. The input features from the two databases are combined based on the ship IMO numbers. The selected features are listed and described in Table 1. For new ships appearing in the Tokyo MoU, the values for "the number of deficiencies in the last inspection" and

"last inspection time" are set to the median and the values for "state of last inspection" are set to be the mode. The inspection records period we consider is from January 2, 2015 to November 24, 2020 and we obtain a total of 3672 records.

3. Models for predicting the number of ship deficiencies

3.1. Conventional method: LR model

The LR model is the most common and simplest regression algorithm and is often considered as the baseline model for evaluating the predictive performance of other models. Let $D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n}$ denote the sample data, where $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \cdots, x_{ir})$ represents the ith sample with r features and y_i denotes the output target. The output in the LR model is calculated by a linear combination of the input variables [46,47]:

$$\hat{y}_i = w_0 + w_1 x_{i1} + \cdots + w_r x_{ir} = w_0 + \sum_{j=1}^{r} w_j x_{ij}, \tag{1}$$

where $\hat{y}_i$ is the prediction of sample $\boldsymbol{x}_i$. Parameters $w_j (j = 0,1,\cdots,r)$ in Eq (1) can be estimated by minimizing half of the mean squared error (MSE) shown in Eq (2)

$$\widehat{\boldsymbol{w}} = \underset{\boldsymbol{w}}{\operatorname{argmin}} \left\{ \frac{1}{2n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \right\}. \tag{2}$$

3.2. ML algorithm: ANN model

An artificial neuron is the basic element of an ANN model. It receives one or more inputs from the training data or the previous layer and then processes them via an activation function to produce an output. Figure 1 shows the structure of a neuron. The neuron receives inputs of r dimensions that are associated with weights $w_i (i = 1,2,\cdots,r)$. The weighted sum of all of the inputs denoted by t is calculated by $t = \sum_{i=1}^{n} w_i x_i + b$, where b is the bias. The weighted sum t is passed through an activation function f to generate the output z, which can be represented as $z = f(t)$ (see [47–49]). Three widely used activation functions are rectified linear unit (ReLU), logistic and tanh.

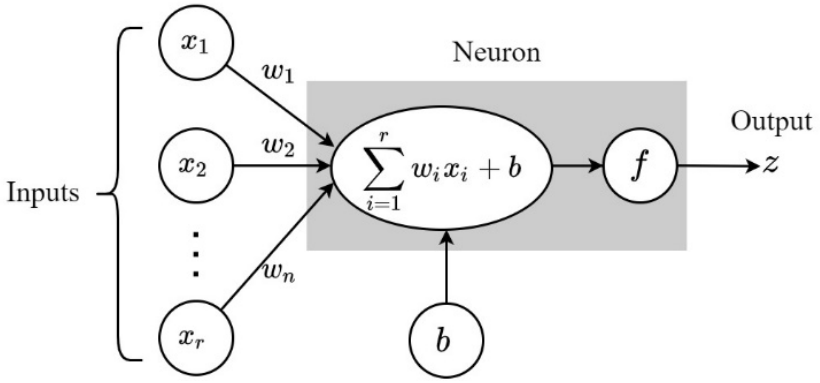


Figure 1. The structure of a neuron.

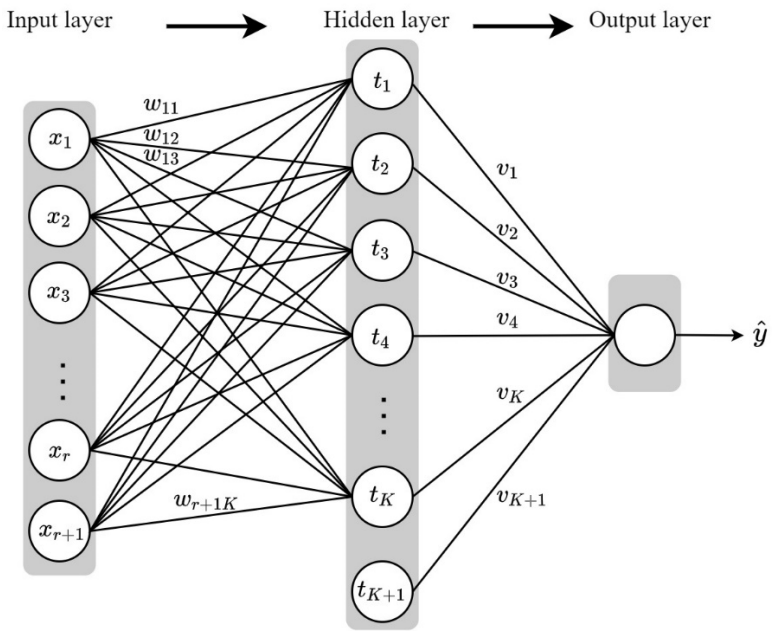


Figure 2. The network structure of an ANN model with three layers.

Figure 2 shows an ANN model with three layers, and its structure can be presented using the following equations. Let $t_k(k = 1,2,\cdots,K)$ denote the weighted sum of the inputs to all neurons in HL. t_k can be calculated by

$$t_k = \sum_{j=1}^{r+1} w_{jk} x_j,\ k = 1,2,\cdots,K. \tag{3}$$

The output of neurons in the HL, denoted by $z_k(k = 1,2,\cdots,K)$, is calculated by

$$z_k = f(t_k),\ k = 1,2,\cdots,K, \tag{4}$$

$$z_{K+1} = 1. \tag{5}$$

$z_k(k = 1,2,\cdots,K)$ and z_{K+1} are then regarded as the input to the output layer. The predicted value generated by the output layer is calculated as follows:

$$\hat{y} = f(\sum_{k=1}^{K+1} v_k z_k). \tag{6}$$

Thus, the relationship between the input features $x_j(j = 1,2,\cdots,m+1)$ and the final prediction $\hat{y}$ can be presented by

$$\hat{y} = f(\sum_{k=1}^{K+1} v_k z_k) = f(\sum_{k=1}^{K} f(t_k) z_k + v_{K+1}) = f\left(\sum_{k=1}^{K} f\left(\sum_{j=1}^{r+1} w_{jk} x_j\right) z_k + v_{K+1}\right). \tag{7}$$

In this regression problem, half of the MSE in Eq (2) is usually typically regarded as the loss function for training an ANN model. The training process of an ANN aims to minimize the loss function by successively adjusting the weights in the model, including $w_{jk}(j = 1,2,\cdots,r+1; k = 1,2,\cdots,K)$ and $v_k(k = 1,2,\cdots,K+1)$, so that the neural network can produce an output that is increasingly similar to the target output. Various algorithms can be adopted to adjust the weights, such as adaptive moment estimation (Adam) [50].

To simulate the scenario of comparing multiple ML models and the conventional model, we generate multiple ANNs with different structures as our ML models, as shown in Algorithm 1. We adopt the ANN models with one hidden layer (HL) while varying the number of neurons in the hidden layer, the activation function and the number of edges connecting the input layer (IL) and HL to create an adequate number of ANN models. The strategy of varying the number of edges involves randomly deleting some that connect the IL to the HL. The ratio of the number of deleted edges to the maximum number of edges between the IL and HL, i.e., the disconnection rate, can be set manually.

Algorithm 1. Algorithm for generating multiple different ANN models.

Algorithm for generating multiple different ANN models

Input

The number of input features: f;
The number of generated ANNs: q;
Candidate set for the number of neurons in the HL: N;
Candidate set for the disconnection rate: R;
Candidate set for the activation function: A;

Output

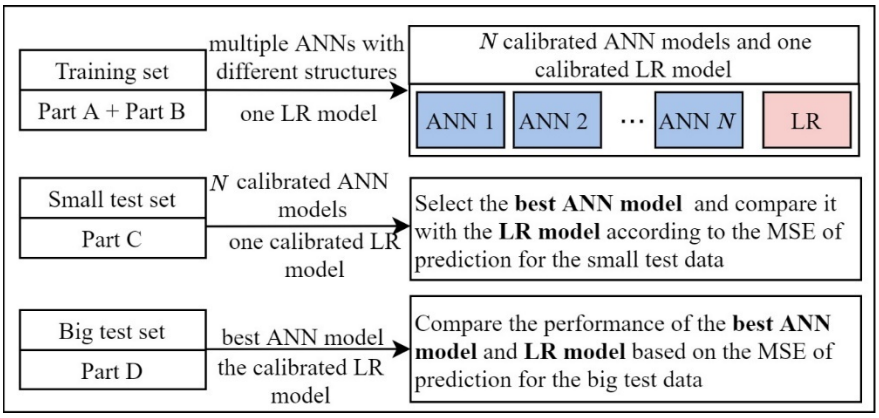
Set of ANNs with different structures: M.

For n in N:
The number of neurons in the HL is set to n;
If the ANN is fully connected, the number of edges between the IL and the HL denoted by N_c will be $f \times n$;
For r in R:
The disconnection rate is set to r;
The deleted number of edges between the IL and the HL is $r \times N_c$;
For a in A:
Among N_c edges connecting the IL and the HL, $r \times N_c$ edges are randomly deleted, obtaining $C_{N_c}^{r \times N_c}$ different ANNs which adopt the same activation function a;
Randomly select $q/\big(size(N) \times size(R) \times size(A)\big)$ ANNs obtained in the last step and add them into the set M.
Return set M.

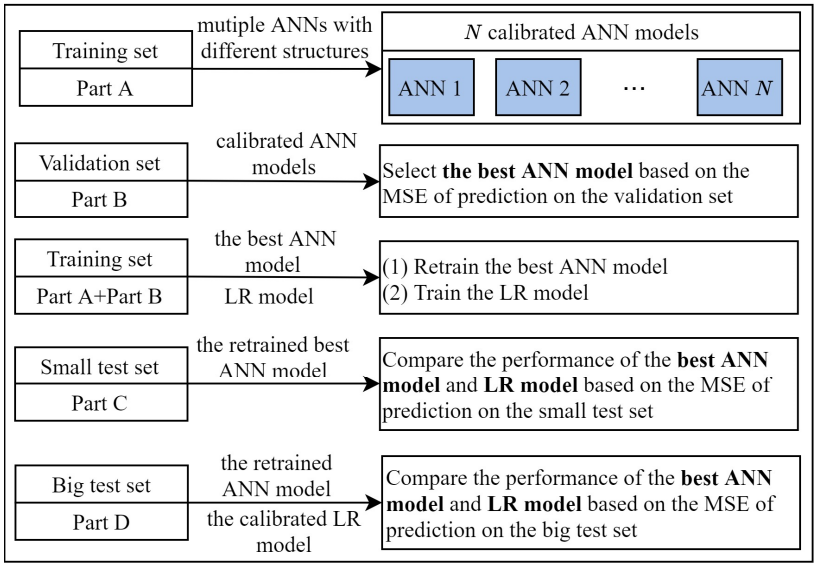
4. Comparisons of the common process and the novel process for model comparison

Figure 3 shows two processes for comparing the performance of ANNs and the LR model, which are executed multiple times to calculate the probability of selecting the ANN model with consistent performance. In a problem with a small dataset, only a small amount of data are available for model training and testing. The final model should be applied to solve practical problems associated with an open dataset which is unforeseeable and is usually significantly larger than the datasets used in the training and testing processes. To simulate such a problem, a larger proportion of data in our dataset is preserved as the unseen data while the remaining smaller proportion is used for model training and testing. Therefore, in each execution, we randomly split the whole dataset into four parts: 300 samples (Part A); 100 samples (Part B); 100 samples (Part C); and 3172 samples (Part D). The combination of Parts A and B is regarded as the training set and Part C as the test set, while Part D is regarded as the

unseen data. The common process and the novel process are then conducted.



(a) The common process



(b) The novel process

Figure 3. Two processes for comparing ANNs and LR models.

In the common process, as shown in Figure 3(a), Parts A and B are first merged into the training set containing 400 samples, which are used to train many ANN models and one LR model. Second, Part C is regarded as the first test set, and the trained ANN models and the LR model can then be utilized to predict outputs in the Part C dataset. By comparing the MSE of all of the ANN models and the LR model on the first test set, we can select the best ANN and evaluate its predictive performance and that of the LR model. We can then determine whether the best ANN performs better than the LR model on the test set. Third, Part D is regarded as the second test set, and the best ANN and the LR model are used to predict the output in the test set. Similarly, we can determine whether the best ANN performs better or worse than the LR model on the current test set.

In the novel process shown in Figure 3(b), Part A is first used to train many ANN models and the

performance of these calibrated ANN models on the validation set, i.e., Part B, is compared to identify the best model. Second, the best ANN model is retrained using the new training set consisting of Parts A and B. The LR model is also trained using this training set. Third, the best ANN model is compared with the LR model on the first test set, i.e., Part C. We can then determine whether the best ANN performs better or worse than the LR model on the test set in terms of MSE. Finally, based on the second test set, i.e., Part D, the same comparison procedure can be conducted between the best ANN and the LR model to evaluate which model has better predictive performance on this test set.

When the preset number of executions is reached, the number of executions for which the best ANN has consistent performance can be obtained for each process, and thus we can calculate its consistency rate denoted by $C_i(i = 1,2)$ ($i = 1$ presents the common process; $i = 2$ presents the novel process), using the following equation:

$$C_i = \frac{B_i+W_i}{M},\ i = 1,2, \tag{8}$$

where M represents the overall number of executions; B_i presents the number of executions that the best ANN model identified using process i performs better than the LR model on two test sets; W_i presents the number of executions that the best ANN model identified using process i performs worse than the LR model on two test sets.

5. Computational experiments

We conduct two processes 1000 times and 1000 ANN models are generated each time. Thus, 10^6 ANN models with different structures are generated in the experiment. The inputs of Algorithm 1 are set as follows: the candidate set for the number of neurons N is $[3, 6, 9]$; the candidate set for the disconnection rate R is $[0.2, 0.3]$; the candidate set for the activation function A is $[ReLU, tanh, logistic]$; the number of input features is 15; and the number of generated ANNs q is 10^6. For each type of activation function, the maximal number of generated ANNs under different settings (n, r), where $n \in N$ and $r \in R$, is shown in Table 2. For each combination of (n, r, a), where $n \in N$, $r \in R$ and $a \in A$, we randomly select 555,556 ANNs to obtain 10^6 ANNs.

Table 2. The maximal number of generated ANNs under each setting of (n, r) for each activation function.

(n, r)	0.3	0.2
3	1.67×10^{11}	8.87×10^{8}
6	6.87×10^{22}	7.13×10^{18}
9	3.19×10^{34}	1.87×10^{28}

Note: n represents a member of the set N (the candidate set for the number of neurons); r represents a member of the set R (the candidate set for the disconnection rate).

The hyperparameters of each ANN, such as the number of HLs, the number of neurons in each HL and the activation function, are set according to Algorithm 1. Other hyperparameters, including the learning rate and the number of epochs, are optimized according to a constructed ANN. This constructed ANN has one HL with nine neurons, its activation function is ReLU, and its 20% of the edges between the IL and the HL are randomly deleted. Therefore, the hyperparameters of this constructed ANN, including the number of HLs, the number of neurons in each HL, and the activation

function are fixed. Based on 400 training samples, we then tune the learning rate and the number of epochs for this constructed ANN using five-fold cross validation combined with a grid search method. Table 3 shows that the search spaces for the learning rate and the number of epochs are $[0.005, 0.01, 0.05, 0.1, 0.2]$ and $[100, 300, 500, 700]$. The optimal learning rate and the number of epochs for this constructed ANN are thus 0.01 and 500, respectively. Then, the learning rate and the number of epochs for each ANN are set to 0.01 and 500. All experiments are conducted using a desktop (Intel Core i7-12700H CPU, 2.30 GHz). The computational time for a single execution of the novel and common processes is 371.32 s and 307.14 s, respectively.

Table 3. Hyperparameters in the ANN model.

Hyperparameter	Search space	Best value
Learning rate	[0.005, 0.01, 0.05, 0.1, 0.2]	0.01
The number of epochs	[100, 300, 500, 700]	500

Two processes are then performed, first to select the best ANN among the 1000 ANN models with different structures and then to compare the best ANN model with the LR model in terms of MSE. These are performed 1000 times to calculate the consistency rate of each process, as noted in Section 4. Table 4 gives the experimental results for the novel process. In 498 out of 1000 executions, our novel process chooses the best ANN, which outperforms the LR model on the small test set, whereas in 502 out of 1000 executions, the best ANN performs worse than the LR model. Of the 498 executions, we find that the best ANN performs better than the LR model on the large test set for 331 executions. Of the 502 executions, the best ANN performs worse than the LR model on the larger test set for 313 executions. Thus, the consistency rate of the novel process is 62.6% according to Eq (8). Table 5 gives the experimental results of the common process, with a consistency rate of 58.1%. The consistency rate of the novel process is 4.5% higher than that of the common process, demonstrating that it performs better in selecting the best ANN with consistent performance among multiple ANN models compared to the common process.

Table 4. Experimental results of novel process.

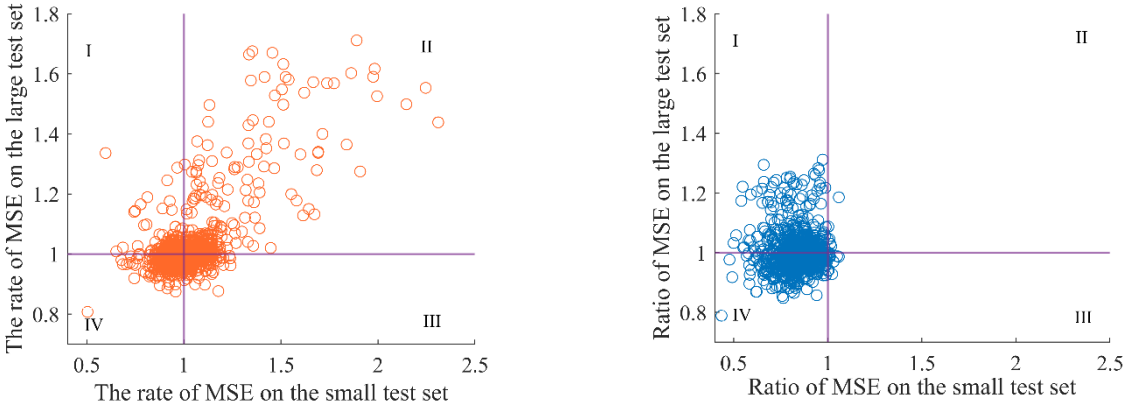
Performance comparison on the small test set	Counts	Performance comparison on the large test set	Counts
Best ANN > LR	498	Best ANN > LR (consistent)	**331**
		LR > Best ANN (not consistent)	167
LR > Best ANN	502	Best ANN > LR (not consistent)	189
		LR > Best ANN (consistent)	**313**
Overall consistency rate = **62.6%**			

Table 5. Experimental results of common process.

Performance comparison on the small test set	Counts	Performance comparison on the large test set	Counts
Best ANN > LR	982	Best ANN > LR (consistent)	**576**
		LR > Best ANN (not consistent)	406
LR > Best ANN	18	Best ANN > LR (not consistent)	13
		LR > Best ANN (consistent)	**5**
Overall consistency rate = **58.1%**			

Note: Greater than sign means that the former prediction model performs better than the latter one in terms of MSE.

In addition, for the two processes, we visually compare the MSE of the best ANN and the LR model in each execution. For the novel process, we depict a scatter plot in Figure 4(a) to show the ratio between MSE of the best ANN model and that of the LR model on the small test set (x-axis) and on the large test set (y-axis), where each point on the plot represents a trial. In region II (where both ratios are larger than 1) and region IV (where both ratios are smaller than 1), the best ANN model demonstrates consistent performance on both the small and large test sets. In contrast, regions I and III correspond to trials where the performance of the best ANN model on the two test sets is inconsistent. Similarly, we also generate such a scatter plot for the common process, as shown in Figure 4(b). It is found that most of the points in regions I and III in Figure 4(a) are concentrated around point (1, 1). This indicates that there is not much deviation in the executions with inconsistent performance (i.e., the best ANN and LR model have similar predictive performance on the two test sets). However, in Figure 4(b), points in region I are far away from point (1, 1), which demonstrates significant inconsistency of these executions (i.e., the best ANN performs highly inconsistently on both test sets).



(a) Novel process

(b) Common process

Figure 4. The ratio between MSE of the best ANN model and that of the LR model on the small test set and on the large test set.

6. Conclusions and future research

With the widespread use of ML techniques in maritime transportation, researchers often train ML models and compare their predictive performance with conventional models on the same test set to confirm that ML models have higher accuracy. Due to the high cost and difficulty of collecting data, the training and comparison processes are typically conducted on a small amount of data, but this process has two major problems. First, many studies use inadequate training data when training ML models, leading to the problem of underfitting, i.e., the ML models cannot learn the patterns in the data well and thus the predictive performance on unseen data is reduced. Second, the common process of comparing the performance of multiple ML models and the conventional model on a small dataset is unfair. To address these problems, we propose a novel process to fairly compare the predictive performance of ML models and the conventional model. In this process, we first select the best model based on the predictive performance of multiple ML models on the validation set, and then retrain it using a combination of the original training set and the validation set. This ML model is then compared

with the conventional model on the same test set.

We perform the two processes 1000 times to calculate the consistency rate, and thus compare the performance of the two processes in terms of their ability to select the best ML model with consistent performance. The results show that the consistency rate of the novel process is 4.5% higher than that of the common process. This demonstrates that the novel process has a higher probability of selecting the best ML model with consistent performance from all candidate ML models. Our study can thus assist researchers in selecting the best ML model in scenarios where there are small datasets, such that the selected model has consistent performance on different test sets, i.e., it performs better/worse than the conventional model in all cases. In addition, such consistent predictive model can provide prediction results for the decision problem in the transport industry. For example, in the berth scheduling problems (see [51–59]), knowing a vessel's arrival time is a prerequisite. However, a vessel's arrival time is uncertain until the vessel actually arrives. Therefore, predictive models such as LR and ML models can to be employed to estimate the vessel's arrival time [60]. Another example is predicting the energy consumption of electric vehicles to better plan the route and charging station choice (see [61–65]). Deploying models with consistent predictive performance could be essential to obtain robust berth scheduling results.

Nevertheless, this research has several limitations and points out future directions, as listed below:
1) One limitation of this study is that only MSE is used as the evaluation metric to select the ML model with the best performance from multiple ML models and to compare the performance of the ML models and the conventional model. Although MSE is a widely used metric for regression problems and provides a measure of the overall error between the predicted and actual values, it may not capture all aspects of the model's performance. For example, an ML model with a low MSE may perform poorly in predicting extreme values. In future research, it would be beneficial to consider a wider range of evaluation metrics, such as mean absolute error or mean absolute percentage error, to obtain a more comprehensive assessment of the models' performance. Additionally, if multiple metrics are used to evaluate model prediction performance, it is important to consider how to combine multiple evaluation metrics to select the best performing ML model and to compare the best performing ML model with the conventional model.
2) Another limitation is that this research only discusses the fair comparison problem in regression problems but not in the classification problems. In order to extend the proposed fair comparison framework to the classification problem, the experimental design should be modified to evaluate the ability of the novel process to select the best ML model with the consistent performance. One concern is the class imbalance problem. In current experimental design, we run the two processes 1000 times, with the whole dataset randomly split into four parts in each run. However, with random splitting of the dataset, there is a risk that some classes may be underrepresented in the training or test set, resulting in underperformance of the predictive model on the minority class. Therefore, splitting the data in a random manner would affect the consistency rate of the novel and common processes. One possible modification could be to use stratified sampling when splitting the dataset. Stratified sampling divides the dataset so that the proportions of classes in the training and test sets are the same as in the overall dataset. This can help to ensure that all classes are adequately represented in the training and test sets, which can lead to more reliable and unbiased results.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. M. I. Jordan, T. M. Mitchell, Machine learning: Trends, perspectives, and prospects, *Science*, **349** (2015), 255–260. https://doi.org/10.1126/science.aaa8415
2. I. H. Sarker, Machine learning: Algorithms, real-world applications and research directions, *SN Comput. Sci.*, **2** (2021), 1–21. https://doi.org/10.1007/s42979-021-00592-x
3. Y. Zhang, C. Ling, A strategy to apply machine learning to small datasets in materials science, *npj Comput. Mater.*, **4** (2018). https://doi.org/10.1038/s41524-018-0081-z
4. N. Ghadami, M. Gheibi, Z. Kian, M. G. Faramarz, R. Naghedi, M. Eftekhari, et al., Implementation of solar energy in smart cities using an integration of artificial neural network, photovoltaic system and classical Delphi methods, *Sustainable Cities Soc.*, **74** (2021), 103149. https://doi.org/10.1016/j.scs.2021.103149
5. R. Yan, S. Wang, H. N. Psaraftis, Data analytics for fuel consumption management in maritime transportation: Status and perspectives, *Transp. Res. Part E Logist. Transp. Rev.*, **155** (2021), 102489. https://doi.org/10.1016/j.tre.2021.102489
6. R. Yan, S. Wang, L. Zhen, G. Laporte, Emerging approaches applied to maritime transport research: Past and future, *Commun. Transp. Res.*, **1** (2021), 100011. https://doi.org/10.1016/j.commtr.2021.100011
7. T. Uyanık, Ç. Karatuğ, Y. Arslanoğlu, Machine learning approach to ship fuel consumption: A case of container vessel, *Transp. Res. Part D Transp. Environ.*, **84** (2020), 102389. https://doi.org/10.1016/j.trd.2020.102389
8. A. Mazaheri, J. Montewka, P. Kujala, Modeling the risk of ship grounding—a literature review from a risk management perspective, *WMU J. Marit. Aff.*, **13** (2014), 269–297. https://doi.org/10.1007/s13437-013-0056-3
9. B. Wu, X. Yan, T. L. Yip, Y. Wang, A flexible decision-support solution for intervention measures of grounded ships in the Yangtze River, *Ocean Eng.*, **141** (2017), 237–248. https://doi.org/10.1016/j.oceaneng.2017.06.021
10. R. Yan, S. Wang, C. Peng, Ship selection in port state control: Status and perspectives, *Marit. Policy Manage.*, **49** (2022), 600–615. https://doi.org/10.1080/03088839.2021.1889067
11. Z. Yang, Z. Yang, J. Yin, Realising advanced risk-based port state control inspection using data-driven Bayesian networks, *Transp. Res. Part A Policy Pract.*, **110** (2018), 38–56. https://doi.org/10.1016/j.tra.2018.01.033
12. Y. Leonov, V. Nikolov, A wavelet and neural network model for the prediction of dry bulk shipping indices, *Marit. Econ. Logist.*, **14** (2012), 319–333. https://doi.org/10.1057/mel.2012.10
13. Z. Yang, E. E. Mehmed, Artificial neural networks in freight rate forecasting, *Marit. Econ. Logist.*, **21** (2019), 390–414. https://doi.org/10.1057/s41278-019-00121-x
14. Q. Bi, K. E. Goodman, J. Kaminsky, J. Lessler, What is machine learning? A primer for the epidemiologist, *Am. J. Epidemiol.*, **188** (2019), 2222–2239. https://doi.org/10.1093/aje/kwz189

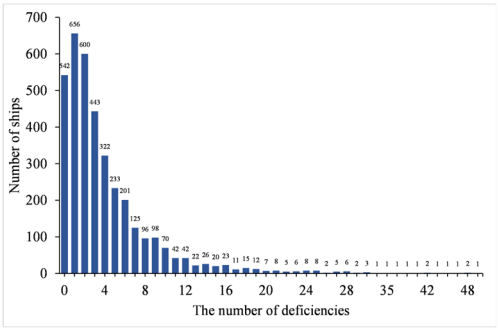
15. F. A. Faber, A. Lindmaa, O. A. Von Lilienfeld, R. Armiento, Machine learning energies of 2 million elpasolite (ABC_2D_6) crystals, *Phys. Rev. Lett.*, **117** (2016), 135502. https://doi.org/10.1103/PhysRevLett.117.135502
16. W. Ng, B. Minasny, W. D. S. Mendes, J. A. M. Demattê, The influence of training sample size on the accuracy of deep learning models for the prediction of soil properties with near-infrared spectroscopy data, *Soil*, **6** (2020), 565–578. https://doi.org/10.5194/soil-6-565-2020
17. C. Baur, S. Albarqouni, N. Navab, Semi-supervised deep learning for fully convolutional networks, in *Medical Image Computing and Computer Assisted Intervention−MICCAI 2017*, Springer, (2017), 311–319. https://doi.org/10.48550/arXiv.1703.06000
18. N. Doulamis, A. Doulamis, Semi-supervised deep learning for object tracking and classification, in *2014 IEEE International Conference on Image Processing (ICIP)*, (2014), 848–852. https://doi.org/10.1109/ICIP.2014.7025170
19. H. Wu, S. Prasad, Semi-supervised deep learning using pseudo labels for hyperspectral image classification, *IEEE Trans. Image Process.*, **27** (2017), 1259–1270. https://doi.org/10.1109/TIP.2017.2772836
20. J. P. Petersen, O. Winther, D. J. Jacobsen, A machine-learning approach to predict main energy consumption under realistic operational conditions, *Ship Technol. Res.*, **59** (2012), 64–72. https://doi.org/10.1179/str.2012.59.1.007
21. D. Ronen, The effect of oil price on the optimal speed of ships, *J. Oper. Res. Soc.*, **33** (1982), 1035–1040. https://doi.org/10.1057/jors.1982.215
22. S. C. Ryder, D. Chappell, Optimal speed and ship size for the liner trades, *Marit. Policy Manage.*, **7** (1980), 55–57. https://doi.org/10.1080/03088838000000053
23. S. Wang, Q. Meng, Sailing speed optimization for container ships in a liner shipping network, *Transp. Res. Part E Logist. Transp. Rev.*, **48** (2012), 701–714. https://doi.org/10.1016/j.tre.2011.12.003
24. C. Gkerekos, I. Lazakis, G. Theotokatos, Machine learning models for predicting ship main engine fuel oil consumption: A comparative study, *Ocean Eng.*, **188** (2019), 106282. https://doi.org/10.1016/j.oceaneng.2019.106282
25. T. Uyanık, Y. Yalman, Ö. Kalenderli, Y. Arslanoğlu, Y. Terriche, C. L. Su, et al., Data-driven approach for estimating power and fuel consumption of ship: A case of container vessel, *Mathematics*, **10** (2022), 4167. https://doi.org/10.3390/math10224167
26. X. Li, Y. Du, Y. Chen, S. Nguyen, W. Zhang, A. Schönborn, et al., Data fusion and machine learning for ship fuel efficiency modeling: Part I–Voyage report data and meteorological data, *Commun. Transp. Res.*, **2** (2022), 100074. https://doi.org/10.1016/j.commtr.2022.100074
27. Y. Du, Y. Chen, X. Li, A. Schönborn, Z. Sun, Data fusion and machine learning for ship fuel efficiency modeling: Part II–Voyage report data, AIS data and meteorological data, *Commun. Transp. Res.*, **2** (2022), 100073. https://doi.org/10.1016/j.commtr.2022.100073
28. S. Wang, R. Yan, X. Qu, Development of a non-parametric classifier: Effective identification, algorithm, and applications in port state control for maritime transportation, *Transp. Res. Part B Methodol.*, **128** (2019), 129–157. https://doi.org/10.1016/j.trb.2019.07.017
29. R. Yan, S. Wang, K. Fagerholt, A semi-"smart predict then optimize" (semi-SPO) method for efficient ship inspection, *Transp. Res. Part B Methodol.*, **142** (2020), 100–125. https://doi.org/10.1016/j.trb.2020.09.014

30. S. Wu, X. Chen, C. Shi, J. Fu, Y. Yan, S. Wang, Ship detention prediction via feature selection scheme and support vector machine (SVM), *Marit. Policy Manage.*, **49** (2022), 140–153. https://doi.org/10.1080/03088839.2021.1875141
31. *WRS*, World Shipping Register, 2023. Available from: https://world-ships.com/.
32. W. Yi, S. Wu, L. Zhen, G. Chawynski, Bi-level programming subsidy design for promoting sustainable prefabricated product logistics, *Cleaner Logist. Supply Chain*, **1** (2021), 100005. https://doi.org/10.1016/j.clscn.2021.100005
33. W. Yi, L. Zhen, Y. Jin, Stackelberg game analysis of government subsidy on sustainable off-site construction and low-carbon logistics, *Cleaner Logist. Supply Chain*, **2** (2021), 100013. https://doi.org/10.1016/j.clscn.2021.100013
34. X. Bai, L. Cheng, Ç. Iris, Data-driven financial and operational risk management: Empirical evidence from the global tramp shipping industry, *Transp. Res. Part E Logist. Transp. Rev.*, **158** (2022), 102617. https://doi.org/10.1016/j.tre.2022.102617
35. X. Chen, S. Wu, Y. Liu, W. Wu, S. Wang, A patrol routing problem for maritime crime-fighting, *Transp. Res. Part E Logist. Transp. Rev.*, **168** (2022), 102940. https://doi.org/10.1016/j.tre.2022.102940
36. Z. Song, W. Tang, R. Zhao, G. Zhang, Implications of government subsidies on shipping companies' shore power usage strategies in port, *Transp. Res. Part E Logist. Transp. Rev.*, **165** (2022), 102840. https://doi.org/10.1016/j.tre.2022.102840
37. Z. Tan, X. Zeng, S. Shao, J. Chen, H. Wang, Scrubber installation and green fuel for inland river ships with non-identical streamflow, *Transp. Res. Part E Logist. Transp. Rev.*, **161** (2022), 102677. https://doi.org/10.1016/j.tre.2022.102677
38. Z. Tan, M. Zhang, S. Shao, J. Liang, D. Sheng, Evasion strategy for a coastal cargo ship with unpunctual arrival penalty under sulfur emission regulation, *Transp. Res. Part E Logist. Transp. Rev.*, **164** (2022), 102818. https://doi.org/10.1016/j.tre.2022.102818
39. L. Zhen, W. Wang, S. Lin, Analytical comparison on two incentive policies for shore power equipped ships in berthing activities, *Transp. Res. Part E Logist. Transp. Rev.*, **161** (2022), 102686. https://doi.org/10.1016/j.tre.2022.102686
40. P. Cariou, M. Q. Mejia Jr, F. C. Wolff, An econometric analysis of deficiencies noted in port state control inspections, *Marit. Policy Manage.*, **34** (2007), 243–258. https://doi.org/10.1080/03088830701343047
41. P. Cariou, M. Q. Mejia, F. C. Wolff, Evidence on target factors used for port state control inspections, *Mar. Policy*, **33** (2009), 847–859. https://doi.org/10.1016/j.marpol.2009.03.004
42. Ş. Şanlıer, Analysis of port state control inspection data: The Black Sea Region, *Mar. Policy*, **112** (2020), 103757. https://doi.org/10.1016/j.marpol.2019.103757
43. *Tokyo MoU*, Black–Grey–White lists, 2017. Available from: https://www.tokyo-mou.org/doc/Flag%20performance%20list%202020.pdf.
44. *Tokyo MoU*, Information sheet of the new inspection regime (NIR), 2014. Available from: https://www.tokyo-mou.org/doc/NIR-information%20sheet-r.pdf.
45. *Paris MoU*, Criteria for responsibility assessment of recognized organizations (RO), 2013. Available from: https://www.parismou.org/criteria-ro-responsibility-assessment.
46. C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag, 2006.

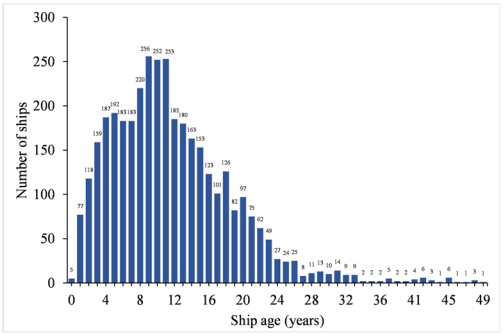
47. T. Hastie, R. Tibshirani, J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer, New York, 2009.
48. M. H. Hassoun, *Fundamentals of Artificial Neural Networks*, MIT press, Cambridge, 1995.
49. K. L. Priddy, P. E. Keller, *Artificial Neural Networks: An Introduction*, Society of Photo-Optical Instrument Engineers (SPIE), Bellingham, 2005.
50. D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, *arXiv preprint*, (2017), arXiv:1412.6980. https://doi.org/10.48550/arXiv.1412.6980
51. A. Dadashi, M. A. Dulebenets, M. M. Golias, A. Sheikholeslami, A novel continuous berth scheduling model at multiple marine container terminals with tidal considerations, *Marit. Bus. Rev.*, **2** (2017), 142–157. https://doi.org/10.1108/MABR-02-2017-0010
52. M. A. Dulebenets, A novel memetic algorithm with a deterministic parameter control for efficient berth scheduling at marine container terminals, *Marit. Bus. Rev.*, **2** (2017), 302–330. https://doi.org/10.1108/MABR-04-2017-0012
53. M. Kavoosi, M. A. Dulebenets, O. Abioye, J. Pasha, O. Theophilus, H. Wang, et al., Berth scheduling at marine container terminals: A universal island-based metaheuristic approach, *Marit. Bus. Rev.*, **5** (2019), 30–66. https://doi.org/10.1108/MABR-08-2019-0032
54. M. Kavoosi, M. A. Dulebenets, O. F. Abioye, J. Pasha, H. Wang, H. Chi, An augmented self-adaptive parameter control in evolutionary computation: A case study for the berth scheduling problem, *Adv. Eng. Inform.*, **42** (2019), 100972. https://doi.org/10.1016/j.aei.2019.100972
55. M. A. Dulebenets, An Adaptive Island Evolutionary Algorithm for the berth scheduling problem, *Memet. Comput.*, **12** (2020), 51–72. https://doi.org/10.1007/s12293-019-00292-3
56. D. Kizilay, D. T. Eliiyi, A comprehensive review of quay crane scheduling, yard operations and integrations thereof in container terminals, *Flexible Serv. Manuf. J.*, **33** (2021), 1–42. https://doi.org/10.1007/s10696-020-09385-5
57. B. G. Zweers, S. Bhulai, R. D. van der Mei, Planning hinterland container transportation in congested deep-sea terminals, *Flexible Serv. Manuf. J.*, **33** (2021), 583–622. https://doi.org/10.1007/s10696-020-09387-3
58. S. Tang, S. Xu, J. Gao, M. Ma, P. Liao, Effect of service priority on the integrated continuous berth allocation and quay crane assignment problem after port congestion, *J. Mar. Sci. Eng.*, **10** (2022), 1259. https://doi.org/10.3390/jmse10091259
59. L. Guo, J. Zheng, H. Du, J. Du, Z. Zhu, The berth assignment and allocation problem considering cooperative liner carriers, *Transp. Res. Part E Logist. Transp. Rev.*, **164** (2022), 102793. https://doi.org/10.1016/j.tre.2022.102793
60. L. Kolley, N. Rückert, M. Kastner, C. Jahn, K. Fischer, Robust berth scheduling using machine learning for vessel arrival time prediction, *Flexible Serv. Manuf. J.*, **35** (2023), 29–69. https://doi.org/10.1007/s10696-022-09462-x
61. J. He, N. Yan, J. Zhang, T. Wang, Battery electric buses charging schedule optimization considering time-of-use electricity price, *J. Intell. Connected Veh.*, **5** (2022), 138–145. https://doi.org/10.1108/JICV-03-2022-0006
62. X. Qu, Y. Liu, Y. Chen, Y. Bie, Urban electric bus operation management: Review and outlook, *J. Automot. Saf. Energy*, **3** (2022), 407–420.
63. C. Sun, B. Liu, F. Sun, Review of energy-saving planning and control technology for new energy vehicles, *J. Automot. Saf. Energy*, **4** (2022), 593–616.

64. H. Wang, M. Ouyang, J. Li, F. Yang, Hydrogen fuel cell vehicle technology roadmap and progress in China, *J. Automot. Saf. Energy*, **2** (2022), 211–224.
65. L. Xu, S. Jin, B. Li, J. Wu, Traffic signal coordination control for arterials with dedicated CAV lanes, *J. Intell. Connected Veh.*, **5** (2022), 72–87. https://doi.org/10.1108/JICV-08-2021-0015

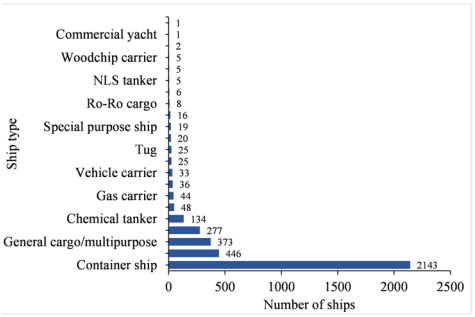
Appendix: Distribution of the features used in this study



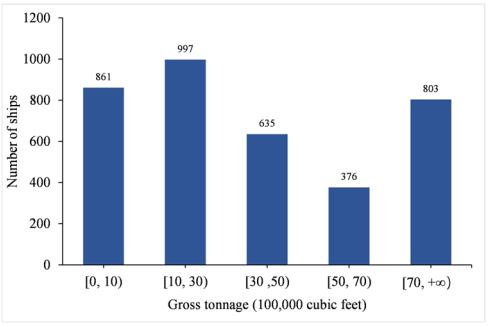
(a) Distribution of the number of deficiencies



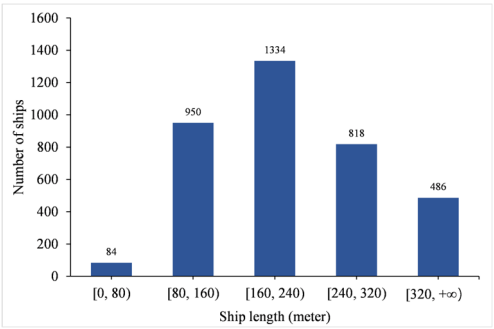
(b) Distribution of ship age



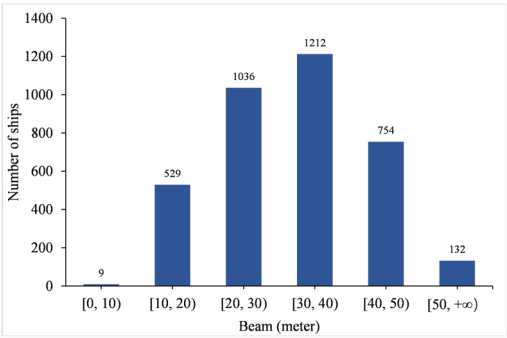
(c) Distribution of ship type



(d) Distribution of gross tonnage

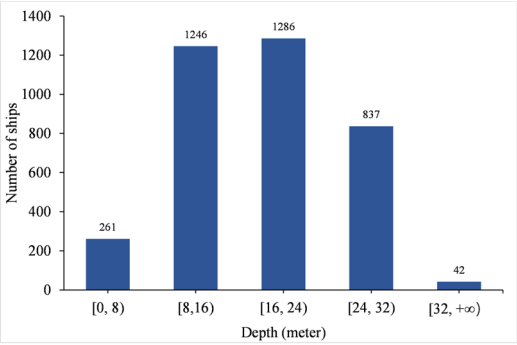


(e) Distribution of ship length

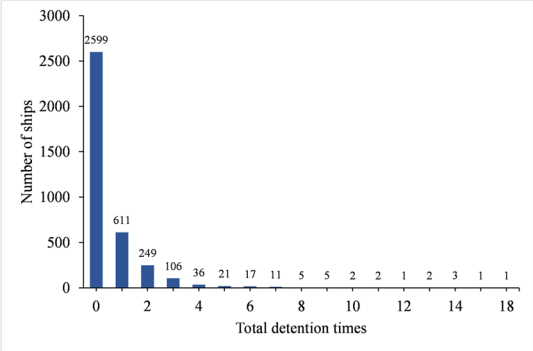


(f) Distribution of ship beam

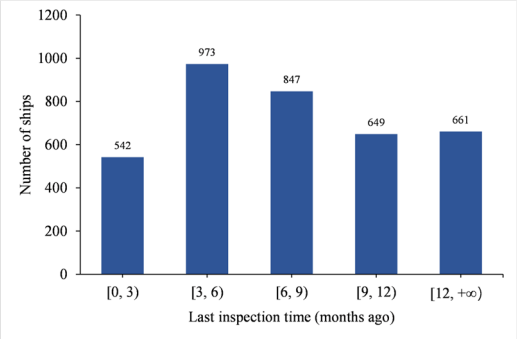
Continued on next page



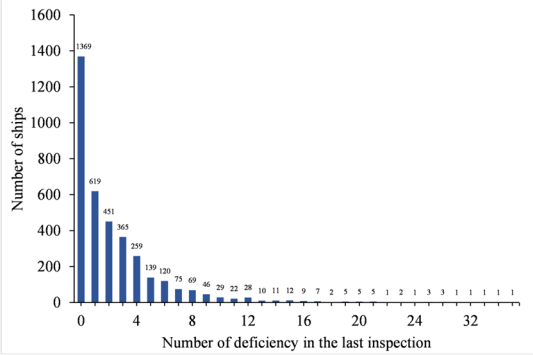
) Distribution of ship depth



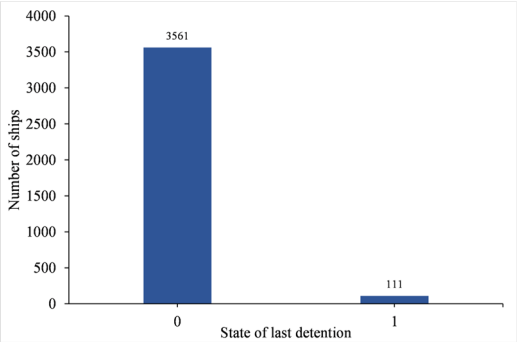
(h) Distribution of total detention times



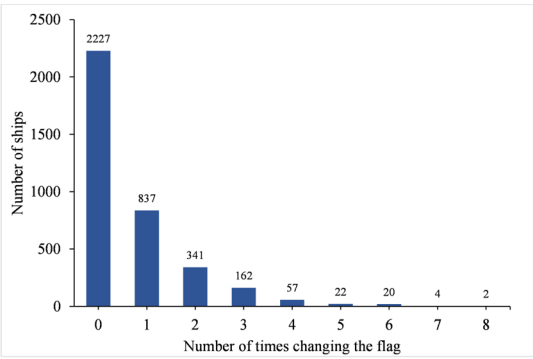
(i) Distribution of last inspection time



(j) Distribution of the number of deficiency in the last inspection

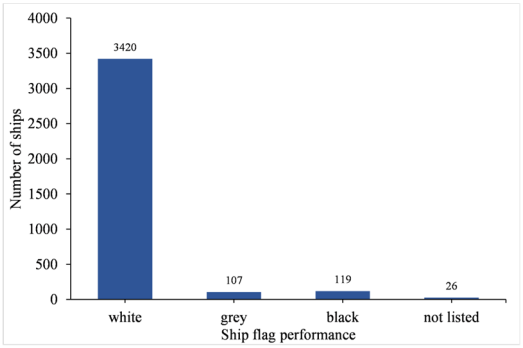


(k) Distribution of the state of last detention

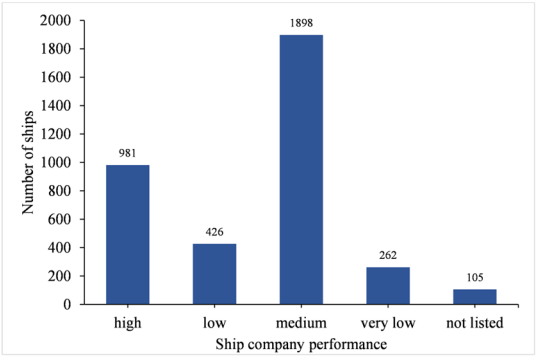


(l) Distribution of the number of times changing the flag

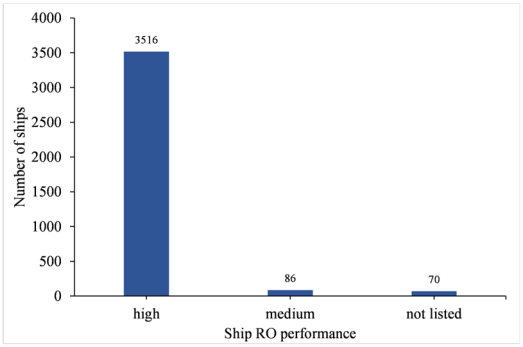
Continued on next page



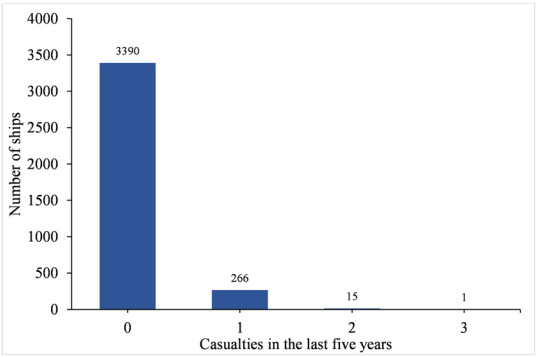
(m) Distribution of the ship flag performance



(n) Distribution of the ship company performance



(o) Distribution of the ship RO performance



(p) Distribution of casualities in the last five years

Figure A1. Distribution of the features used in this study.