mathematics

Article

A Decision-Focused Learning Framework for Vessel Selection Problem

Xuecheng Tian, Yanxia Guan and Shuaian Wang *

Department of Logistics and Maritime Studies, The Hong Kong Polytechnic University, Hung Hom, Hong Kong; xuecheng-simon.tian@connect.polyu.hk (X.T.); minguan2018@gmail.com (Y.G.)
* Correspondence: hans.wang@polyu.edu.hk

Abstract: Maritime transportation safety is pivotal in international trade, with port state control (PSC) inspections being crucial to vessel safety. However, port authorities need to identify substandard vessels effectively because of resource constraints and high costs. Therefore, we propose robust predictive models and optimization strategies for vessel selection, using the random forest (RF) algorithm. We first use a traditional RF model serving as a benchmark, denoted as model M0. Then, we construct model M1 by refining the RF algorithm with a batch-processing method, thereby providing a better measure of the relative relationship between the predicted deficiency counts within a batch of ships. Then, we propose model M2, incorporating a decision-focused learning (DFL) framework into the tree construction process, enhancing the decision performance of the algorithm. In addition, we propose a variant model of M2, denoted as M2-0, considering the worst-case scenario when designing the decision loss function. By conducting experiments with data from the port of Hong Kong, we demonstrate that models M1 and M2 offer superior decision-making performance compared to model M0, and model M2 outperforms model M2-0 in both decision performance and stability. We further verify the robustness of these models by testing them under various instance scales. Overall, our study enhances the PSC inspection efficiency, ultimately bolstering maritime transportation safety.

Keywords: port state control inspection; random forest; decision performance; vessel selection

MSC: 90-10

Citation: Tian, X.; Guan, Y.; Wang, S. A Decision-Focused Learning Framework for Vessel Selection Problem. *Mathematics* **2023**, *11*, 3503. https://doi.org/10.3390/math11163503

Academic Editor: Alejandro Escudero-Santana

Received: 28 July 2023
Revised: 8 August 2023
Accepted: 12 August 2023
Published: 14 August 2023

1. Introduction

Maritime transportation serves as a vital link in international trade and logistics [1,2], and its safety is of paramount importance. With vast amounts of freight transported by sea and the presence of numerous passengers and crew members onboard, the safety of maritime transportation is undeniably non-negotiable [3]. Thus, maritime safety should be at the forefront of our concerns. To enhance vessel safety, PSC inspections are performed globally [4,5]. These inspections are conducted on visiting vessels from various regions, following international conventions [6]. If any aspect of a vessel fails to meet the conventions' stipulations, its deficiencies are duly recorded. If the recorded deficiencies are extensive or particularly critical to a vessel, the port may detain the vessel accordingly [7].

However, port authorities face a significant challenge while conducting PSC inspections: efficiently identifying substandard vessels for scrutiny. Given the expensive costs of inspections and the limited staff resources, this task requires a well-planned approach. Various vessel selection strategies have been developed worldwide. Currently, most port authorities depend on vessels' risk scores for vessel selection. These scores are generated through a weighted sum of factors that are related to vessel conditions, such as ship age and type. While this method is straightforward and easy to implement, it has significant limitations. The scoring process is grounded in subjective expert knowledge, leading to potential bias, which can affect the scheme's effectiveness. Consequently, maritime researchers are exploring more sophisticated and efficient vessel selection models to enhance the efficacy of PSC inspections. In the following, we first review related research on the vessel selection problem and identify several research gaps.

Related Literature

Most of the related research on PSC focuses on improving the inspection efficiency for vessels because the current ship risk profile (SRP) scheme cannot effectively determine whether a vessel is qualified or not. Many researchers try to improve inspection efficiency through the prediction of deficiency counts of vessels or their probabilities of detention. For instance, Wang et al. [8] constructed a tree augmented naive Bayes classifier for the prediction of deficiency counts of vessels, which assists the PSC authorities in determining whether the vessel is qualified or not. Chung et al. [9] obtained the type and the sequence of ship items that should be inspected through a series of experiments. With the rise in ship deficiency prediction models in recent years, predicting vessel deficiency counts also helps to address the allocation of scarce inspection resources. This problem is referred to as the PSC officer (PSCO) scheduling problem as studied by Yan et al. [10,11]. Specifically, Yan et al. [10] compared the deficiency prediction values under different deficiency types of the RF model using different loss functions, and then constructed an optimization model used to achieve an effective match between the inspectors' expertise and the vessels' conditions. Subsequently, Yan et al. [11] combined shipping-related domain knowledge with the extreme gradient boosting model and optimized the PSCO scheduling model.

Most of the reviewed studies tackle the vessel selection problem in two stages. The first stage involves using predictive models to forecast incoming vessels' deficiency counts or detention probabilities. The second stage then selects vessels to inspect based on these predictions by solving optimization models [11–13]. However, these studies, while striving to improve the prediction models' accuracy, often neglect a critical aspect of the vessel selection problem: the emphasis should be on decision performance rather than the quality of the prediction. It is vital to note that a superior prediction model does not necessarily lead to better decisions. This can lead to overlooking the application of the predictive model to the optimization model, possibly resulting in sub-optimal outcomes [14]. Some research has aimed to address this by combining prediction and optimization through a "smart predict and then optimize" (SPO) framework [14–17]. However, most SPO-related studies adopt a worst-case method in their loss functions when the predicted values offer multiple solutions [14]. This worst-case orientation may miss the optimal solution in real-world optimization problems. Additionally, although the study of Yan et al. [15] is the first to adopt an SPO framework for vessel selection, their study is overly reliant on a single decision tree, which can lead to overfitting due to the model's sensitivity to noise [18]. Furthermore, most models rely on a single selection scenario for data analysis when addressing vessel selection problems [15], without considering the robustness of the prediction model under various selection scales. Data testing and analysis should be conducted based on multiple selection scenarios to ensure the prediction model's robustness and stability [19].

To address the identified research gaps, we utilize the RF algorithm to mitigate overfitting and enhance the model's robustness and applicability. We initially develop a benchmark, model M0, which uses the traditional RF model focusing on minimizing the prediction loss. Subsequently, we refine the RF algorithm using the batch-processing method, thereby enabling more accurate measurement of the relative relationships between the predicted deficiency values within a vessel selection problem. This optimized RF algorithm, denoted as model M1, provides an improved selection method. Further optimization is achieved through the implementation of the decision-focused learning (DFL) framework, denoted as model M2. This framework is incorporated into the decision tree construction process, optimizing the decision loss. In detail, this framework is achieved by maximizing the mean of the total actual deficiency counts of all possible optimal solutions based on the predicted values of a vessel selection problem, thereby enhancing the model's decision performance. In parallel, we develop a variant of model M2 following Yan et al. [15], denoted as M2-0, which considers the worst-case solution when designing the loss function. Using data from the port of Hong Kong for experimental validation, we find that compared to model M0, models M1 and M2 demonstrate superior decision-making performance. Moreover, M2 outperforms M2-0, exhibiting greater stability and decision-making efficacy. To further test model robustness, we conduct experiments under various instance scales. Our proposed method consistently outperforms the benchmarks in different instance scales, illustrating its stability and robustness.

The remainder of the paper is structured as follows. Section 2 constructs the mathematical model for the vessel selection problem. Section 3 describes the optimization models proposed in this paper. Section 4 examines and validates the performance of our proposed algorithms using real-world data. Section 5 concludes this paper.

2. Problem Formulation

This study aims to optimize the selection of high-risk vessels for PSC inspections. The goal is to maximize inspection benefits by prioritizing vessels with higher risk factors, quantified by the count of deficiencies per vessel. A higher deficiency count indicates a higher risk factor, making the vessel more likely to be inspected, thus reducing unnecessary inspections on low-risk vessels.

To generalize this problem, we assume that we select n ships with the highest deficiency counts from a pool of N ships, each ship indexed by $i \in \{1, \ldots, N\}$. We denote the deficiency count of ship i as c_i. We use the binary decision variables w_i to indicate whether ship i is selected for inspection ($w_i = 1$) or not ($w_i = 0$). The model is shown as follows:

$$\max_{w} \quad \sum_{i=1}^{N} w_i c_i \tag{1a}$$

$$\sum_{i=1}^{N} w_i \leq n \tag{1b}$$

$$w_i \in \{0, 1\} \quad \forall i \in \{1, \ldots, N\}. \tag{1c}$$

The optimization function (1a) maximizes the total number of identified deficiencies of target vessels. (1b) ensures that the number of selected vessels does not exceed the required number n. By solving this optimization problem, we can determine the subset of n ships with the highest deficiency counts that should be selected for inspection.

3. Methods

This section introduces three methods to predict the deficiency counts of incoming vessels. To select high-risk vessels as described in Section 2, we first construct the model M0 for the prediction of the number of vessel deficiencies based on the original RF algorithm in Section 3.1. In Section 3.2, to better compare the relative number of deficiencies of ships in an optimization problem, we construct model M1 using the batch-processing method. At last, in Section 3.3, we consider the decision-making performance in M1, constructing model M2.

3.1. M0: Traditional RF Method

The RF method is an ensemble learning algorithm used to solve classification and regression problems [20]. Comprising multiple decision trees, the RF model uses decision rules to segment and categorize samples, creating a tree-based predictive model [21,22]. When applying the RF method to classification problems, the category with the most votes from the decision trees is selected as the final prediction. For regression problems, the algorithm averages the predictions from each decision tree to provide a final outcome [23]. The RF method's final prediction results, derived by integrating the output values of multiple decision trees, are often more stable and accurate.

In an RF, we use four common hyperparameters to control the algorithm's dimension [24]: the number of decision trees N_e, the maximum depth of the decision tree M_d, the maximum number of features to be used in one decision tree M_f, and the minimum number of training samples per leaf M_s. In this section, we consider utilizing the traditional RF method to predict the deficiency counts of incoming vessels. We then use these predicted deficiency counts to inform the selection of target vessels.

We use the accumulated historical PSC inspection data $D = \{(x_d, y_d)\}_{d=1}^{|D|}$, obtained from the Asia-Pacific Computerized Information System (https://apcis.tmou.org/public/, accessed on 5 May 2023), to train the RF, where $|D|$ denotes the total number of data samples, x_d is the feature vector of the d-th ship (which contains attributes like ship type, ship age, etc.), and y_d is the deficiency count of the d-th ship. This algorithm outputs the predicted deficiency count of each ship based on its features.

The construction of a decision tree within the RF algorithm depends significantly on the node segmentation principle, with its importance manifesting in three areas: feature selection, division point selection, and evaluation index selection. Firstly, feature selection involves determining appropriate features for division. This decision significantly influences the model's accuracy and efficiency, underscoring the need for a thoughtful approach. Secondly, the division point selection pertains to identifying the optimal split values within the selected features. It requires choosing the most suitable feature value for each specific feature, thereby shaping the structure of the decision tree. Thirdly, evaluation index selection is critical to measure the quality of the dataset division. Selecting an appropriate evaluation index ensures that the divisions made at each node improve the model's performance.

In the context of our regression task, we employ the mean squared error (MSE) as a standard quality measure. Smaller MSE values indicate better prediction quality as shown in (2):

$$MSE_{fs}^{D_c} = \frac{1}{|D_c|} \sum_{\{d|(x_d,y_d)\in D_c\}} (y_d - g_{f,s}(x_d))^2, \tag{2}$$

which is used to obtain the $MSE_{fs}^{D_c}$ based on the chosen feature f and the split value s. Here, D_c means the current training dataset used in this splitting stage, y_d is the real deficiency count of the d-th ship, and $g_{f,s}(x_d)$ means the predicted deficiency count of the d-th ship based on the decision tree under the feature f and the feature value s, calculated as follows:

$$g_{f,s}(x_d) = \frac{1}{|L_{f,s}^{x_d}|} \sum_{\{d'|(x_{d'},y_{d'})\in L_{f,s}^{x_d}\}} y_{d'}, \tag{3}$$

where $L_{f,s}^{x_d}$ represents the corresponding leaf node in which x_d falls, $|L_{f,s}^{x_d}|$ represents the number of examples in $L_{f,s}^{x_d}$, and $y_{d'}$ represents the deficiency count of the d'-th ship in $L_{f,s}^{x_d}$. Finally, $g_{f,s}(x_d)$ is calculated through averaging the outputs of samples in the corresponding leaf node $L_{f,s}^{x_d}$ based on the decision tree under the feature f and the feature value s.

In the RF algorithm, we construct N_e decision trees for the dataset D. Each decision tree T_i, $i \in \{1, 2, \ldots, N_e\}$, is trained using the dataset D_i, which is bootstrapped from D. Dataset D_i contains M_f features, represented by set M_{fi}. During a particular splitting stage in the decision tree T_i, the node segmentation rule is applied to divide the current training dataset D_c^i. We calculate the MSE values for all features in M_{fi} and for every feature value of feature $f \in M_{fi}$ (where each feature value is denoted by $s \in H_f$ and H_f is the set of feature values of feature f). We then select the best feature $f_i^{D_c^i}$ and the best feature value $s_i^{D_c^i}$ based on the smallest MSE value. This selection process is illustrated as follows:

$$(f_i^{D_c^i}, s_i^{D_c^i}) = \mathop{\arg\min}_{f\in M_{fi},\, s\in H_f} \frac{1}{|D_c^i|} \sum_{\{d|(x_d,y_d)\in D_c^i\}} (y_d - g_{f,s}(x_d))^2. \tag{4}$$

In summary, we use the RF method in selecting the target vessels. The whole algorithm is shown in Algorithm 1. Based on the dataset D, the algorithm generates the numerical prediction function $f_{D,\theta^*}(x)$ for estimating the deficiency count of a vessel, where θ^* denotes the optimal set of hyperparameters. The algorithm constructs N_e trees, using the mean of the deficiency counts derived from these trees as the prediction value. It is noted that we repeat step 6 until the depth of T_i (denoted by $Depth(T_i)$) meets M_d or the number of samples of leaf nodes (denoted by $Len(D_c^i)$) is less than M_s.

Algorithm 1 RF Algorithm-M0

1: **Input**: Ship dataset D and hyperparameter tuple $\theta = (N_e, M_d, M_s, M_f)$
2: **Output**: Prediction function $f_{D,\theta}(x)$
3: **for** $i = 1, 2, \ldots, N_e$ **do**
4: Get a random subset D_i from D containing M_f features, represented by M_{fi}
5: Construct tree T_i and set D_i as the current dataset D_c^i
6: **while** $Len(D_c^i) > M_s$ and $Depth(T_i) < M_d$ **do**
7: **for** feature f in M_{fi} **do**
8: **for** s in H_f **do**
9: Split the dataset D_c^i based on f and s and calculate $MSE_{fs}^{D_c^i}$ through (2)
10: **end for**
11: **end for**
12: Select the best feature $f_i^{D_c^i}$ and best feature value $s_i^{D_c^i}$ through (4)
13: Split the dataset based on $f_i^{D_c^i}, s_i^{D_c^i}$, get the dataset for the left child node D_l^i and the dataset for the right child nodeD_r^i
14: Set D_l^i as D_c^i, continue to line 6 to construct the left subtree of T_i
15: Set D_r^i as D_c^i, continue to line 6 to construct the right subtree of T_i
16: **end while**
17: **end for**
18: **return** An ensemble of N_e trees

3.2. M1: Optimized RF Algorithm with Batch Processing Method

The vessel selection problem involves selecting n target vessels out of N vessels. In Section 3.1, we predict the deficiency count for each ship and use these prediction values to guide the selection of target vessels. However, the selection of target vessels is based on the relative ranking of the predicted deficiency counts, not on the absolute deficiency values of each vessel.

To partially consider this characteristic, we adopt the batch-processing approach for our input data, which divides the dataset into smaller subsets equal in size to the current ship selection problem during the training of the RF. For example, consider dividing the current training dataset D_c^i in tree T_i, $i \in \{1, 2, \ldots, N_e\}$, and $D_c^i = \{(x_1, y_1), (x_2, y_2), \ldots, (x_{|D_c^i|}, y_{|D_c^i|})\}$. We apply batch processing to D_c^i, generating multiple subsets $Q_{D_c^i} = \{Q_{D_c^i}^1, Q_{D_c^i}^2, \ldots, Q_{D_c^i}^{k_{D_c^i}}\}$, where $k_{D_c^i} = \left\lfloor \frac{|D_c^i|}{N} \right\rfloor$ and $Q_{D_c^i}^d = \{(x_{(d-1)N+1}, y_{(d-1)N+1}), (x_{(d-1)N+2}, y_{(d-1)N+2}), \ldots, (x_{dN}, y_{dN})\}$, $d \in \{1, 2, \ldots, k_{D_c^i}\}$.

Upon integrating $Q_{D_c^i}$ into the RF training process, our main focus shifts to modifying the node-splitting rules. Rather than directly computing the MSE value of D_c^i, we sum the MSE values for subsets in $Q_{D_c^i}$. When considering the current dataset D_c^i under the feature f and the feature value s, we denote the sum of MSEs of all subsets as $tmse_{fs}^{D_c^i}$. Thus, (4) is transformed into (5), shown as follows:

$$(f_i^{D_c^i}, s_i^{D_c^i}) = \underset{f \in M_{fi},\, s \in H_f}{\arg\min} \sum_{d=1}^{k_{D_c^i}} \sum_{\{j|(x_j,y_j) \in Q_{D_c^i}^d\}} (y_j - g_{f,s}(x_j))^2$$
$$= \underset{f \in M_{fi},\, s \in H_f}{\arg\min} \sum_{d=1}^{k_{D_c^i}} \sum_{\{j|(x_j,y_j) \in Q_{D_c^i}^d\}} (y_j - \frac{1}{|L_{f,s}^{x_j}|} \sum_{\{j'|(x_{j'},y_{j'}) \in L_{f,s}^{x_j}\}} y_{j'})^2, \quad (5)$$

which enables us to obtain the best feature $f_i^{D_c^i}$ and the best feature value $s_i^{D_c^i}$ when adopting the batch-processing method. Specifically, in (5), the MSE is computed through the set $Q_{D_c^i}^d, d \in \{1, 2, \ldots, k_{D_c^i}\}$ and $k_{D_c^i} = \left\lfloor \frac{|D_c^i|}{N} \right\rfloor$. We sum all the MSE values derived from the subsets in $Q_{D_c^i}$ to obtain $tmse_{fs}^{D_c^i}$. In Equation (5), $g_{f,s}(x_j)$ represents the average values of samples in the corresponding leaf node $L_{f,s}^{x_j}$ that x_j falls into as demonstrated in the second term of (5). (5) iterates across all current features and feature values to identify the minimum value, which corresponds to the optimal feature $f_i^{D_c^i}$ and the optimal feature value $s_i^{D_c^i}$.

Batch processing the dataset allows us to more accurately emulate the structure of the vessel selection problem, effectively capturing the characteristics and patterns of the selection issue at hand and enhancing the performance of the RF model. Additionally, the RF method can yield a more precise model evaluation. Typically, the number of vessels in the training and testing sets do not align in scale. By dividing these sets into multiple subsets equal in size to the current vessel selection problem, we can more accurately evaluate the model's performance in the vessel selection task at hand.

We optimized the RF algorithm using batch processing to handle datasets, making it more suitable for vessel selection tasks. This procedure is outlined in Algorithm 2.

3.3. M2: Optimized RF Algorithm under the DFL Framework

Algorithm 1 prioritizes generating highly accurate predictive models for predicting vessel deficiency counts. However, the vessel selection problem's focus extends beyond the accurate prediction of a vessel's deficiency count. As exemplified by Yan et al. [15], an accurate prediction value does not necessarily lead to an optimal vessel selection decision. The primary objective of the vessel deficiency number prediction model, as outlined in Algorithm 1, is to minimize the prediction error. While Algorithm 2 leverages batch processing to mimic the data structure of the current problem scenario and enhance model performance, it utilizes a similar metric to MSE by aggregating MSEs for subsets in $Q_{D_c^i}$.

To better address the vessel selection problem, we introduce an alternative approach: the DFL framework. Instead of using prediction loss to train the model, we use decision loss. This metric measures the discrepancy between benefits derived from decisions based on predictive outcomes and those gained from decisions based on actual and known values. Crucially, to achieve this objective, we need to adapt the model's loss function. This modification shifts our focus towards decision-making accuracy rather than strictly emphasizing prediction accuracy, necessitating the use of decision loss as the evaluation index in the node segmentation process.

The target vessel selection problem entails selecting n vessels from a pool of N vessels. For each decision-making group $q \in Q_{D_c^i}$—divided based on the batch-processing method when the current training dataset is D_c^i and in tree T_i, $i \in \{1, 2, \ldots, N_e\}$—we choose the top n target vessels from dataset q based on the predicted values in the set q. We denote the set of these chosen indices as $P(n, q, f, s)$ under the feature f and the feature value s. It is worth noting that there may be ties of predictive values, resulting in multiple sets of possible $P(n, q, f, s)$ with an equal number of total ship deficiency counts, particularly during the early stages of the decision tree construction. For instance, in the $N = 3$, $n = 1$ scenario, three samples from the subset q might appear in the same leaf during the initial tree generation phase, each with identical deficiency prediction values. If selecting the ship with the highest predicted deficiency value at this stage, there would be three possible selection results for $P(n, q, f, s)$. In such cases, existing works in the literature, such as those of Yan et al. [15] and Elmachtoub et al. [25], adopt a worst-case decision-making approach to define $P(n, q, f, s)$, as demonstrated in Definition 1.

Algorithm 2 RF optimized with batch-processing method-M1

1: **Input**: Ship DataSet D and hyperparameter tuple $\theta = (N_e, M_d, M_s, M_f)$
2: **Output**: Prediction Function $f^b_{D,\theta}(x)$ optimized by batch-processing method
3: **for** $i = 1, 2, \ldots\ N_e$ **do**
4: Get a random subset D_i from D containing M_f features, represented by M_{fi}
5: Construct tree T_i and set D_i as the current dataset D^i_c
6: **while** $Len(D^i_c) > M_s$ and $Depth(T_i) < M_d$ **do**
7: **for** feature f in M_{fi} **do**
8: **for** s in H_f **do**
9: Divide D^i_c into multiple subsets, $Q_{D^i_c} = \{Q^1_{D^i_c}, Q^2_{D^i_c}, \ldots, Q^{k_{D^i_c}}_{D^i_c}\}, k_{D^i_c} = \lfloor \frac{|D^i_c|}{N} \rfloor$
10: $Q^d_{D^i_c} = \{(x_{(d-1)N+1}, y_{(d-1)N+1}), (x_{(d-1)N+2}, y_{(d-1)N+2}), \ldots, (x_{dN}, y_{dN})\},$
11: $d \in \{1, 2, \ldots, k_{D^i_c}\}$
12: **for** q in $Q_{D^i_c}$ **do**
13: Split the dataset q based on f and s and calculate mse^q_{fs} through (2)
14: **end for**
15: $tmse^{D^i_c}_{fs} = \sum_{\{q|q \in Q^i_c\}} mse^q_{fs}$
16: **end for**
17: **end for**
18: Select the best feature $f^{D^i_c}_i$ and best feature value $s^{D^i_c}_i$ calculated through (5)
19: Split D^i_c based on $f^{D^i_c}_i, s^{D^i_c}_i$, getting left child node D^i_l and right child nodeD^i_r
20: Set D^i_l as D_{c^i}, continue to line 6 to construct the left subtree of T_i
21: Set D^i_r as D_{c^i}, continue to line 6 to construct the right subtree of T_i
22: **end while**
23: **end for**
24: **return** An ensemble of N_e trees

Definition 1. *The set $P(n, q, f, s)$ is defined as follows: (1) $P(n, q, f, s) \subset q$, (2) $|P(n, q, f, s)| = n$, and (3) for all $(x_d, y_d) \in P(n, q, f, s)$ and $(x_{d'}, y_{d'}) \notin P(n, q, f, s)$, $(x_{d'}, y_{d'}) \in q$, one of the following two conditions hold: (1) $g_{f,s}(x_d) > g_{f,s}(x_{d'})$, and (2) $g_{f,s}(x_d) = g_{f,s}(x_{d'}), y_d \leq y_{d'}$.*

Definition 1 provides three conditions to guide the selection of target vessels. Condition (1) implies that target ships are chosen from the top n vessels, ranked according to the predicted deficiency counts in set q, to form the set $P(n, q, f, s)$. Condition (2) stipulates that the size of target vessels should be n, while Condition (3) states that in the event of a tie—when two vessels have identical predicted deficiency counts, $g_{f,s}(x_d) = g_{f,s}(x_{d'})$—the vessel with the lower actual deficiency count should be included in set $P(n, q, f, s)$.

Even with the guidance of Definition 1, there may be instances where the predicted and actual deficiency numbers for two vessels are identical. In such scenarios, a vessel will be randomly selected from these equally ranked options. In line with this, when constructing a decision-focused tree T_i, $i \in \{1, 2, \ldots, N_e\}$, Yan et al. [15] employ the sum of actual deficiency values for vessels in set $P(n, q, f, s)$ as the evaluative criterion in the node-splitting process. In practical scenarios, ties may frequently occur—indicating that more than one node-split scenario leads to the maximum total actual deficiencies. In the event of a tie, the traditional MSE metric, as depicted in (5), remains in use for node splitting.

Nonetheless, the design of $P(n, q, f, s)$ may not be consistent with reality. For example, when $N = 3$, $n = 1$, the worst-case scenario assumes that the decision maker may choose the worst-case combination of ships with an equal number of predicted deficiency counts. However, in reality, the expected sum of identified ship deficiencies should equal n times the average of the actual number of ship deficiencies. Thus, we consider using the average of all alternative combinations, whose sums of predicted deficiency counts are the highest, as the evaluation criterion.

Specifically, we define set $P'(n,q,f,s)$, denoting the set of combinations of n samples in set q whose sums of predicted deficiency numbers are the highest. The average value of all the combinations in $P'(n,q,f,s)$ is computed as follows:

$$\frac{\sum_{\{p|p\in P'(n,q,f,s)\}}\sum_{\{d|(x_d,y_d)\in p\}} y_d}{|P'(n,q,f,s)|}. \tag{6}$$

Hence, when the current training dataset is D_c^i in one decision-focused tree T_i, $i \in \{1,2,\ldots,N_e\}$, we redefine the node segmentation rules, utilizing the calculation for averaging all the combinations in $P'(n,q,f,s)$. Firstly, we process the dataset D_c^i using the similar method in Algorithm 2, dividing the D_c^i into multiple subset $Q_{D_c^i} = \{Q^1_{D_c^i}, Q^2_{D_c^i}, \ldots, Q^{k_{D_c^i}}_{D_c^i}\}$, $k_{D_c^i} = \left\lfloor \frac{|D_c^i|}{N} \right\rfloor$, where $Q^d_{D_c^i} = \{(x_{(d-1)N+1}, y_{(d-1)N+1}), (x_{(d-1)N+2}, y_{(d-1)N+2}), \ldots, (x_{dN}, y_{dN})\}, d \in \{1,2,\ldots,k_{D_c^i}\}$ and $(x,y) \in D_c^i$. For each set q in $Q_{D_c^i}$, the average of all the combinations' objective function values in $P'(n,q,f,s)$ is calculated by (6) and we define the outcome of it as dl^q_{fs}. Correspondingly, when we evaluate the overall real deficiencies of target vessels in $Q_{D_c^i}$, we sum the results for all subsets of $Q_{D_c^i}$, and denote it as $dl^{D_c^i}_{fs}$ under the current feature f and the feature value s. Then, the node is segmented based on the feature $f_i^{D_c^i}$ and its corresponding feature value $s_i^{D_c^i}$ with the highest total number of actual deficiencies in set $Q_{D_c^i}$ and calculated by

$$(f_i^{D_c^i}, s_i^{D_c^i}) = \underset{f\in M_{fi},\, s\in H_f}{\arg\max} \sum_{q\in Q_{D_c^i}} \frac{\sum_{\{p|p\in P'(n,q,f,s)\}}\sum_{\{d|(x_d,y_d)\in p\}} y_d}{|P'(n,q,f,s)|}. \tag{7}$$

In summary, we modify the loss function shown in (7) and adopt the batch-processing method to handle the training dataset. The whole process is shown in Algorithm 3.

Indeed, the evolution from M0 to M2 demonstrates a progression in the complexity and potential performance of the decision-focused models in the RF algorithm. In M0, the basic decision tree construction within the RF algorithm is employed without any alterations. In M1, a batch-processing method is incorporated to handle the dataset, dividing the input into smaller and manageable chunks. This strategy helps to improve the performance of the model by increasing the computational efficiency and potentially enhancing the accuracy of predictions. M2 goes a step further, introducing a significant modification to the evaluation criteria in the node-splitting process during the construction of decision-focused trees. This adjustment takes into account the combination of target vessels that lead to the highest sum of predicted deficiencies, providing a more nuanced and realistic approach to decision-making scenarios. Consequently, this adjustment aims to deliver better model performance and more accurate predictions.

The progressive enhancements from M0 to M2 demonstrate a trajectory towards increasingly sophisticated and realistic modeling of decision-making situations in the context of targeting vessel selection based on predicted deficiency counts. The enhancements aim to align the models more closely with real-world complexities and improve their performance in decision-making tasks.

Algorithm 3 RF optimized with DFL framework-M2

1: **Input**: The same as Algorithm 2
2: **Output**: A final prediction function $f_{D,\theta}^{DFL}(x)$ optimized by DFL framework
3: **for** $i = 1, 2, \ldots, N_e$ **do**
4: Get a random subset D_i from D containing M_f features, represented by M_{fi}
5: Construct tree T_i and set D_i as the current dataset D_c^i
6: **while** $Len(D_c^i) > M_s$ and $Depth(T_i) < M_d$ **do**
7: **for** feature f in M_{fi} **do**
8: **for** s in H_f **do**
9: Divide D_c^i into multiple subsets, $Q_{D_c^i} = \{Q_{D_c^i}^1, Q_{D_c^i}^2, \ldots, Q_{D_c^i}^{k_c}\}, k_{D_c^i} = \lfloor \frac{|D_c^i|}{N} \rfloor$
10: $Q_{D_c^i}^d = \{(x_{(d-1)N+1}, y_{(d-1)N+1}), (x_{(d-1)N+2}, y_{(d-1)N+2}), \ldots, (x_{dN}, y_{dN})\},$
11: $d = \{1, 2, \ldots, k_{D_c^i}\}$
12: **for** q in $Q_{D_c^i}$ **do**
13: Split q based on the f and s and calculate dl_{fs}^q and mse_{fs}^q through (6) and (2), respectively
14: **end for**
15: $dl_{fs}^{D_c^i} = \sum_{\{q|q \in Q_{D_c^i}\}} dl_{fs}^q$, $tmse_{fs}^{D_c^i} = \sum_{\{q|q \in Q_{D_c^i}\}} mse_{fs}^q$
16: **end for**
17: **end for**
18: Select the max value of $dl_{fs}^{D_c^i}$ calculated through (7)
19: **if** $dl_{fs}^{D_c^i}$ has more than one maximum value **then**
20: Select the min value of $tmse_{fs}^{D_c^i}$ through (5)
21: **end if**
22: Select the best feature $f_i^{D_c^i}$ and best feature value $s_i^{D_c^i}$ based on line 17–20
23: Split D_c^i based on $f_i^{D_c^i}, s_i^{D_c^i}$, get the dataset for the left child node D_l^i and the dataset for the right child node D_r^i
24: Set D_l^i as D_c^i, continue to line 6 to construct the left subtree of T_i
25: Set D_r^i as D_c^i, continue to line 6 to construct the right subtree of T_i
26: **end while**
27: **end for**
28: **return** An ensemble of N_e decision-focused trees

4. Evaluation

In this section, we carry out a case study to evaluate the efficacy and robustness of our proposed models M0, M1, and M2. Specifically, Section 4.1 provides a comprehensive description of the experimental setup. Subsequently, we evaluate and compare the prediction performance and decision-making quality of the three proposed models within the context of the vessel selection problem in Section 4.2.

4.1. Experiment Settings

In our experiments, the dataset D is derived from PSC records in the port of Hong Kong from January 2015 to December 2019, a total of 3026 entries. The PSC inspection records of the Port of Hong Kong are sourced from the Asia-Pacific Computerized Information System (https://apcis.tmou.org/public/, accessed on 1 May 2023), while ship-related factors are sourced from the World Shipping Register database (https://world-ships.com/, accessed on 1 May 2023). We consider multiple features that are closely related to ship conditions. These include ship age, gross tonnage, length, depth, beam, ship type, the total detention times of the ship, the total number of the flag changes, the total number of casualties in the last 5 years, the total number of deficiencies in the last inspection, and flag performance, recognized organization performance, and company performance in the Tokyo MoU. The data-processing procedures of these features follow those of Yan et al. [15].

As described in Section 3, we first construct four models. Among them, M0 is the traditional RF model, focusing on minimizing prediction loss of the whole training dataset, seen as the benchmark. M1 is the optimized RF algorithm using the batch-processing method for the input data, whose purpose is to simulate the current selection scenario and enhance the model performance. M2 is the optimized RF algorithm based on the DFL framework, whose purpose is to minimize decision loss. Meanwhile, we also construct M2-0 based on Definition 1 in Section 3.3, as the benchmark to M2.

We divide D into the training set D_{train} and the test set D_{test} at a 4:1 ratio. To effectively evaluate model performance, we process the test dataset in batches, simulating the current ship selection problem framework. D_{test} is divided into multiple subsets, $Q_{D_{test}} = \{Q^1_{D_{test}}, Q^2_{D_{test}}, \ldots, Q^{k_{D_{test}}}_{D_{test}}\}$, $k_{D_{test}} = \left\lfloor \frac{|D_{test}|}{N} \right\rfloor$, where $Q^d_{D_{test}} = \{(x_{(d-1)N+1}, y_{(d-1)N+1}), (x_{(d-1)N+2}, y_{(d-1)N+2}), \ldots, (x_{dN}, y_{dN})\}$, $d \in \{1, 2, \ldots, k_{D_{test}}\}$ and $(x, y) \in D_{test}$. We use the MSE of the real deficiency values versus the predicted deficiency values in the test set to assess the prediction quality, calculated through (8). In (8), y'_d means the predicted value of the d-th ship in q, obtained from the model we proposed. Meanwhile, we use the sum of the real deficiency counts of the target vessels to evaluate the decision performance, shown in (9). In (9), the set $O(n, q)$ is the top n vessels obtained from the predicted values y'_d sorted from largest to smallest:

$$mse_{D_{test}} = \sum_{q \in Q_{D_{test}}} \sum_{\{d | (x_d, y_d) \in q\}} (y_d - y'_d)^2 \tag{8}$$

$$dp_{D_{test}} = \sum_{q \in Q_{D_{test}}} \sum_{\{d | (x_d, y_d) \in O(n,q)\}} y_d. \tag{9}$$

To simplify the experiments, we first assume that we select one ship from three ships, namely $N = 3$, $n = 1$. We select this particular scale to facilitate testing, and we also experiment with different scales to demonstrate the robustness of our models, which means that our models can be used across a wider range of scales. Furthermore, to improve computational efficiency and reduce computational expenses, we adopt fixed hyperparameters when performing model training, where $N_e = 100$, $M_d = 5$, $M_s = 1$, $M_f = 3$, These specific hyperparameter values are chosen through grid search, taking into account the prior experience and practical considerations. By fixing the hyperparameters of the RF algorithm, we can focus more on the optimization aspects of the model, thus improving its performance.

4.2. Evaluation of Models

In the previous sections, we introduce various models for addressing the high-risk ship selection problem, where we aim to choose n ships with higher risk factors for inspection from a total of N ships. We consider the number of ship deficiencies as the indicator of its risk factor, as a higher number of deficiencies indicates a higher risk and a greater likelihood of requiring inspection. Initially, we use the $N = 3$, $n = 1$ scenario for testing purposes. Subsequently, to validate the robustness of the model, we randomly modify the values of N and n and construct scenarios, such as $N = 4$, $n = 2$ and $N = 3$, $n = 2$ scenarios for testing.

We now proceed to compare and analyze the experimental results of models M0, M1, M2-0, and M2, with the aim of elucidating the differences in model performance. We offer a detailed evaluation of how these models perform, concerning the evaluation metrics of MSE ($mse_{D_{test}}$) and decision performance ($dp_{D_{test}}$). The overall results are shown in Table 1. Through this comparative analysis, our goal is to deepen understanding and provide a solid foundation for choosing the optimal model for high-risk ship selection scenarios.

Table 1. The prediction quality and decision performance of models.

N, n	Decision Performance				MSE			
	M0	M1	M2-0	M2	M0	M1	M2-0	M2
$N = 3, n = 1$	1108.0	1125.0	1120.0	1125.0	4256.26	4461.55	4416.34	4459.98
$N = 3, n = 2$	897.50	904.00	897.00	905.50	2915.98	3046.46	3115.49	3112.72
$N = 4, n = 2$	772.50	775.50	776.50	777.00	2487.02	2622.47	2656.03	2639.85

Table 1 presents the experimental results. From the data analysis, we find that while our proposed optimization models M1 and M2 exhibit a slight increase in prediction error ($mse_{D_{test}}$), their decision performance ($dp_{D_{test}}$), which is critical for ship selection, is improved compared to the benchmark model M0.

Particularly, model M2 has a rise in prediction error of about 196, reducing the prediction quality by roughly 6.7% compared to M0, for the instance scale $N = 3, n = 2$. However, model M2, based on the DFL framework, and model M1, utilizing the batch-processing method, improve decision performance by approximately 1.6% (an increase of 17) compared to M0 for the instance scale $N = 3, n = 1$. Despite the same decision performance for M1 and M2, model M2 reduces prediction error by 2, improving the prediction quality by approximately 0.04% compared to M1, indicating a better overall performance.

Compared to the benchmark Model M2-0, which uses the worst-case method, model M2 is more stable and improves the decision performance by roughly 0.8% in the $N = 3, n = 2$ scenario. Therefore, we interestingly observe that better prediction quality does not necessarily lead to better decision performance, evidenced in the $N = 3, n = 2$ scenario. This counterintuitive result provides a nuanced perspective on prediction and decision-making performance. Furthermore, models M1 and M2 outperform the benchmark M0 in multiple instance scales, demonstrating adaptability to diverse data. These results attest to the robustness and stability of our proposed models.

In conclusion, our proposed models demonstrate superior decision-making performance for high-risk vessel selection problems as shown in Table 1. Summarizing the above experimental results, we conclude that in our experiments, model M2, although not the most effective in terms of MSE, outperforms the other two models in terms of decision performance, having the largest value of $dp_{D_{test}}$. Overall, our experimental results provide support for the validity and robustness of models M2 and M1, providing useful experience for solving the high-risk ship selection problem.

5. Conclusions

PSC inspections play a pivotal role in ensuring the safety of maritime transportation. Port authorities face the challenging task of selecting from numerous incoming vessels those that need to be inspected, in a bid to maximize inspection efficiency and overall benefits. In this study, we use a vessel's deficiency count as a measure of its risk level and also as a representation of the inspection's benefit. We first utilize a classical RF algorithm as a benchmark for predicting deficiency counts in vessels in model M0. To better mirror the current selection framework and enhance decision making, we employ batch processing in model M1. Finally, in model M2, we use the decision performance to guide the node-splitting process in the RF model, effectively integrating the prediction and decision processes.

In order to validate our models, we employ the Port of Hong Kong as a case study and construct prediction models M0, M1, and M2. These models are evaluated based on MSE and decision performance. Our analysis of the experimental data shows that, while there is no improvement in prediction quality, our proposed models M1 and M2 surpass model M0 in decision performance. These findings demonstrate the effectiveness and robustness of our models in high-risk vessel selection problems and can serve as valuable references for other studies that address similar issues. Our research underlines the significance of a data-driven and scientifically rigorous approach to vessel selection for PSC inspections, with the potential to enhance maritime safety and operational efficiency.

Author Contributions: Conceptualization, X.T. and S.W.; methodology, S.W.; software, Y.G.; validation, Y.G.; formal analysis, Y.G.; investigation, Y.G.; resources, S.W.; data curation, S.W.; writing—original draft preparation, X.T. and Y.G.; writing—review and editing, X.T. and S.W.; supervision, S.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Available if requested.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ng, M. Container vessel fleet deployment for liner shipping with stochastic dependencies in shipping demand. *Transp. Res. Part B Methodol.* **2015**, *74*, 79–87. [CrossRef]
2. Zhuo, S.; Zheng, J. Finding potential hub locations for liner shipping. *Transp. Res. Part B Methodol.* **2016**, *93*, 750–761. [CrossRef]
3. Christiansen, M.; Fagerholt, K.; Nygreen, B.; Ronen, D. Maritime transportation. *Handb. Oper. Res. Manag. Sci.* **2007**, *14*, 189–284. [CrossRef]
4. Cariou, P.; Mejia, M.Q., Jr.; Wolff, F.C. On the effectiveness of port state control inspections. *Transp. Res. Part E Logist. Transp. Rev.* **2008**, *44*, 491–503. [CrossRef]
5. Cariou, P.; Mejia, M.Q.; Wolff, F.C. Evidence on target factors used for port state control inspections. *Mar. Policy* **2009**, *33*, 847–859. [CrossRef]
6. Heij, C.; Bijwaard, G.E.; Knapp, S. Ship inspection strategies: Effects on maritime safety and environmental protection. *Transp. Res. Part D Transp. Environ.* **2011**, *16*, 42–48. [CrossRef]
7. Li, K.X.; Zheng, H. Enforcement of law by the Port State Control (PSC). *Mar. Policy Manag.* **2008**, *35*, 61–71. [CrossRef]
8. Wang, S.; Yan, R.; Qu, X. Development of a non-parametric classifier: Effective identification, algorithm, and applications in port state control for maritime transportation. *Transp. Res. Part B Methodol.* **2019**, *128*, 129–157. [CrossRef]
9. Chung, W.H.; Kao, S.L.; Chang, C.M.; Yuan, C.C. Association rule learning to improve deficiency inspection in port state control. *Mar. Policy Manag.* **2020**, *47*, 332–351. [CrossRef]
10. Yan, R.; Wang, S.; Fagerholt, K. A semi-"smart predict then optimize"(semi-SPO) method for efficient ship inspection. *Transp. Res. Part B Methodol.* **2020**, *142*, 100–125. [CrossRef]
11. Yan, R.; Wang, S.; Cao, J.; Sun, D. Shipping domain knowledge informed prediction and optimization in port state control. *Transp. Res. Part B Methodol.* **2021**, *149*, 52–78. [CrossRef]
12. Yang, Z.; Yang, Z.; Yin, J. Realising advanced risk-based port state control inspection using data-driven Bayesian networks. *Transport. Res. Part A Policy Pract.* **2018**, *142*, 38–56. [CrossRef]
13. Yang, Z.; Yang, Z.; Yin, J.; Qu, Z. A risk-based game model for rational inspections in port state control. *Transport. Res. Part A Policy Pract.* **2018**, *118*, 477–495. [CrossRef]
14. Elmachtoub, A.N.; Grigas, P. Smart "predict, then optimize". *Manag. Sci.* **2022**, *68*, 9–26. [CrossRef]
15. Yan, R.; Wang, S.; Zhen, L. An extended smart "predict, and optimize"(SPO) framework based on similar sets for ship inspection planning. *Transp. Res. Part E Logist. Transp. Rev.* **2023**, *68*, 9–26. [CrossRef]
16. Tian, X.; Yan, R.; Wang, S.; Laporte, G. Prescriptive analytics for a maritime routing problem. *Ocean Coast. Manag.* **2023**, *242*, 106695. [CrossRef]
17. Tian, X.; Yan, R.; Liu, Y.; Wang, S. A smart predict-then-optimize method for targeted and cost-effective maritime transportation. *Transp. Res. Part B Methodol.* **2023**, *172*, 32–52. [CrossRef]
18. Ali, J.; Khan, R.; Ahmad, N.; Maqsood, I. Random forests and decision trees. *Int. J. Comput. Sci. Issues (IJCSI)* **2012**, *9*, 272.
19. Wagenmakers, E.J.; Sarafoglou, A.; Aczel, B. One statistical analysis must not rule them all. *Nature* **2022**, *605*, 423–425. [CrossRef]
20. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32.:1010933404324. [CrossRef]
21. Myles, A.J.; Feudale, R.N.; Liu, Y.; Woody, N.A.; Brown, S.D. An introduction to decision tree modeling. *J. Chemom. A J. Chemom. Soc.* **2004**, *18*, 275–285. [CrossRef]
22. Kotsiantis, S.B. Decision trees: A recent overview. *Artif. Intell. Rev.* **2013**, *39*, 261–283. [CrossRef]
23. Biau, G. Analysis of a random forests model. *J. Mach. Learn. Res.* **2012**, *13*, 1063–1095. [CrossRef]

24. Probst, P.; Wright, M.N.; Boulesteix, A.L. Hyperparameters and tuning strategies for random forest. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2019**, *9*, e1301. [CrossRef]
25. Elmachtoub, A.N.; Liang, J.C.N.; McNellis, R. Decision trees for decision-making under the predict-then-optimize framework. *Int. Conf. Mach. Learn.* **2020**, *9*, 2858–2867. [CrossRef]