



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

Dissertation for the Degree of Doctor of Philosophy

Cost-sensitive Learning in Estimation of Anchor Dragging Risk for a Vessel at Anchor

Shem Otoi Onyango

Department of Marine Industry & Maritime Police

College of Ocean Sciences

Graduate School

Jeju National University






February 2024

Cost-sensitive Learning in Estimation of Anchor Dragging Risk for a Vessel at Anchor

A dissertation submitted to the graduate school of Jeju National University in partial fulfillment of the requirement for the degree of
Doctor of Philosophy
under the supervision of
Professor Kim Kwang-II

The dissertation for the degree of Doctor of Philosophy by
Shem Otoi Onyango
has been approved by the dissertation committee.

Date: December 2023

Chairman	Prof. Kang Won-Sik	 (seal)
Vice-chairman	Prof. Kim Kwang-II	 (seal)
Member	Prof. Moon Ilju	 (seal)
External examiner	Prof. Jung Cho-Young	 (seal)
External examiner	Dr. Yoo Sang-Lok	 (seal)

Dedicated to James Onyango Olendo and Daniel Otinyo Wanjero.

Contents

List of Tables	iii
List of Figures	iv
List of Abbreviations	vii
Abstract	ix
Abstract in Korean	xi
1 Introduction	1
2 Literature Review	4
3 Anchorage dataset extraction	8
3.1 Behavioral characteristics of ship at anchorage	9
3.2 Features of an AIS data	12
3.3 Research study area	15
3.4 Anchorage dataset extraction	17
4 Cost-sensitive learning on imbalanced data	20
4.1 Class imbalance problem	20
4.2 Cost-sensitive learning	22
4.2.1 Cost-sensitive LR	23
4.2.2 Cost-sensitive SVM	24
4.2.3 Cost-sensitive XGB	24
4.2.4 BBC	25
4.3 Data resampling	26
4.3.1 RU undersampler	26
4.3.2 RENN undersampler	27
4.3.3 AKNN undersampler	27
4.3.4 OSS undersampler	27
4.3.5 NCR undersampler	28
4.4 Performance metrics	29

5	Data analysis and experiment procedure	31
5.1	Class dispersion	31
5.2	Exploratory data analysis	33
5.2.1	Dragging samples analysis	38
5.3	CSA and resampling techniques	41
6	Experiment results and discussion	49
6.1	Case study validation	53
6.2	Risk criterion	57
6.3	Anchor safety assessment at anchorage areas	62
7	Conclusion and recommendations	66
7.1	Conclusion	66
7.2	Recommendation	67
	APPENDICES	68
	REFERENCES	74
	Acknowledgement	81

List of Tables

Table 1	A summary of the studies on anchor dragging by various authors.	7
Table 2	Transmission frequency of AIS messages in relation to dynamic information.	13
Table 3	AIS broadcast data.	14
Table 4	Capacity, area, seabed condition and depth of E-group anchorage in Ulsan port. Source: (Park et al., 2016)	16
Table 5	Constitution of the anchorage dataset.	19
Table 6	The cost matrix for a binary class classification.	22
Table 7	The confusion matrix for a binary class classification.	29
Table 8	Measure of class dispersion in comparison to RFECV.	32
Table 9	Descriptive statistics for the continuous features of the anchorage dataset.	33
Table 10	Selected optimal parameter values for the undersamplers.	42
Table 11	A comparison of the training dataset with the undersampled non-dragging class for each ML algorithm	43
Table 12	Cost-sensitive algorithms parameters properties.	48
Table 13	Experimental prediction performance of the selected undersampler-CSA models in comparison to the CSA.	49
Table 14	Risk ranking comparing ES, PARK, CoRI, and ML models.	58

List of Figures

Fig. 1	Flow diagram of the proposed framework for dragging risk prediction.	8
Fig. 2	Swinging ship motion at anchor.	10
Fig. 3	The AIS network. Source: (Bošnjak et al., 2012)	12
Fig. 4	The research study area. Ulsan port E-group anchorage area.	15
Fig. 5	Dredged sample of ocean floor showing muddy seabed in E-group anchorage. Source: (Lee, 2014).	16
Fig. 6	Ship trajectory at anchor. (a) A sample of a ship motion trajectory with a stable anchor that is not dragging, and (b) a sample of a ship motion trajectory with its anchor dragging.	18
Fig. 7	Imbalanced dataset classification problem from dataset complexity. Source:(Sakri and Basheer, 2023)	20
Fig. 8	Class distribution and class overlap of features in anchorage data: (a) ship length distribution, (b) ship draft distribution, (c) sea depth distribution, (d) wind speed distribution, (e) wind direction distribution, and (f) seabed type distribution.	32
Fig. 9	The positions of non-dragging anchor ships in Ulsan port E-group anchorage area.	33
Fig. 10	The positions of dragging anchor ships in Ulsan port E-group anchorage area.	34
Fig. 11	Distribution of the features for the anchorage dataset.	35
Fig. 12	Wind speed and direction for the non-dragging and dragging cases in the anchorage dataset.	36
Fig. 13	Wind speed and direction distribution by the month of the year.	37
Fig. 14	Anchor dragging direction compared to wind direction for the dragging samples of the anchorage dataset.	38
Fig. 15	Distribution of the features for the dragging samples.	39
Fig. 16	Process flow from data resampling, CSA training, and performance metrics-based model selection. The model naming scheme is a combination of undersampler and CSA initials. For example, RU-CLR denotes Random Undersampler - Cost-sensitive Logistics Regression.	41

Fig. 17 The scatter and distribution plots of the training dataset before and after resampling with cost-insensitive LR as the wrapper algorithm.	44
Fig. 18 The scatter and distribution plots of the training dataset before and after resampling with cost-insensitive SVM as the wrapper algorithm.	45
Fig. 19 The scatter and distribution plots of the training dataset before and after resampling with cost-insensitive XGB as the wrapper algorithm.	46
Fig. 20 The scatter and distribution plots of the training dataset before and after resampling with BBC as the wrapper algorithm.	47
Fig. 21 Performance comparison of the models.	50
Fig. 22 Confusion matrix for the undersampler-CSA and CSA models from a test dataset of 2,257 samples consisting of 18 dragging and 2,239 non-dragging samples.	51
Fig. 23 Case 1 trajectory plot (a) from AIS data in comparison to prediction plot (b), (c), and (d) from AKNN-CSVM, AKNN-CXGB, and RU-CSVM models, respectively.	54
Fig. 24 Case 2 trajectory plot (a) from AIS data in comparison to prediction plot (b), (c), and (d) from AKNN-CSVM, AKNN-CXGB, and RU-CSVM models, respectively.	55
Fig. 25 Case 3 trajectory plot (a) from AIS data in comparison to prediction plot (b), (c), and (d) from AKNN-CSVM, AKNN-CXGB, and RU-CSVM models, respectively.	56
Fig. 26 Case 1 trajectory plot (a) from AIS data in comparison to dragging risk plot (b), (c), and (d) from AKNN-CSVM, AKNN-CXGB, and RU-CSVM models, respectively.	59
Fig. 27 Case 2 trajectory plot (a) from AIS data in comparison to dragging risk plot (b), (c), and (d) from AKNN-CSVM, AKNN-CXGB, and RU-CSVM models, respectively.	60
Fig. 28 Case 3 trajectory plot (a) from AIS data in comparison to dragging risk plot (b), (c), and (d) from AKNN-CSVM, AKNN-CXGB, and RU-CSVM models, respectively.	61

Fig. 29 Real-time estimation of dragging risk for vessels at anchorage for three days using the RU-CSVM model.	62
Fig. 30 Monitoring the anchor dragging risk of tankers anchoring at Ulsan E-group anchorage area.	64
Fig. 31 Monitoring the anchor dragging risk of cargo and tanker ships anchoring at Ulsan E-group anchorage area.	65

List of Abbreviations

AKNN	all k-nearest neighbors editing
AIS	Automatic identification system
AUC	Area under the receiver operating characteristic curve
BBC	Balanced bagging classifier
CE	Cross entropy
CLR	Cost-sensitive logistic regression
CoRI	Collision risk
CSA	Cost-sensitive algorithm
CSAs	Cost-sensitive algorithms
CSVM	Cost-sensitive support vector machine
CXGB	Cost-sensitive extreme gradient boosting
ENC	Electronic navigation chart
ERM	Empirical risk minimization
ES	Environmental stress
ETA	Expected time of arrival
FN	False negative
FP	False positive
GM	Geometric mean
IMO	International Maritime Organization
LR	Logistics regression
M	Mud

ML	Machine learning
MMSI	Maritime mobile satellite identity
MSi	Mud over silt
NCR	Neighborhood cleaning rule
OOW	Officer on watch
OSS	One sided selection
PARK	Potential assessment of risk
R	Rocky
RENN	Repeated edited nearest neighbors
RFECV	Recursive feature elimination cross-validated
ROC	Receiver operating characteristic
RU	Random undersampler
SSh	Sand and shell
SiSh	Silt shells
StSi	Sticky silt
SJ	Subjective judgement
SOLAS	Safety of life at sea
SVM	Support vector machine
TN	True negative
TP	True positive
VHF	Very high frequency
VTSO	Vessel traffic service officer
VTS	Vessel traffic service
XGB	Extreme gradient boosting

Abstract

Heavy maritime traffic and the associated rise in vessel traffic density at anchorages have recently become prominent in maritime traffic safety. Anchoring operation is a complex task that necessitates experience, knowledge, and the capacity to anticipate all potential consequences of any decision. Despite the numerous publications available on anchoring practices, requirements, and procedures, there are still incidents of anchor dragging. Anchor dragging for ships at anchorage poses a serious risk to marine traffic as it may lead to collisions and the destruction of seabed infrastructure such as cables and pipelines. This study analyzed a substantial dataset of ships in anchorage areas. The goal was to develop a machine learning model for estimating the risk of anchor dragging among anchored ships using a binary classification system that distinguishes between non-dragging and dragging cases, thereby promoting maritime traffic and ship safety within harbors and anchorage zones.

Previous research studies conducted on dragging anchors were used to examine factors that cause the dragging of anchors at anchorage. The factors identified were; ship characteristics which consisted of ship length and draft, and hydro-meteorological factors which consisted of wind speed, direction of wind, depth of the sea, and nature of the seabed. The factors identified were used as input features for the anchorage dataset. The length and draft of the ship were extracted from historical automatic identification system, wind speed and direction were extracted from Korea Meteorological database, whereas the depth of the sea and nature of the seabed were extracted from hydrographic publications. The data were collated for each study case to form anchorage dataset.

A preliminary analysis of the anchorage dataset revealed a significant class imbalance ratio of 126:1 between the majority class, which consisted of non-dragging cases, and the minority class, which consisted of dragging cases, suggesting that the best machine learning strategy was majority-class undersampling in combination with cost-sensitive learning. Cost-sensitive learning is a type of machine learning where misclassification costs are taken into consideration to minimize costs. In this study, the cost was identified as the consequence of misclassification of a dragging candidate. Therefore, four cost-sensitive algorithms were

proposed. Five undersampling techniques were proposed and optimized to resample the majority class to increase the prediction accuracy of the cost-sensitive algorithms.

A total of twenty models were developed from the combination of five data undersampling techniques and four cost-sensitive algorithms. The prediction performance of the twenty models was compared to the cost-sensitive algorithms to highlight the importance of undersampling techniques. The ideal model was envisioned to be that which minimizes misclassification of the dragging cases and has a robust ability to distinguish the classes. The recall score is the measure of algorithm ability to minimize minority class misclassification, whereas the area under the receiver operating characteristic curve (AUC) score is the ability to distinguish the classes. Therefore, the model with the highest recall and AUC scores was selected as the best machine learning model for estimating anchor-dragging risk on a binary scale of dragging or non-dragging and risk rank profiling of the anchor situation. The all k-nearest neighbors editing undersampler combined with cost-sensitive support vector machine and random undersampler combined with cost-sensitive support vector machine outperformed the other models, with recall, geometric mean, and AUC scores of 0.944, 0.63, 0.684, and 0.889, 0.792, 0.824, respectively.

Case studies were conducted to demonstrate the prediction accuracy of the models. The case study showed a consistent prediction of the anchor situation in comparison to the anchor situation of the trajectory plot. Finally, the potential applications and improvements for the prediction model were highlighted.

Keywords: Anchor dragging, Undersampling, Cost-sensitive learning, Binary scale, Machine learning, Misclassification cost

Abstract in Korean

해상 교통이 증가함에 따른 정박지에서 선박 교통 밀도 증가는 최근 해양 교통 안전에 서 주목을 받고 있습니다. 묘박은 경험, 지식, 그리고 어떤 결정의 모든 잠재적 결과를 예측할 수 있는 역량을 필요로 하는 복잡한 작업입니다. 투묘 방법, 요구 사항 및 절차에 대한 수많은 간행물이 있음에도 불구하고, 주요는 여전히 많이 발생하고 있습니다. 주요 현상은 케이블과 파이프라인과 같은 해저 인프라의 충돌과 파괴로 이어질 수 있기 때문에 해상 교통에 심각한 위험을 초래합니다. 본 연구에서는 묘박지에 있는 선박의 데이터를 분석했습니다. 본 연구의 목표는 항만 및 묘박지 내의 해상 교통 및 선박 안전을 증진시키기 위해 묘박중인 선박의 주요 현상과 정상투묘를 구분하는 이중 분류 시스템을 이용하여 기계 학습(ML) 모델을 개발하는 것입니다.

선행 연구를 통해 주요를 유발하는 요인을 분석하였습니다. 식별된 요인은 선박의 길이와 흘수와 같은 선박 특성과 풍속, 풍향, 수심 그리고 해저 저질과 같은 해양 특성 요인이었습니다. 식별된 요인들은 묘박지 데이터의 특징으로 사용되었습니다. 선박 자동 식별 장치에서 배의 길이와 흘수를 추출하였으며, 풍속과 풍향은 한국 기상 데이터베이스에서, 수심과 해저 특성은 해양 간행물로부터 추출했습니다. 각 연구 사례에 대한 데이터는 앵커리지 데이터 세트를 형성하기 위해 수집되었습니다.

묘박 데이터셋의 예비 분석 결과 주요로 구성된 대다수 클래스와 정상투묘로 구성된 소수 클래스 간의 불균형 비율은 133:1로 유의하게 나타났습니다. 이는 cost-sensitive learning과 결합된 다수 계급의 언더샘플링이 최고의 ML 전략임을 나타냅니다. Cost-sensitive learning은 비용을 최소화하는 목표로 오분류 비용을 고려하는 기계 학습 유형입니다. 이 연구에서 비용은 주요 후보의 잘못된 분류의 결과로 식별되었습니다. 따라서 네 가지 cost-sensitive learning 알고리즘이 제안되었습니다. Cost-sensitive learning 알고리즘의 예측 정확도를 높이기 위해 다수 클래스를 다시 샘플링하기 위해 5가지의 언더샘플링 기술이 제안했습니다. 5개의 데이터 언더샘플링 기술과 4개의 cost-sensitive learning 알고리즘을 결합하여 총 20개의 모델이 개발했습니다. 20개의 모델의 예측 성능을 cost-sensitive learning 알고리즘과 비교하여 언더샘플링 기법의 중요성을 강조했습니다. 이상적인 모델은 주요 케이스의 잘못된 분류를 최소화하고 클래스를 식별할 수 있다. 리콜 점수는 소수 클래스 오분류를 최소화하는 알고리즘 능력의 척도이며, 수신기 작동 특성

곡선 (AUC) 점수 아래의 영역은 클래스를 구별하는 능력입니다. 따라서, 가장 높은 리콜 및 AUC 점수를 가진 모델을 주요 또는 정상투묘의 척도로 주요 위험을 예측하는 최상의 ML 모델로 선정하였습니다. cost-sensitive support vector machine과 random undersampler cost-sensitive support vector machine을 결합한 모든 k-nearest neighbors editing undersampler는 리콜, GM 및 AUC 점수가 각각 0.944, 0.63, 0.684 및 0.889, 0.792, 0.824로 다른 모델보다 높았습니다.

모델의 예측 정확성을 증명하기 위해 사례 연구를 수행하였습니다. 사례 연구는 항적의 투묘 상황과 비교하여 투묘상황의 일관된 예측을 보여주었습니다. 마지막으로, 예측 모델에 대한 잠재적 응용 프로그램과 개선이 강조되었습니다.

키워드: 주요, Undersampling, Cost-sensitive learning, 바이너리 스케일, 기계 학습, 오분류 비용

1 Introduction

Maritime transport accounts for 90% of the world trade by volume (Chen et al., 2019). Indeed, the economies and essential supply chains of numerous countries were sustained throughout the COVID-19 epidemic largely through the utilization of maritime transportation, which proved to be highly efficient in facilitating the movement of enormous quantities of goods. Despite the implementation of maritime traffic safety-enhancing measures aimed at increasing the degree of navigation safety, marine accidents continue to be a significant cause for concern due to the frequent occurrence of various incidents and their consequential disastrous outcomes in recent years. Maritime accidents are widely recognized as outcomes stemming from a combination of intricate technical, human, organizational, and environmental factors.

The expanding number of trade routes and the rising demand of global business for seaborne shipping has led to an increase in ship traffic, which strains the limited port infrastructure due to traffic congestion, hence making the effective and safe operation of anchorages a critical task. Heavy maritime traffic, combined with increased ship density in anchorages, has recently emerged as a major issue in maritime traffic safety. Anchorages are an effective technique for reducing maritime traffic congestion and enhancing the overall quality of maritime transportation. In addition to reducing maritime congestion by serving as a temporary holding area, anchorages provide vital services to vessels, such as land services (fueling, legal concerns, repairs, etc.), cargo loading and unloading, and shelter from adverse weather. Thus, anchorages serve a similar purpose for vessels as car parks do for automobiles and trucks. As a result, many nations have set aside a large portion of the ocean as anchorages (Oz et al., 2015).

Anchor dragging poses a significant risk to maritime traffic due to its potential to result in several adverse events. These include collisions with vessels in transit, collisions with nearby vessels at anchor, the grounding of ships as a consequence of drifting caused by anchor dragging, and the destruction of underwater infrastructure such as pipelines and cables (Mulyadi et al., 2014; Yoon and Na, 2013). Anchorage accidents are extremely disruptive to marine traffic since they not only pose the risk of human injury, environmental degradation, and vessel damage but also render anchorages largely inaccessible until the disaster is cleared.

Dragging anchor accidents are caused by not only natural disasters such as bad weather and typhoon but also by human error (Rutkowski, 2019) because ships at anchorage are always under the supervision of an officer on watch (OOW) and are monitored by a vessel traffic service officer (VTSO) (Lee and Song, 2018). As a result, it is critical to conduct research on the characteristics that cause the dragging of anchors in order to predict and prevent a vessel on anchor from dragging.

This study introduces a machine learning (ML) approach to design a dragging risk model from a combination of cost-sensitive learning and data undersampling techniques thereby promoting maritime traffic and ship safety within port harbors and anchorage areas. The proposed approach will be the first to apply ML to predict anchor dragging risk at the time of publication of the study. Current studies propose dragging risk assessment computed from numerical and simulation models. This study offers a departure from numerical modeling of dragging risk assessment to the newly-embraced ML approach in the digital technology revolution under the content of shipping 4.0 and the maritime internet of things (Aiello et al., 2020).

Section 2 provides a detailed description of current research on anchor operation, equipment design, and dragging prediction models. Section 3 illustrates an overview of the proposed ML approach using a flow chart and a detailed description of the dynamic behavior of the ship when at anchor. Specifically, the interaction of the environmental factors with the ship and the effect on anchor-holding power. The factors that cause dragging of anchor are expounded. The factors identified are used as input features to the anchorage dataset. The data are extracted from historical AIS, hydrographic publications, and meteorological databases.

Section 4 introduces the problems associated with machine learning from an imbalanced dataset. Subsection 4.2 defines cost-sensitive learning, introduces the theoretical background, and describes associated applications in other disciplinary that are relevant to this study approach. Moreover, a method of converting standard ML algorithms to cost-sensitive algorithm (CSA) is explained. Subection 4.3 introduces literature and theoretical background of dataset resampling. The section dwells on undersampling procedures. Subsection 4.4 describes performance metrics that are suitable to measure the prediction performance of the proposed ML models.

Section 5 and 6 describes experimental results for the proposed models in comparison to the cost-sensitive algorithms (CSAs), in addition to a case study to demonstrate excellent prediction performance of the selected model. Section 7 provides a summary of the study and highlights the contributions of the study. Subsection 7.2 identifies shortcomings of the proposed ML approach and highlights suggestions for future research to improve prediction performance.

2 Literature Review

Previous academic research on dragging anchors has focused primarily on factors such as bad weather, anchor design, seabed composition, and anchor trajectory kinematics in an attempt to identify the root cause of dragging anchors. Anchor trajectory kinematics involves numerical analysis of the interaction of the anchor chain, anchor, and vessel as a system to describe the behavior of the anchor in different seabed types. The central objective is to model the behavior of the anchor and the chain on different seabed types to establish the risk of dragging the anchor. [Zhuang et al. \(2021\)](#) developed a mathematical model that can predict the dragging trajectory to determine the embedding depth of anchors by analyzing the kinematic interaction of the chain and anchor in clay soil seabed. The study investigated the effect of soil strength and anchor shank/flute pivot angle on the embedment depth of the dragging anchor when carrying out the anchoring process. [Liu et al. \(2012\)](#) developed a theoretical framework, kinematic model and drag equations to describe the embedment of dragging anchors in different soil types. The framework assumed that the anchor, chain herein referred to as dragline, and the vessel operate as one interactive system. [Gao et al. \(2016\)](#) proposed a prediction process for the penetration depth of dropped anchors from a combination of experimental, numerical, and theoretical analysis. The experimental test analysis involved dropping the anchor from a pre-determined height to determine the influence of anchor weight on penetration depth. A finite element method and theoretical analysis were developed and performance was compared to the tests to validate the prediction process.

The research on anchor trajectory kinematics is primarily focused on the mathematical modeling of anchor-chain-seabed interaction, which defines anchor-holding power, to determine the anchor penetration depth. The studies are of significance in recommending the safest subsea infrastructure burial depth or the best cover material for the exposed seabed infrastructure.

Bad weather from typhoons and rough waves causes a significant risk of dragging anchors. The focal point in most studies on bad weather is to establish the relationship between strong winds or waves and the risk of dragging anchor for a vessel in anchorage. [Sasa and Incecik \(2012\)](#) evaluated ship motions caused by storm-generated wave and wind forces and developed a numerical simulation to detect the stranding of vessels caused by dragging

anchors in rough weather. A questionnaire survey on the dangers of harbor refuge collected oceanographic conditions and ship motions information that aided in modeling the numerical simulation. [Lee et al. \(2022\)](#) developed a risk assessment technique for anchored ships to predict stranding or collision from dragging anchor by numerical simulation of anchored ship motions in open-sea rough waves. The stranding risk was modeled from the relationship between vertical ship motion and under-keel clearance, whereas, collision risk was modeled from the closest point of approach, and the ship domain overlapping index. [Jung et al. \(2009\)](#) carried out sea trials to establish the dragging starting point by comparing the critical external forces (wind force, frictional resistance, drifting force, and ship motion moment), with the anchor holding power. The study also established that a unique trajectory pattern and heading angle are exhibited by a dragging anchor. [Kim et al. \(2022\)](#) carried out practical trials to determine the characteristics of anchor-holding power due to the nature of the seabed in the presence of strong winds and heavy currents. The holding power was measured as tension values of the anchor chain. [Kang et al. \(2021\)](#) recommended an anchorage safety management of ships at anchor in Jinhae typhoon refuge bay by determining the minimum wind speed required to cause dragging for ships of different tonnage under varied sea depth, anchor type, and anchor chain pay-out.

The primary goal of the study on adverse weather conditions is to determine the specific thresholds of meteorological features, such as wind speed, wave heights, and current, that can cause anchor dragging, to recommend the study threshold as anchorage safety guidelines for ships of various sizes.

Various studies have developed dragging risk prediction assessment programs and models. [Kim et al. \(2018\)](#) proposed a dragging risk assessment program that generates an alarm; "warning" for high dragging risk, and "safe" for low dragging risk. The program input data included the ship basic particulars, anchoring condition, and external environment for computing the wind pressure, frictional force, drift force, and holding power. A comparison of total external forces and anchor holding power was used to weigh the risk decision. The inputs were manually inputted into the assessment program. [Okazaki and Hirai \(2011\)](#) developed a decision support system that predicts the dragging anchor phenomenon by comparing the program-simulated anchor cable tension with the anchor holding power. A dragging anchor was detected when the simulated anchor cable tension value, computed from ship motion

analysis, was greater than the anchor holding power. The input to the decision support system was information obtained from the bridge such as; ship length, breadth, draft, displacement, the weight of the anchor, weight of the chain, etc., that are used to simulate the anchor cable tension value. The proposed risk assessment programs cannot be used for real-time assessment of dragging risk as the inputs are manually inputted into the programs.

A summary of studies applied by various authors to improve anchorage safety is summarized in [Table. 1](#). The reviewed studies have focused on numerical analysis, computational methods, mathematical modeling, and simulations to describe the behavior of ship anchorage systems in the estimation of anchor dragging risk. Furthermore, there is insufficient studies regarding the application of historical anchorage data to predict the dragging risk through implementing ML algorithms. Thus, the goal of this study was to develop a ML model that can predict the dragging risk of a ship at anchorage, thereby promoting maritime traffic and ship safety within harbors and anchorage zones. This study used anchorage data samples created by aggregating historical automatic identification system (AIS) and hydro-meteorological data. A ML approach was developed that combines optimized data undersampling techniques to address class imbalance distribution together with a CSA that is optimized to minimize misclassification costs. The best scoring model was selected using performance metrics such as recall, specificity, geometric mean (GM), and AUC. A case study was conducted to demonstrate the prediction accuracy and robustness of the selected model.

Table 1: A summary of the studies on anchor dragging by various authors.

Authors	Features investigated	Objectives	Methods	Outcome
Zhuang et al. (2021)	Chain, anchor, and clay seabed	Predict the dragging trajectory of the anchor to determine embedding depth.	Variation of shank/flute pivot angle. Effect of clay soil strength.	Mathematical model.
Liu et al. (2012)	Anchor, chain, and the anchor handling vessel	Theoretical modeling of drag embedment problem of the anchor in seabed soils.	Kinematic modeling	Kinematic equations. Mathematical model.
Gao et al. (2016)	Anchor, chain, and the soil	Penetration depth of dropped anchors.	Experiments, finite element method, and theoretical analysis.	Semi-empirical model.
Sasa and Incecik (2012)	Storm-generated wind and waves	Simulation model to reproduce the stranding of anchored ships.	Questionnaire survey. Numerical simulations.	Simulation model
Lee et al. (2022)	Open-sea rough waves	Risk assessment model	Ship motion numerical simulation. Risk modelling	Stranding risk assessment. Collision risk assessment.
Jung et al. (2009)	Wind and waves	Critical dragging threshold.	Sea trials, numerical ship motion analysis.	Dragging and heading angle pattern.
Kim et al. (2022)	Wind, current, and seabed.	Characteristic of holding power on different seabed types.	Sea trials.	Anchor holding power model.
Kang et al. (2021)	Wind speed	Minimum wind speed to cause dragging.	Sea trials. Ship motion analysis.	Minimum wind speed.
Kim et al. (2018)	Wind speed, current, anchor, chain, and ship particulars	Develop a dragging risk assessment program	numerical ship motion analysis.	Risk assessment program.
Okazaki and Hirai (2011)	ship particulars, meteorological data, anchor and anchor chain and wave	Develop a decision support system that predicts a dragging anchor phenomenon.	Anchor holding power analysis. Ship motion simulation	Decision support program.

3 Anchorage dataset extraction

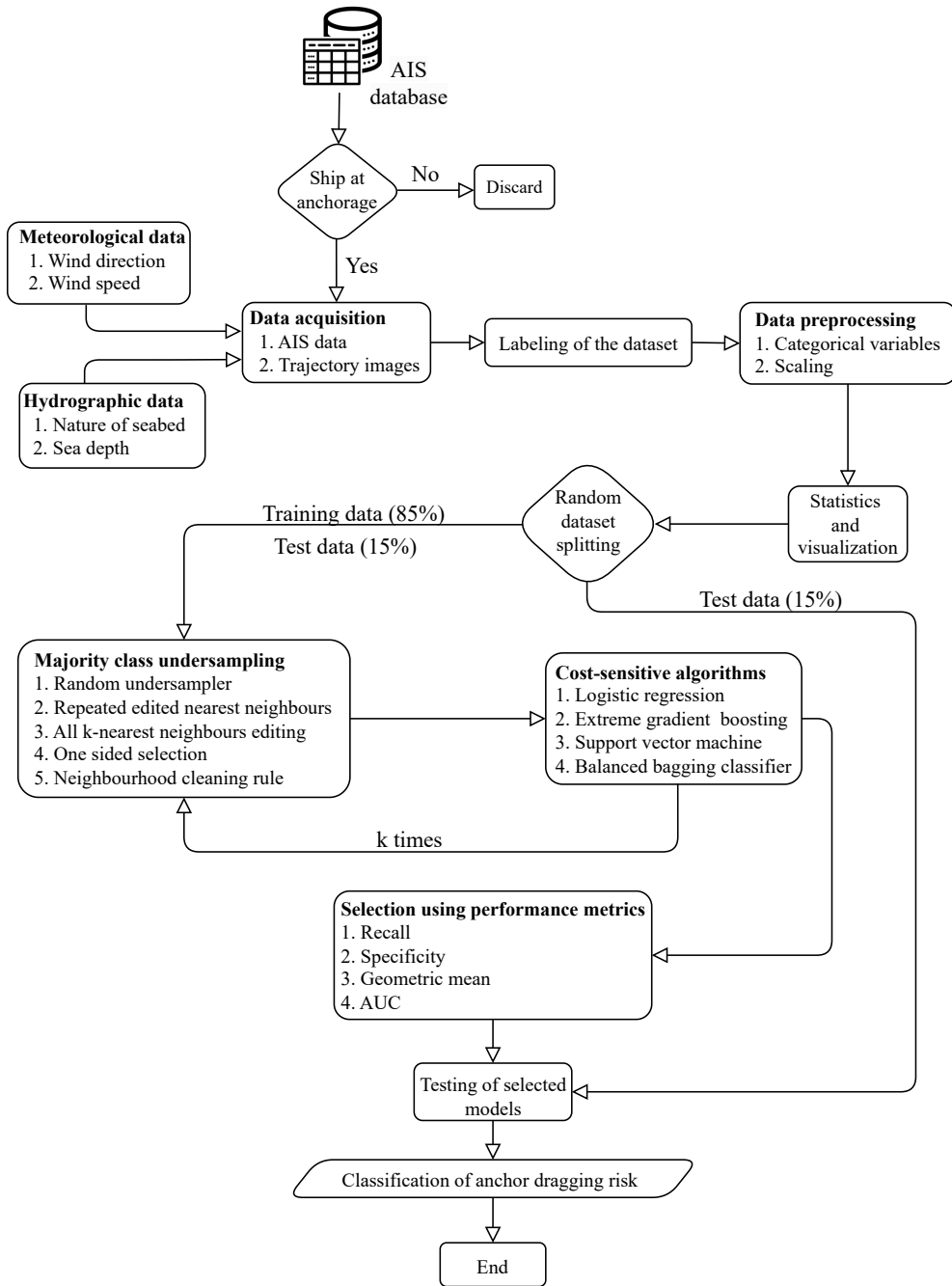


Fig. 1: Flow diagram of the proposed framework for dragging risk prediction.

Fig. 1 depicts the framework of the proposed study. First, the designated anchorage area was identified. AIS and hydro-meteorological data were extracted for any ship in the anchorage area. The AIS data was used to extract meteorological and hydrographic data for the ship at anchor, in addition to ship characteristics data. Ship trajectories while at anchor were captured as images and subsequently used to label the data. A one-hot encoder encoded the categorical features in the dataset. Statistical analysis, data visualization, and data preprocessing were carried out in preparation for the undersampling and subsequent training of the CSA. The data set was split using stratified random sampling to preserve data distribution in the training and test sets. The majority class samples were undersampled using five undersamplers. The output data from the undersampling was the input to the four CSAs. The best combination of the undersampler and CSA model was selected based on the highest scores achieved for recall, specificity, GM, and AUC.

3.1 Behavioral characteristics of ship at anchorage

Anchoring operation is a complex task that necessitate experience, knowledge, and the capacity to anticipate all potential consequences of any decision. The operation is affected by the direction and strength of the wind (Kang et al., 2021), wave and current (Jung et al., 2009; Sasa and Incecik, 2012), maneuvering room for approach (Kray, 1973), swinging room after anchoring, water depth, nature of seabed (Gao et al., 2016; Kim et al., 2022; Liu et al., 2012), anchoring chain length (Liu et al., 2019; Okazaki and Hirai, 2011), location of shoals or hazards such as submarine cables and other obstacles (Green and Brooks, 2011; Yoon and Na, 2013), and conditions affecting visibility. Considering these factors, anchoring operations must be meticulously planned and preceded by a thorough investigation of the conditions that affect anchoring. Anchors are designed to work in normal to optimal conditions.

In strong winds, an anchored vessel will yaw around the anchored position from point A to point G, drawing a figure-eight as illustrated in Fig. 2. At higher wind speeds, the yaw swiftly surges from one extremity to the other. This may place a shock load on the anchor cable, thereby causing the anchor to lose its hold on the seabed, and increasing the risk of anchor dragging. The anchor is dragged when the tension in the anchor cable exceeds the combined holding power of the anchor and cable (Kim et al., 2022). An anchor provides the maximum

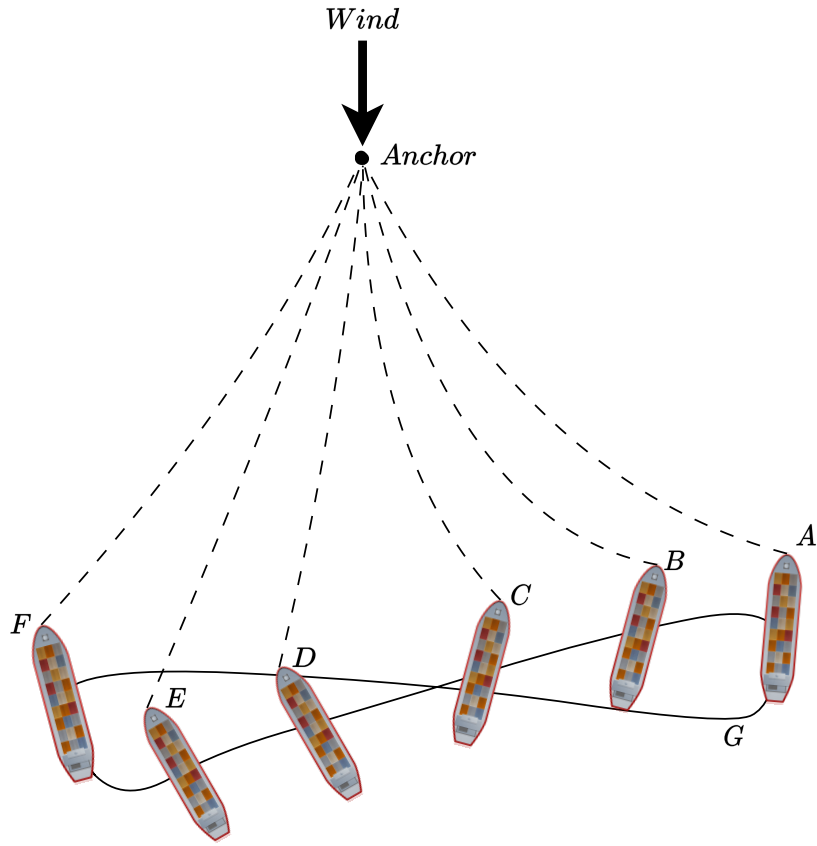


Fig. 2: Swinging ship motion at anchor.

holding power when its flukes are fully embedded in the seabed. This occurs when the anchor shank rests on the seabed and the anchor cable is pulled in the horizontal direction. An increase in cable tension causes the cable to rise, creating an angle between the cable and the seabed. As the angle increases, the total holding power of the anchor decreases.

The anchor holding power of the ship is heavily reliant on the nature of the seabed, external forces, anchor design, and cable length. Based on the above-mentioned characteristics of ships at anchor, factors that have a big effect on the risk of anchor dragging were identified through the analysis of previous research and are used as input features. The factors were derived from ship characteristics and hydro-meteorological data. The input variables from the ship characteristics included the ship length and draft, whereas the hydro-meteorological factors included the water depth, wind speed, wind direction, and nature of the seabed. The interaction between the length of the ship and wind speed and direction

results in the generation of a wind resistance force, which has a significant influence as an external force that contributes to dragging risk (Okazaki and Hirai, 2011). The nature of the seabed determines the anchor holding power, whereas the ship draft indicates the loading state of the ship at anchor. The input variable data sources are publicly accessible and are derived from AIS and hydro-meteorological publications.

3.2 Features of an AIS data

The AIS is a communication system that operates on the Very High Frequency (VHF) radio spectrum. It facilitates the transmission of data packets through the VHF data link, allowing vessels and shore-based stations equipped with AIS to exchange identification information. This information can then be displayed on electronic charts, computer screens, or compatible radar systems. AIS, as the primary source of ship data, has often been used for ship collision modeling, navigation risk assessment, and traffic management, to enhance navigation safety (Yang et al., 2019). AIS can also serve as a valuable tool for navigation assistance, as it offers precise location data and supplementary information on buoys and lights. Fig. 3 illustrates the AIS communication network between the ships, base stations, VTS control centers, and navigation buoys.

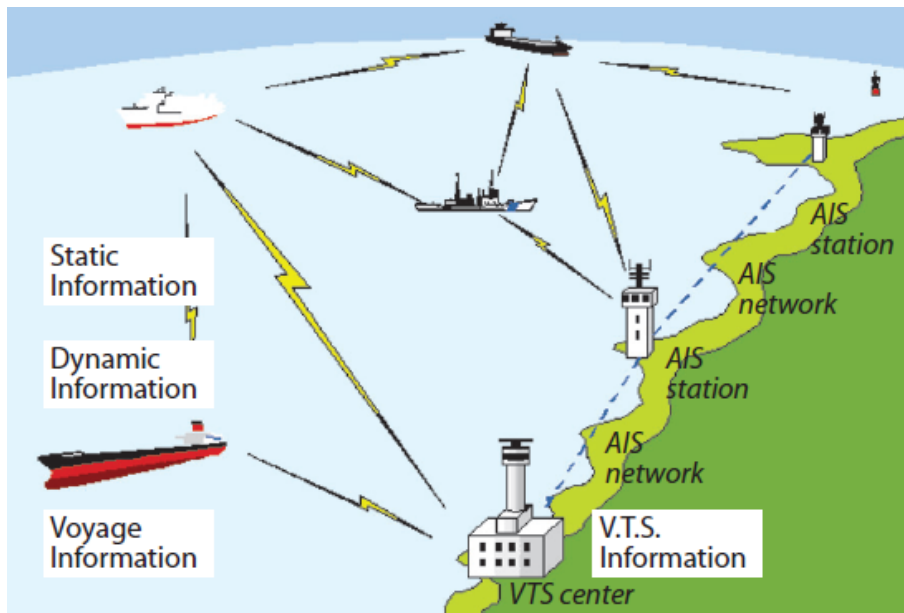


Fig. 3: The AIS network. Source: (Bošnjak et al., 2012)

The mandatory installation of an AIS generates a large amount of ship trajectory data around the world. As established by the International Maritime Organization Convention for the Safety of Life at Sea (SOLAS), an AIS transponder must be on board all ships with a gross tonnage of 300 or more that sail in international waters, ships with a gross tonnage of 500 tons or more that do not sail in international waters, fishing vessels with a length greater than 15m, and most commercial vessels such as cargo, tankers, and passenger ships of any size (IMO, 2014). In addition, the regulation requires that AIS shall: provide information including the

ship identity, type, position, course, speed, navigational status and other safety-related information automatically to appropriately equipped shore stations, other ships, and aircraft; receive automatically such information from similarly fitted ships, monitor and track ships, and exchange data with shore-based facilities. AIS technology is a system designed to transmit messages containing dynamic, static, and voyage information from vessels, in addition to its capability to receive messages from both base stations and other ships. Dynamic information, such as vessel position, speed, course, heading, navigation status, and time in coordinated universal time, is broadcast every 3 minutes for a vessel at anchor as illustrated in [Table 2](#).

Table 2: Transmission frequency of AIS messages in relation to dynamic information.

Navigation Status	Ship Speed	Transmission Period
At Anchor or Moored	<3 knots	3 min
	>3 knots	10 s
Cruising	0 - 14 knots	10 s
	0 - 14 knots and changing course	3.3 s
	14 - 23 knots	6 s
	14 - 23 knots and changing course	2 s
	>23 knots	2 s

Static information such as vessel identity and dimensions are transmitted every 6 minutes, as is voyage-related data such as vessel destination, hazardous nature of its cargo, and ship draft. Dynamic data are automatically updated via various ship sensors, whereas voyage-related data are manually input during the voyage. Static data are input into the AIS during installation and only need to be changed if the ship type changes owing to a major conversion or if the ship name or call sign changes ([Harati-Mokhtari et al., 2007](#)). [Table 3](#) is a list of broadcast AIS data types and their classification. An ever-growing amount of maritime traffic data is sent by AIS receivers through terrestrial and satellite networks. This can improve general maritime situational awareness in harbors, along the coast, and in open waters ([Forti et al., 2019](#)). The vessel traffic service (VTS) uses the AIS to ensure navigation safety, protect the environment, and optimize navigation efficiency. It accomplishes these objectives by actively monitoring marine traffic to prevent ship collisions, facilitating the sharing of information, and serving as a traffic management tool.

Table 3: AIS broadcast data.

Static data	Antenna position
	MMSI number (Maritime Mobile Satellite Identity-MMSI)
	Ship call sign and name
	IMO number
	Ship length and beam
	Ship type
Dynamic data	Ship position (latitude and longitude) with indication of accuracy (automatic updating using DGPS (eng. Differential Global Positioning System) sensor connected to AIS)
	Time in UT (eng. Universal Time - Central European Time)
	Speed over ground
	Course over ground
	True heading
	Rate of turn
	Navigational status (self-propelled, anchored, unable to maneuver, of a restricted ability to maneuver, moored, restricted by draught, aground, engaged in fishing or sailing boat)
	Angular velocity
Voyage related data	Ship draught
	Dangerous cargo
	Port of destination and ETA
	Passage plan
VTS data	Short information related to safety warnings and information on areas with navigational warnings and other dangers. The data should be addressed to AIS receiver for all ships and coastal stations within range.

3.3 Research study area

The study area is Ulsan port anchorage area located on the southeast coast of the Republic of Korea, bounded by the coordinates $35^{\circ}28'N$, $129^{\circ}22'E$ and $35^{\circ}26'N$, $129^{\circ}29'E$, as shown in Fig. 4. The port of Ulsan has played the leading role in the development of economic and industrial growth with the help of geographical location and port infrastructure. The port serves as the largest hub for the transport of liquid cargoes. The extensive investment in port infrastructure and the relevant handling and storage facilities has positioned the region as a prominent crude oil cargo port in Asia through the Far East oil hub business.

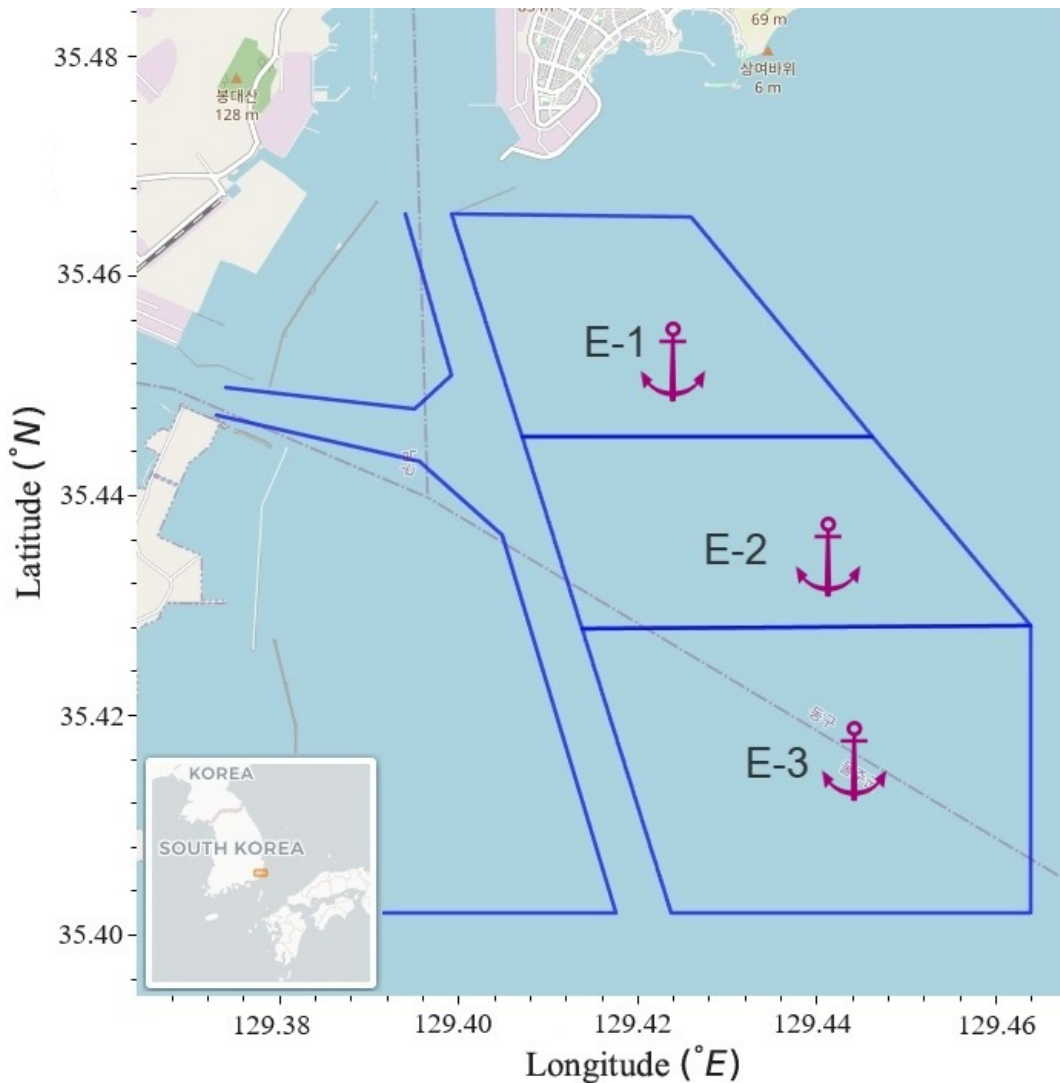


Fig. 4: The research study area. Ulsan port E-group anchorage area.

The Ulsan port features several anchorages, including E, W, M, and T anchorages, which have been designed to accommodate varying sizes of ships, taking into consideration natural

environmental factors such as depth and wave condition. The port complex in Ulsan consists of several distinct ports, namely the main port of Ulsan, Onsan Port, Ulsan New Port, and Mipo Port. These ports are categorized based on their geographical location and principal operational purpose. However, for this paper, the investigation will not include W and T anchorages due to their limited usage. W anchorage is infrequently utilized, while T anchorage is exclusively employed by vessels that call at Mipo port. The capacity and condition of E-group anchorage in Ulsan port are highlighted in Table 4. E-1, E-2, and E-3 anchorages are the main anchorage

Table 4: Capacity, area, seabed condition and depth of E-group anchorage in Ulsan port.
Source: (Park et al., 2016)

Anchorage	Capacity	Area	Seabed condition	Average depth
E-1	10,000 GT	9.34 km^2	Mud	40 m
E-2	30,000 GT	10.21 km^2	Mud	50 m
E-3	150,000 GT	11.76 km^2	Mud	60 m

area in Ulsan port, adjacent to the right side of the fairway. It is easy for ships calling at Ulsan port to proceed to the fairway after anchoring at the anchorage. The E-3 anchorage is the largest of the E-group anchorage area by area, depth, and vessel handling capacity. The seafloor is predominantly covered by mud, as shown by a sampled seafloor dredge in Fig. 5.

As the premier liquid cargo handling port, Ulsan is susceptible to environmental catastrophe in the event of an anchor-related incident. This study contributes to the improvement of anchorage safety in the Ulsan port E-group anchorage area through the development of a machine learning model that estimates the anchor dragging risk for a ship at anchor.



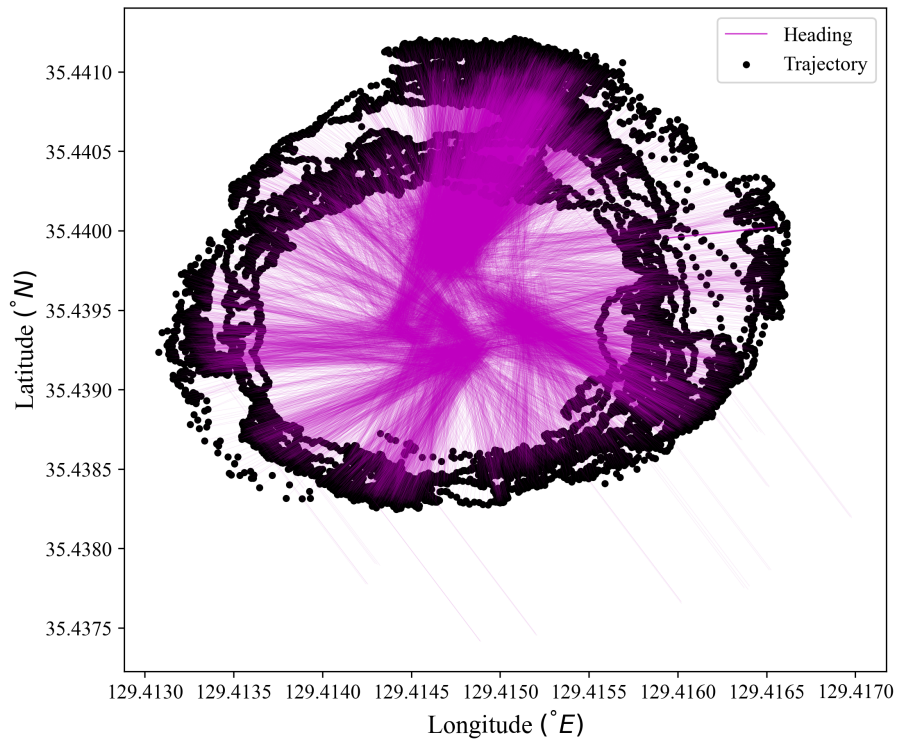
Fig. 5: Dredged sample of ocean floor showing muddy seabed in E-group anchorage.
Source: (Lee, 2014).

3.4 Anchorage dataset extraction

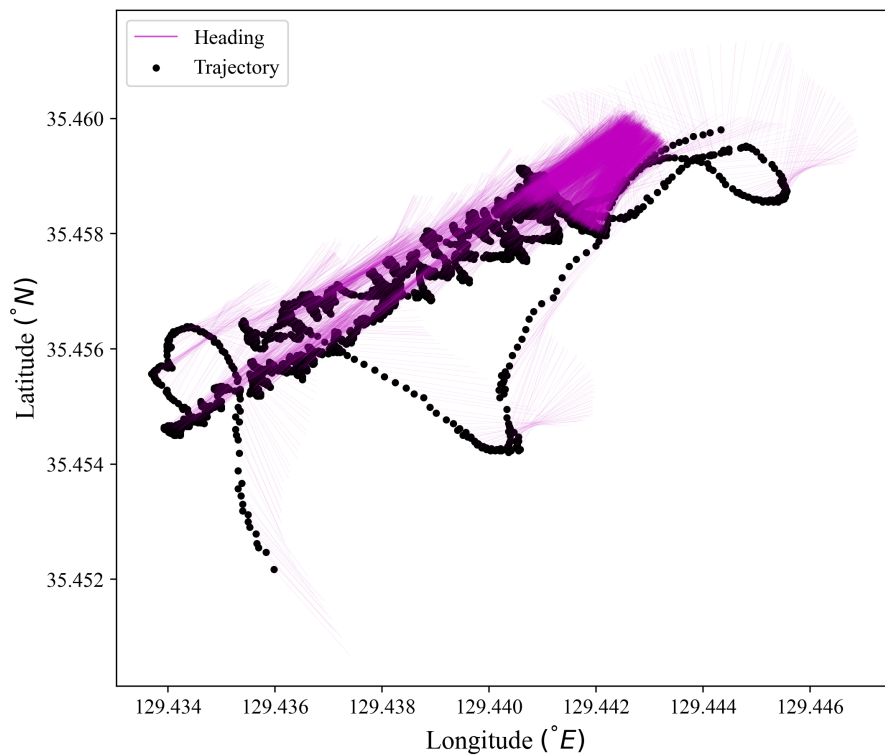
Visual observation (Fuji, 1977), radar observation (Yao et al., 2010), and simulation models (Hara and Nakamura, 1995) are generally used to describe marine traffic behavior. However, obtaining accurate data and understanding the true behavior of ships is impossible with the aforementioned methods. Consequently, the behavior of the anchored ships has been poorly understood (Gao and Makino, 2017). AIS data quantitatively depicts the accurate behavior of ships at anchor. The length and draft information of the ship were obtained from the AIS database, whereas the sea depth and nature of the seabed were obtained from the Korean Hydrographic and Oceanographic Agency publications. Additionally, the wind direction and speed were acquired from the Korea Meteorological Administration database.

The initial step in obtaining the features was to define the target anchorage area by specifying its positional coordinates. Vessels at anchor were identified by applying a speed threshold of 1 knot to discriminate between stationary vessels from underway vessels. A ship length threshold was applied to select vessels with a length over 60m. The AIS data for each detected vessel was extracted from the SQL database, and the values of ship length and draft were extracted. The trajectories were plotted and saved as image files for labeling the data samples. Fig. 6(a) and Fig. 6(b) show examples of images of anchor trajectory patterns extracted for non-dragging and dragging ships at anchor, respectively, to be used for labeling of samples. The sea depth value and nature of the seabed were obtained by selecting the Electronic Navigation Chart (ENC) values that had the closest match to the AIS positional 170 coordinates.

Collation of wind direction and wind speed values involved matching the AIS datetime with the meteorological database datetime. The meteorological data point with the lowest time discrepancy from the AIS data was selected. Thereafter, ship length and draft from the AIS, wind speed and direction values from the meteorological database, and sea depth and seabed type from the ENC chart for all the trajectory points for a ship at anchor were collated for each trajectory point. A sample consisting of the ship length, ship draft, depth of the sea, nature of the seabed, maximum wind speed and corresponding wind direction for the period at anchor, and associated labels, for each vessel at anchor was extracted.



(a)



(b)

Fig. 6: Ship trajectory at anchor. **(a)** A sample of a ship motion trajectory with a stable anchor that is not dragging, and **(b)** a sample of a ship motion trajectory with its anchor dragging.

The data points formed a set of n samples represented by $G = \{(\mathbf{x}_i, y_i), (\mathbf{x}_{i+1}, y_{i+1}), \dots, (\mathbf{x}_n, y_n)\}$, in which \mathbf{x}_i is vector data in the samples expressed as $\mathbf{x}_i = (ln_i, dr_i, dt_i, wv_i, wd_i, s_i)$, where n is the number of data samples, $i = 1, 2, \dots, n$ are real and positive integers, ln_i is the ship length, dr_i is ship draft, dt_i is the depth of the sea, wv_i is wind speed, wd_i is wind direction, s_i is the nature of the seabed, and y_i is the class label associated with the vector data. Table 5 lists the data type, composition, and attributes of the anchorage dataset.

Table 5: Constitution of the anchorage dataset.

Variable type		Variable name	Attributes
Input	Continuous	Ship length	The length of the ship in meters (m)
		Ship draft	The distance from the keel to the waterline in meters (m)
		Depth of the sea	Distance from water surface to ocean floor in meters (m)
		Wind speed	Wind velocity in meters per second (m/s)
Categorical	Categorical	Wind direction	1: N, 2: NE, 3: E, 4: SE, 5: S, 6: SW, 7: W, 8: NW
		Seabed type	0: Mud, 1: Mud over silt, 2: Sand and shell, 3: Rocky, 4: Silt shells, 5: Sticky silt
Target	Categorical	Data point label	Binary class label, -1: non-dragging, 1: dragging

4 Cost-sensitive learning on imbalanced data

4.1 Class imbalance problem

A dataset comprising 15,042 samples was gathered from the anchorage area of the Ulsan port in the Republic of Korea. The samples were obtained from 2017 to 2021 and exhibited an intrinsic class distribution (Chawla et al., 2004) of 118 samples belonging to the minority class, which represented the dragging cases, and 14,924 samples belonging to the majority class, which were classified as non-dragging cases. The non-dragging cases exceeded the dragging cases by a ratio of 126 to 1. By definition (He and Garcia, 2009; Krawczyk, 2016), this dataset qualified as an imbalanced dataset, which would pose class imbalance problems when training the ML algorithms. The problems associated with imbalanced datasets are low training data density, particularly if the minority class knowledge is of interest, class overlap, small disjuncts within the dataset, and the presence of noisy and borderline samples in the dataset. Fig. 7 is a sample two-feature dataset showing the data complexity problems associated with an imbalanced dataset.

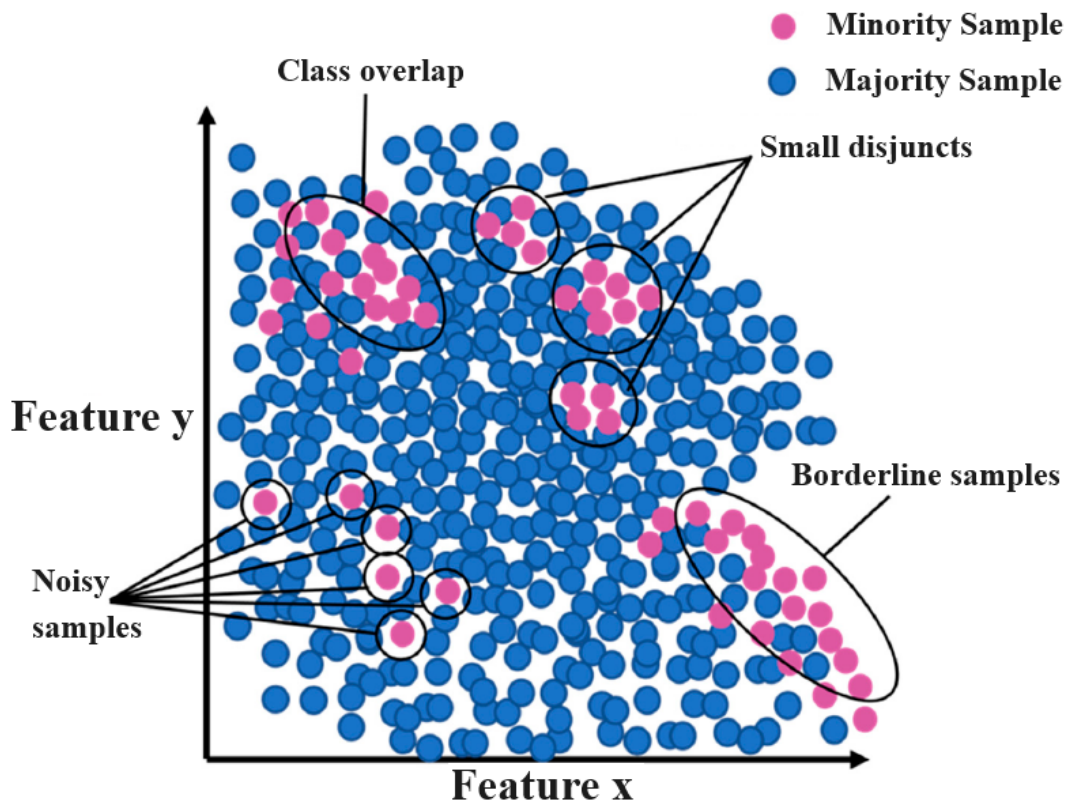


Fig. 7: Imbalanced dataset classification problem from dataset complexity. Source:(Sakri and Basheer, 2023)

A standard algorithm employed in binary classification problems typically assumes that the classes are evenly distributed. Consequently, when training a model with highly imbalanced data, the model tends to favor the majority class, which compromises its performance, particularly when the minority class is of primary importance (Yang and Wu, 2006). In addition, the algorithm assumes an equal penalty for both false negative and false positive misclassification errors. Nonetheless, this assumption presents a drawback in scenarios involving imbalanced classification tasks, as the consequences of misclassifying a positive instance (minority class) are more severe than those of misclassifying a negative instance (Zhang and Zhang, 2016).

Class overlap makes it difficult to apply separation rules and to categorize samples from minority classes. Simple classifiers can learn to classify correctly when there is no class overlap in the dataset. Small disjunctions develop when the concept of the minority group incorporates auxiliary ideas, which further confuses the classifier. A classifier may dismiss disjuncted clusters of minority samples as noise when it encounters them during learning (Krawczyk, 2016).

Borderline samples are discovered where the majority and minority classes intersect. ML algorithms have difficulty learning noisy and ambiguous samples. To improve prediction accuracy, the classifiers seek to discover the most precise boundary of each class during training. Borderline samples carry a greater risk of misdiagnosis in comparison to nonborderline samples; thus, their classification becomes more critical (Sakri and Basheer, 2023).

Researchers devised several strategies to address class imbalance problems, as outlined in the study by (Chawla et al., 2004). These strategies primarily involve resampling data (Liu et al., 2008; Nickerson et al., 2001), modifying and adapting algorithms (Kong et al., 2019; Wang and Japkowicz, 2010; Yan et al., 2003), hybrid method (Wang et al., 2012), and selecting performance metrics that are sensitive to class imbalance (Bradley, 1997; Davis and Goadrich, 2006; Goutte and Gaussier, 2005). This study employed a hybrid approach that integrates majority-class data undersamplers with CSAs to estimate the risk of anchor dragging for ships at anchor. A comprehensive overview of the cost-sensitive learning and data undersampling techniques is elucidated in subsections 4.2 and 4.3, respectively.

4.2 Cost-sensitive learning

Cost-sensitive learning is a special type of learning that considers misclassification costs to minimize the total misclassification costs. The confusion matrix in Table 6 describes algorithm performance for a binary-class problem. In the binary classification problem, assuming that $G = \{(\mathbf{x}_i, y_i) | \mathbf{x}_i \in \mathbb{R}^d, y_i \in \{-1, +1\}\}_{i=1}^n$ is a training set with n samples and d features, and y_i is the class label associated with instance \mathbf{x}_i . To accomplish the goal of classification, a predictor $f : \mathbf{x} \rightarrow \mathbb{R}$ is obtained, and a classification rule is often considered to be $\text{sign } f(\mathbf{x})$. The loss function L measures the performance of the algorithm. The regularized empirical risk minimization (ERM) is expressed by Eqn. (1), (Tian and Wang, 2022):

$$\min_{f \in F} (f, G) = \min_{f \in F} \left\{ \frac{1}{n} \sum_{i=1}^n L(f(\mathbf{x}_i), y_i) + \lambda H(f) \right\} \quad (1)$$

Where, λ denotes the regularization parameter, $H(\bullet)$ represents the regularizer. Most ML algorithms achieve error minimization using the ERM configuration. These algorithms assume that all misclassification errors, false negatives and false positives, incur equal costs resulting in cost-insensitive algorithms. In reality, many problems involving ML, such as medical diagnosis (Ding et al., 2020), financial distress (Safi et al., 2022), and fraud detection (Makki et al., 2019), are cost-sensitive. Cost-sensitive learning considers uneven

Table 6: The cost matrix for a binary class classification.

		Predicted class	
		Negative $f(\mathbf{x}) = -1$	Positive $f(\mathbf{x}) = +1$
Actual class	Negative ($y = -1$)	$C(-1, -1) = C_{TN}$	$C(+1, -1) = C_{FP}$
	Positive ($y = +1$)	$C(-1, +1) = C_{FN}$	$C(+1, +1) = C_{TP}$

misclassification costs. Correct classifications incur no cost; that is, $C_{TN} = C_{TP} = 0$. Furthermore, incorrectly classifying an instance often costs more than correctly classifying it i.e., ($C_{FP} > C_{TP}$ or $C_{FN} > C_{TN}$), as indicated in the cost matrix in Table 6. For example, the cost of misclassifying a non-terrorist as a terrorist is less than that of misclassifying an actual terrorist carrying a bomb on a flight ($C_{FN} > C_{FP}$) as a non-terrorist. In cancer diagnosis, misclassification of cancer is far more serious, as patients may lose their lives owing to late diagnosis and treatment ($C_{FN} > C_{FP}$). In this study, the cost of misclassifying

a dragging candidate was higher than that of misclassifying a non-dragging candidate, because the potential consequences of misclassifying a dragging vessel are of greater concern owing to the increased risk of collision with a ship in transit, another ship at anchor, or damage to the seabed infrastructure ($C_{FN} > C_{FP}$). The cost is not always measured in monetary value; it can take the form of time wasted, loss of life, damage to infrastructure, or the severity of an event or decision (Ling and Sheng, 2008). Cost-sensitive learning is implemented by modifying or adapting the ML algorithm. Existing cost-sensitive learning techniques include direct methods (Drummond and Holte, 2000; Ling and Sheng, 2008), thresholding (Chai et al., 2004; Domingos, 1999), and weighting (Ting, 1998).

The direct method integrates misclassification costs into the ERM loss function of the ML algorithm. Thresholding is a technique that transforms cost-insensitive algorithms into cost-sensitive ones without modifying the structure using a threshold to classify training examples based on probability estimations. Weighting involves assigning weights to the ERM loss function on each instance according to the misclassification cost (Günemann and Pfeffer, 2017). That is, instances in the minority class are assigned weights that are proportionally higher than instances in the majority class. The weighting technique is effective when a cost-insensitive algorithm has a parameter that is capable of accepting class weights. This study employed the weighting method to transform cost-insensitive algorithms, namely, logistics regression (LR), extreme gradient boosting (XGB), and the support vector machine (SVM) into CSAs. Additionally, the study included the Balanced Bagging Classifier (BBC), which is already a CSA.

4.2.1 Cost-sensitive LR

Standard LR, similar to most ML algorithms, assumes equal misclassification errors for imbalanced datasets. Therefore, it is essential to modify the algorithm to account for the imbalanced classes. To accomplish this, a class weighting configuration is used to control the updating of the algorithm coefficients during training. The weighting configuration is designed to impose a greater penalty on errors in the minority class samples. Furthermore, the model incurs a lower penalty for errors committed in the majority class samples. The modified log-likelihood function of a cost-sensitive LR (CLR) is represented by Eqn. (2), (Mienye and

Sun, 2021).

$$L = -\frac{1}{n} \sum_{i=1}^n \left[C_{FP} y_i \log(f(\mathbf{x}_i)) + C_{FN} (1 - y_i) \log(1 - f(\mathbf{x}_i)) \right] \quad (2)$$

Where $f(\mathbf{x}_i)$ is the predicted probability that y_i is true label for i . C_{FP} is the cost of a false positive, and C_{FN} is the cost of false negative.

4.2.2 Cost-sensitive SVM

The SVM was introduced by (Cortes and Vapnik, 1995) and is based on statistical learning theory. SVM is a class of ML algorithms that are most commonly utilized for classification, regression, and other applications. In a binary classification problem, an SVM searches for an optimal hyperplane that separates patterns between the two classes by maximizing the margin. The cost-sensitive SVM (CSVM) employs two distinct penalty weights to account for misclassification errors in the minority and majority classes. The optimal model for a CSVM is defined by Eqn. (3) and Eqn. (4) (Guido et al., 2023).

$$\arg \min_{\mathbf{w}, b, \xi} \frac{1}{2} \|\mathbf{w}\|^2 + \alpha \left[C_{FN} \sum_{i|y_i=1} \xi + C_{FP} \sum_{i|y_i=0} \xi \right] \quad (3)$$

$$\text{subject to } y_i(\mathbf{w}^T \mathbf{x} + b) \geq 1 - \xi \quad (4)$$

Where \mathbf{w} is the vector normal to the hyperplane, α is a penalty parameter that is a trade-off between the size of the margin and the training errors ξ , and b is the bias indicating the hyperplane offset from the origin.

4.2.3 Cost-sensitive XGB

XGB is an advanced gradient tree boosting-based algorithm that can handle large-scale ML tasks. The fundamental concept of the XGB algorithm is to continue adding trees and splitting the features to grow a tree. For binary classification problems, the default loss function

of XGB is the cross entropy (CE) loss, which is expressed by Eqn. (5), (He et al., 2021).

$$L = -\frac{1}{n} \sum_{i=1}^n \left[y_i \log(f(\mathbf{x}_i)) + (1 - y_i) \log(1 - f(\mathbf{x}_i)) \right] \quad (5)$$

Considering the misclassification case for a cost-sensitive XGB algorithm (CXGB), such that, $C_{TN} = C_{TP} = 0$, $C_{FN} = k(k > 0)$, and $C_{FP} = 1$, the loss function with cost-sensitive factor is denoted as shown in Eqn. (6) (He et al., 2021).

$$L = -\frac{1}{n} \sum_{i=1}^n \left[ky_i \log(f(\mathbf{x}_i)) + (1 - y_i) \log(1 - f(\mathbf{x}_i)) \right] \quad (6)$$

where k is a cost-sensitive factor.

4.2.4 BBC

Breiman (1996) introduced the first bagging algorithm which is an ensemble-based algorithm. He demonstrated that the ensemble model typically outperforms a single model. Bagging samples the training subsets from the entire dataset with replacement, builds multiple base learners and aggregates the outputs of the base learners to make the final predictions. A BBC classifier employs a balanced sampling methodology to conduct bootstrap sampling exclusively on the majority samples. This ensures that the sample size matches the number of minority samples in the original dataset while retaining all minority samples. The use of the negative binomial sampling technique in the undersampling of the majority class, without necessitating any adjustments to its hyperparameters, allows the BBC algorithm to be classified as a CSA (Hido et al., 2009).

4.3 Data resampling

Data resampling in imbalanced learning applications involves modifying imbalanced data through a mechanism that is designed to undersample the majority instances or oversample the minority instances to achieve a balanced class distribution and improve the classification performance of standard classifiers, particularly for minority class instances. Elimination can be executed randomly or by employing more precise and efficient criteria that eliminate borderline and noisy samples. Specifically, the resultant dataset should contain only instances that are similar to those in the original dataset; that is, all instances in the modified dataset should be drawn from the same distribution as those in the original dataset. Thus, resampling techniques are not required to generate a perfectly balanced class distribution, but rather a distribution that is more amenable to standard classifiers. Data resampling techniques are employed as wrapper-based approaches combined with CSA or cost-insensitive algorithms (Weiss et al., 2007).

There are well-documented drawbacks of using data resampling in cost-sensitive learning. However, undersampling has the disadvantage of discarding potentially valuable data. The primary disadvantage of oversampling is that it increases the likelihood of overfitting by creating artificial samples from the existing minority class samples. In addition, oversampling artificially increases the number of training examples, thereby lengthening the learning duration. In this study, our primary interest is the dragging cases, which are the minority class. Thus, we aim to preserve the dragging samples while resampling the non-dragging samples using undersampling techniques such as random undersampler (RU), repeated edited nearest neighbors (RENN), all k-nearest neighbors editing (AKNN), one sided selection (OSS), and neighborhood cleaning rule (NCR).

4.3.1 RU undersampler

RU is a heuristic method that balances the data by randomly eliminating the majority class samples. Using the RU function in python, the number of samples in the majority class will be reduced to attain balanced class sizes.

4.3.2 RENN undersampler

RENN which was proposed by [Tomek \(1976\)](#) utilizes ENN developed by [Wilson \(1972\)](#) to identify and eliminate noisy, redundant, and borderline majority class samples in a dataset. The ENN rule involves applying $k = 3$ nearest neighbors to every sample in the majority class. This allows for elimination of samples that are misclassified as belonging to the minority class while retaining those that are correctly classified. The ENN rule is repeated an unlimited number of times, hence the name repeated edited nearest neighbor, until it converges to an elimination limit, resulting in the optimal distribution of the majority class.

4.3.3 AKNN undersampler

The AKNN proposed by [Tomek \(1976\)](#), shares a similar characteristic with RENN in that it iteratively examines and eliminates majority class samples until no further elimination can be made from the dataset. AKNN iteratively applies ENN to each sample, by increasing value of k in each iteration. If a sample label does not match the predominant label for at least one k value, it is eliminated from the dataset ([El Hajjami et al., 2021](#)).

4.3.4 OSS undersampler

OSS, which was proposed by [Kubat et al. \(1997\)](#), is an undersampling technique that integrates the use of Tomek links and the Condensed Nearest Neighbor (CNN) Rule. Tomek links refer to noise as samples at the class boundary, which are identified and eliminated from the majority class. The CNN method is then applied to eliminate samples from the majority class that are redundant and distant from the decision boundary. CNN is a one-step procedure that involves adding all minority class samples and a certain number of majority class samples to the store, classifying all remaining majority class samples with KNN, and adding misclassified samples to the store.

4.3.5 NCR undersampler

NCR (Laurikkala, 2001) is an undersampling technique that combines the CNN and Wilson ENN rules to eliminate redundant and noisy samples. Similar to OSS, CNN is applied in a one-step manner, after which examples misclassified by a KNN classifier are eliminated, according to the ENN rule. This approach begins with the selection of all examples from the minority class. Subsequently, using the ENN rule, all noisy samples in the majority class are identified and eliminated. Finally, a one-step version of CNN is used, in which the remaining misclassified samples in the majority class are eliminated, only if the number of samples in the majority class is greater than the specified size of the minority class.

A significant limitation associated with sampling techniques is the need to establish the appropriate extent of sampling. Similarly, it is imperative to select an appropriate level of undersampling that preserves a significant amount of information pertaining to the majority class while simultaneously promoting a balanced class distribution. This challenge is addressed through the use of the wrapper method to optimize the undersampler hyperparameters using the grid-search technique.

4.4 Performance metrics

Most standard algorithms are accuracy driven. However, in a class imbalanced dataset, the classification accuracy metric fails to capture the performance of the minority class accurately, leading to an inaccurate and misleading evaluation of classifier effectiveness. Cost-sensitive learning approaches are designed to prioritize the accurate classification of the positive class by assigning a higher cost to false negative (FN) than to false positive (FP). TP and TN are the correct positive and negative predictions, respectively, as shown in Table 7. In this study, a FN was an error in which the model incorrectly predicted a dragging instance as non-dragging. By contrast, a FP is an error in which the model incorrectly predicted a non-dragging instance as dragging. Therefore, to evaluate the performance of the proposed algorithms, we employed assessment metrics that offered precise measurements of the penalized FN. Additionally, a metric was employed to quantify the best trade-off between FN and FP. This study used the performance evaluation metrics derived from the confusion matrix presented in Table 7: the recall, specificity, GM, and AUC.

The recall, also known as the sensitivity or true positive rate (TP_{rate}), is the fraction of the dragging instances correctly classified as positive for all instances that are positive and is computed as shown in Eqn. (7).

$$TP_{rate} = \frac{TP}{TP + FN} \quad (7)$$

A higher recall value indicates a better ability of the classifier to minimize FN predictions.

Table 7: The confusion matrix for a binary class classification.

		Predicted	
		Negative	Positive
Actual	Negative	True Negative (TN)	False Positive (FP)
	Positive	False Negative (FN)	True Positive (TP)

The specificity, or true negative rate (TN_{rate}), as shown in Eqn. (8), is the fraction of non-dragging instances correctly classified as negative for all instances that are negative.

$$TN_{rate} = \frac{TN}{TN + FP} \quad (8)$$

The GM, also known as G-mean, is defined as the geometric mean of sensitivity and specificity. GM measures the balanced trade-off between the two classes (Barandela et al., 2003). The higher the GM value, the better the comprehensive performance of a classifier, as illustrated in Eqn. (9).

$$GM = \sqrt{TP_{rate} \times TN_{rate}} \quad (9)$$

The AUC measures the ability of the model to distinguish between negative and positive classes (Bradley, 1997). The graphical plot of the receiver operating characteristic (ROC) curve is constructed by plotting the FP_{rate} on the x-axis and TP_{rate} on the y-axis. A perfect score is attained when a model achieves a TP_{rate} of 1 and FP_{rate} of 0. Hence, a good classification model yields points near the upper left coordinates of ROC curve (Ali et al., 2013). Furthermore, a high AUC value demonstrates the effectiveness of the model in distinguishing the classes. The AUC score was calculated using Eqn. (10).

$$AUC = \frac{1 + TP_{rate} - FP_{rate}}{2}; \quad \text{where } FP_{rate} = \frac{FP}{FP + TN} \quad (10)$$

5 Data analysis and experiment procedure

The five-year anchorage data used in this research comprised six input features, with a total of 15,042 samples collated from historical AIS data, Korea Meteorological Administration meteorological information, and Korean Hydrographic and Oceanographic Agency nautical publications. Statistical analysis of the dataset revealed that the dragging samples accounted for 0.78% of the anchorage data, indicating highly imbalanced data (Dal Pozzolo et al., 2015). A visual inspection of the data provided vital information that helped in identifying data properties such as noise, borderline sample and class overlapping which could degrade a classifier performance. The effects of noise and borderline samples were minimized by deleting noise and borderline samples when carrying out data resampling; in this case, undersampling criteria were employed.

5.1 Class dispersion

Fig. 8, illustrates the class distribution and class overlap for all features in the anchorage dataset. Classifiers tend to exhibit better performance when trained on data with a high degree of divergence in the class overlap, as opposed to data with a low degree of divergence in the class overlap. Table. 8 shows a comparison of the class overlay measurement and rank from cluster mean difference in comparison to the recursive feature elimination cross-validated (RFECV) algorithm. RFECV is a popular feature selection wrapper-based algorithm that selects and ranks features for a supervised ML dataset according to importance. Divergence is a measure of class overlay.

The ship length and wind speed in Figs. 8(a) and (d), respectively, exhibited a significant degree of divergence in the class overlap. The ship draft, sea depth, wind direction, and seabed type, as illustrated in Fig. 8(b),(c),(e), and (f), respectively, exhibited minimal to complete overlap of the minority and majority classes. Most classifiers follow the separate-and-conquer strategy, which recursively divides and solves smaller problems to induce the whole concept (Prati et al., 2004). Therefore, wind speed and ship length features were expected to significantly contribute to the separation of classes during classifier training owing to their strong predictive power.

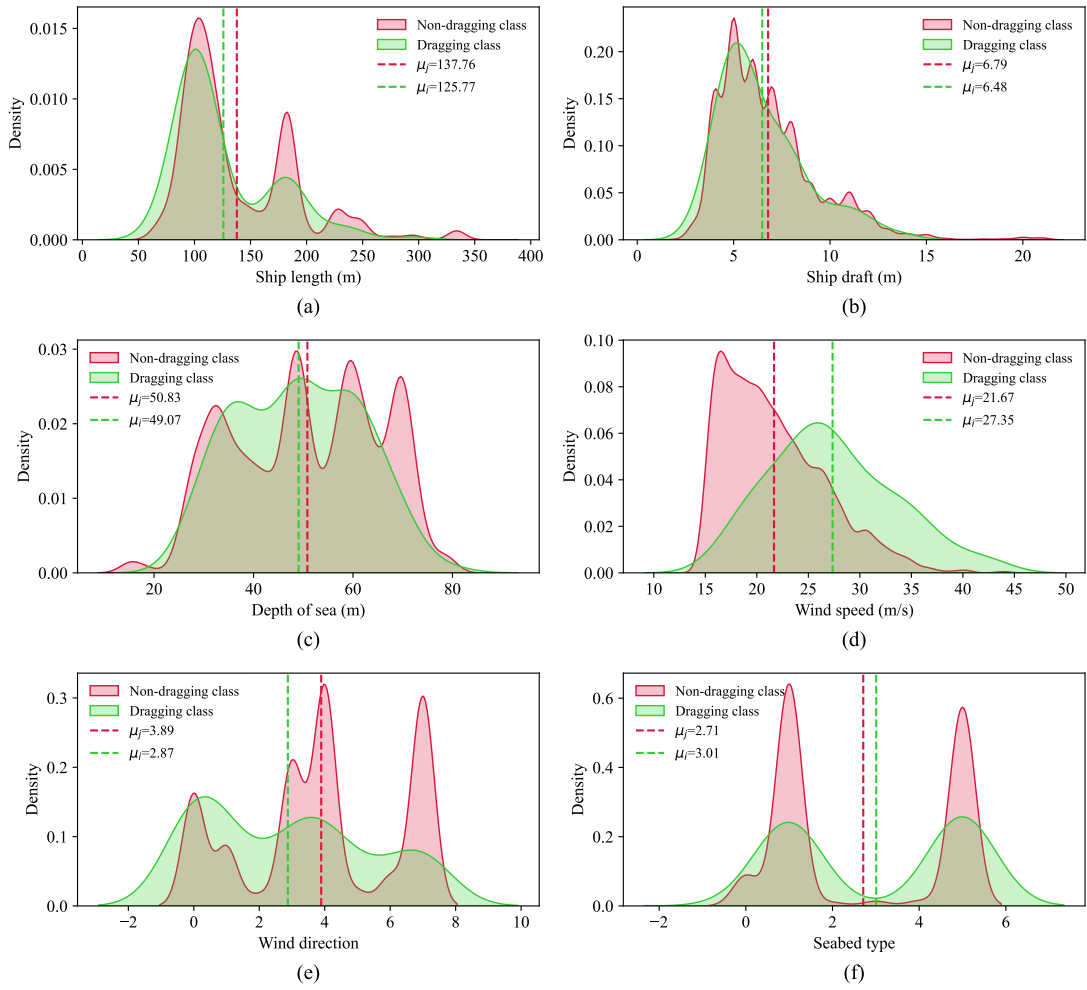


Fig. 8: Class distribution and class overlap of features in anchorage data: **(a)** ship length distribution, **(b)** ship draft distribution, **(c)** sea depth distribution, **(d)** wind speed distribution, **(e)** wind direction distribution, and **(f)** seabed type distribution.

Table 8: Measure of class dispersion in comparison to RFECV.

Feature	Data type	Non-dragging class mean (μ_j)	Dragging class mean (μ_i)	Dispersion ($\mu_j - \mu_i$)	Rank	RFECV rank
Ship length	Continuous	137.76	125.77	11.99	1	2
Ship draft	Continuous	6.79	6.48	0.31	5	5
Sea depth	Continuous	50.83	49.07	1.76	3	3
Wind speed	Continuous	21.67	27.35	5.68	2	1
Wind direction	Categorical	3.89	2.87	1.02	4	4
Seabed type	Categorical	2.71	3.01	0.30	6	6

5.2 Exploratory data analysis

Table 9 shows the descriptive statistics of continuous features of the anchorage dataset. Data with missing values were excluded from statistical analysis and ML model training.

Table 9: Descriptive statistics for the continuous features of the anchorage dataset.

	Ship length		Ship draft		Sea depth		Wind speed	
	Non-dragging class	Dragging class	Non-dragging class	Dragging class	Non-dragging class	Dragging class	Non-dragging class	Dragging class
Count	14924	118	14924	118	14924	118	14924	118
Mean	137.764	125.771	6.787	6.430	50.825	49.068	21.668	27.353
std	51.963	44.688	2.593	2.280	14.516	12.310	4.909	6.104
min	61.000	66.000	2.000	3.000	15.200	26.500	15.162	16.134
25%	100.000	99.000	5.000	5.000	38.000	38.000	17.689	23.326
50%	116.000	106.000	6.000	6.000	50.000	48.000	20.799	26.436
75%	180.000	159.000	8.000	8.000	63.000	59.000	24.687	31.490
Max	366.000	295.000	21.000	14.000	79.000	79.000	43.931	42.764

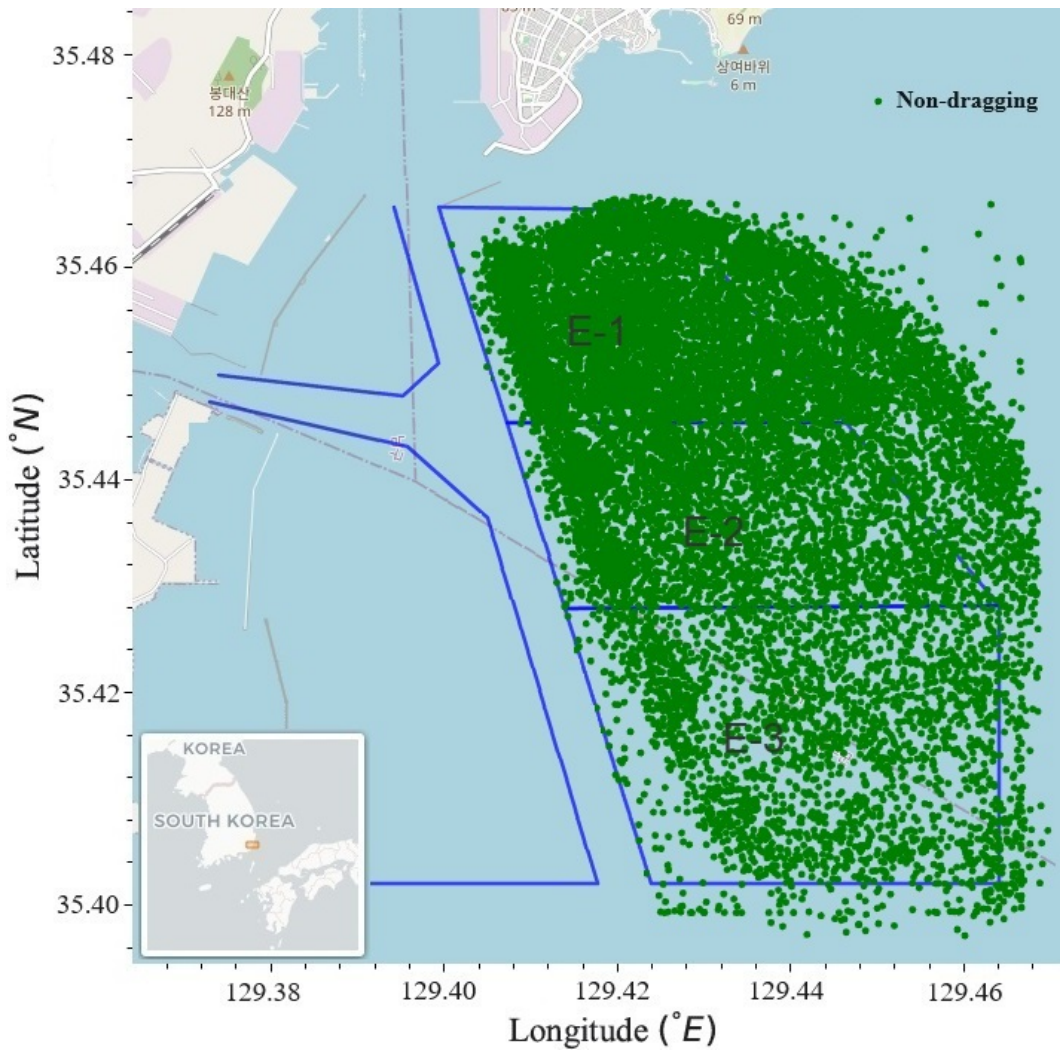


Fig. 9: The positions of non-dragging anchor ships in Ulsan port E-group anchorage area.

A total of 15,042 ships at anchor in the Ulsan E-group anchorage area were extracted from 2017 to 2021. In this period, 7,975 ships (53 %) anchored in E-1, 4,269 ships (28 %) anchored in E-2, and 2,798 ships (19 %) anchored in E-3 anchorage areas. Fig. 9 shows the positions of the 14,924 non-dragging ships at anchor in the Ulsan E-group anchorage area. 7,899 ships in E-1, 4,238 in E-2, and 2,787 in E-3 did not drag anchor. Fig. 10 shows the positions of the 118 anchor dragging ships in the Ulsan E-group anchorage area. 76 ships dragged the anchor in E-1, 31 ships dragged the anchor in E-2, and 11 ships dragged the anchor in E-3 for the period 2017 to 2021. It is observed that E-3 is the least preferred anchorage area for ships.

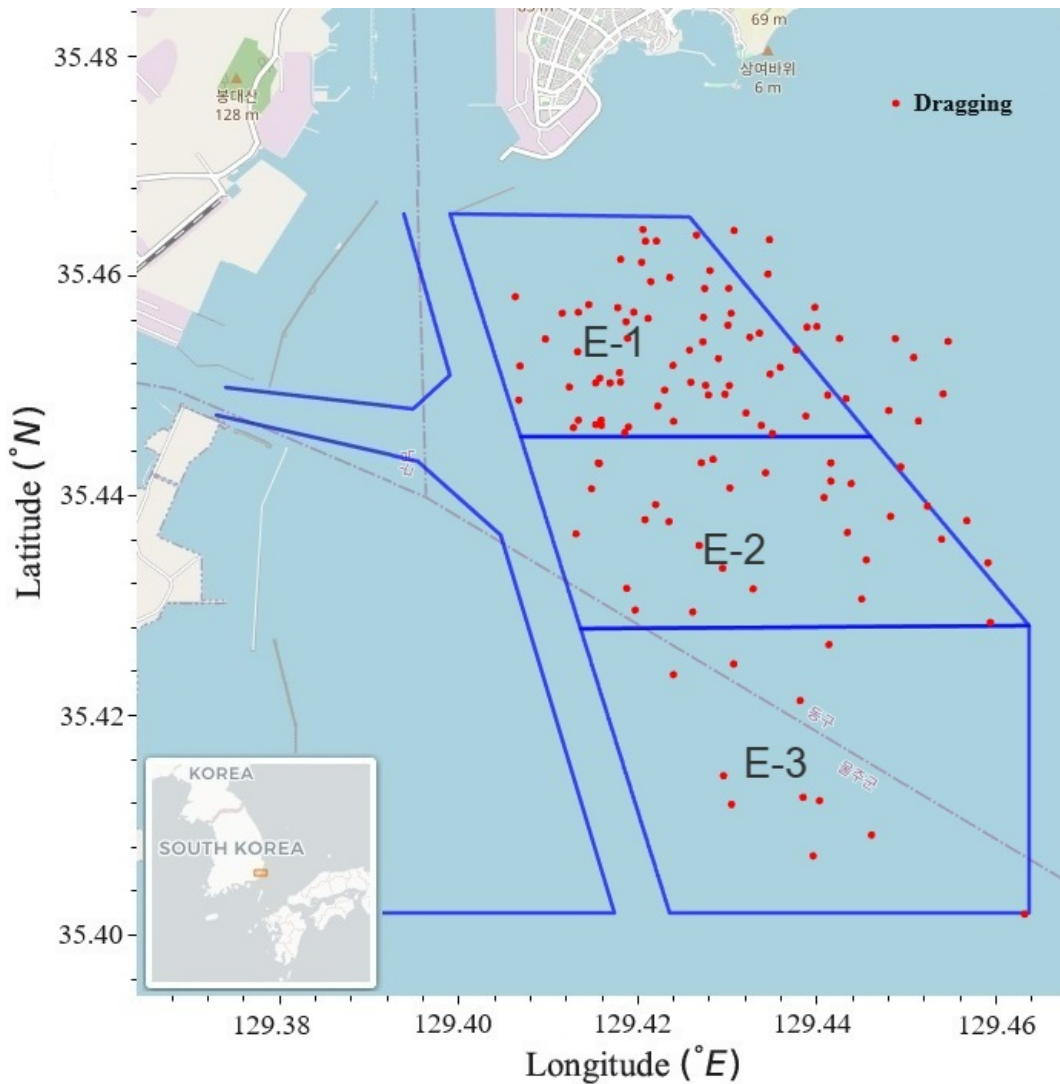


Fig. 10: The positions of dragging anchor ships in Ulsan port E-group anchorage area.

E-3 has deep waters of an average depth of 60 m as highlighted in Table 4. Deep waters are a deterrent for anchorage, as a greater amount of anchor chain payout is required to cover the catenary, leaving less chain for penetration into the seabed for better holding power.

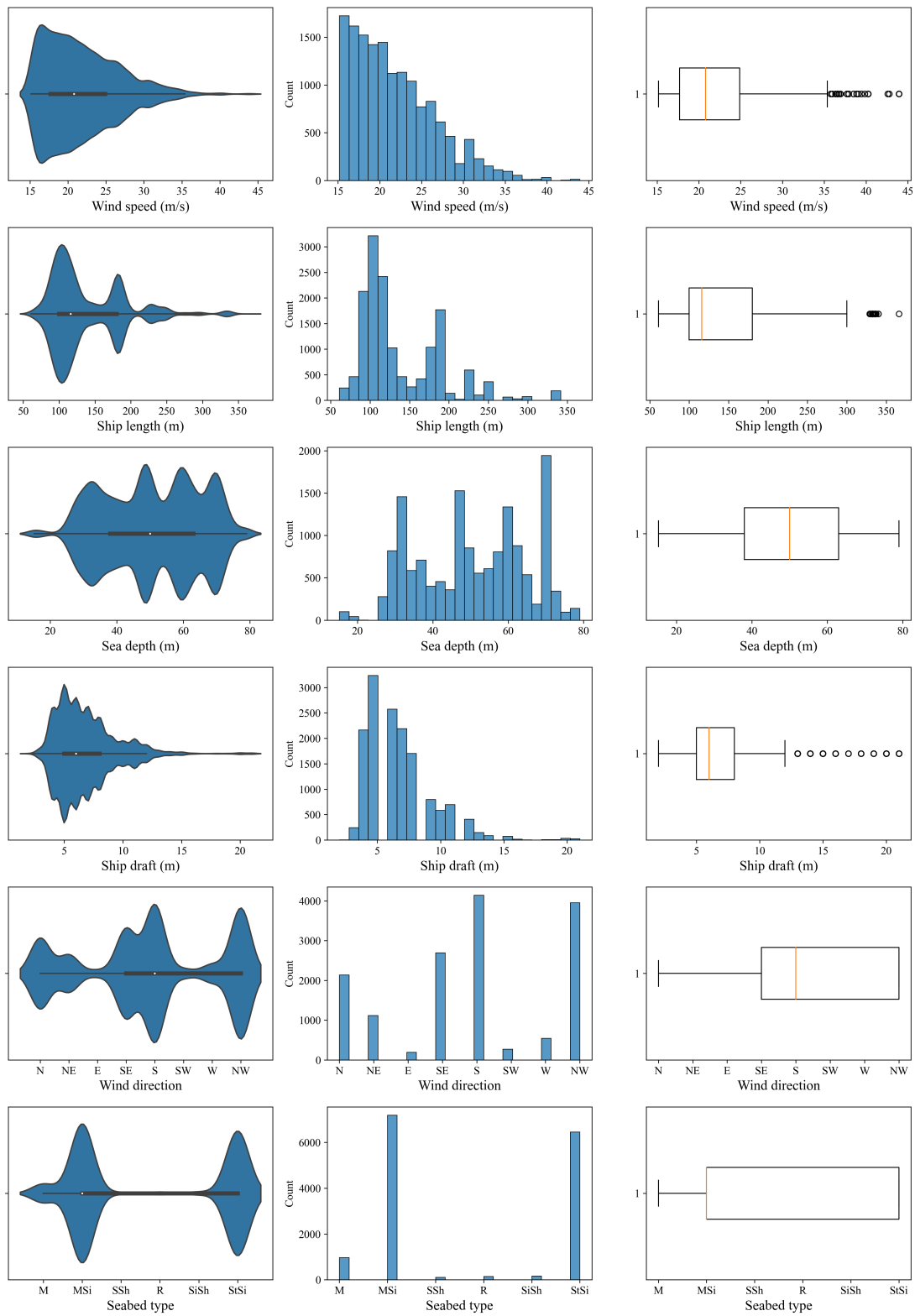


Fig. 11: Distribution of the features for the anchorage dataset.

Fig. 11 is a violin, histogram, and box plot showing the distributions of the anchorage dataset features. The mean scores of each feature can be approximated from the violin and box plots, whereas the distribution patterns of each feature can be observed from the violin and histogram plots. The wind blows mostly from the south (S) and northwest (NW) directions. Ulsan E-group anchorage area is dominated by mud over silt (MSi) and sticky silt (StSi) seabed types.

Fig. 12 shows a rose plot of the wind speed and direction for the dragging and non-dragging anchor labels. The wind mostly blew from the northeast at a speed greater than 30 m/s at 24.6 % and 12.4 % of the time respectively, for the dragging cases. In addition, for the non-dragging cases, the wind mostly blew from the south for 20.5 % of the time

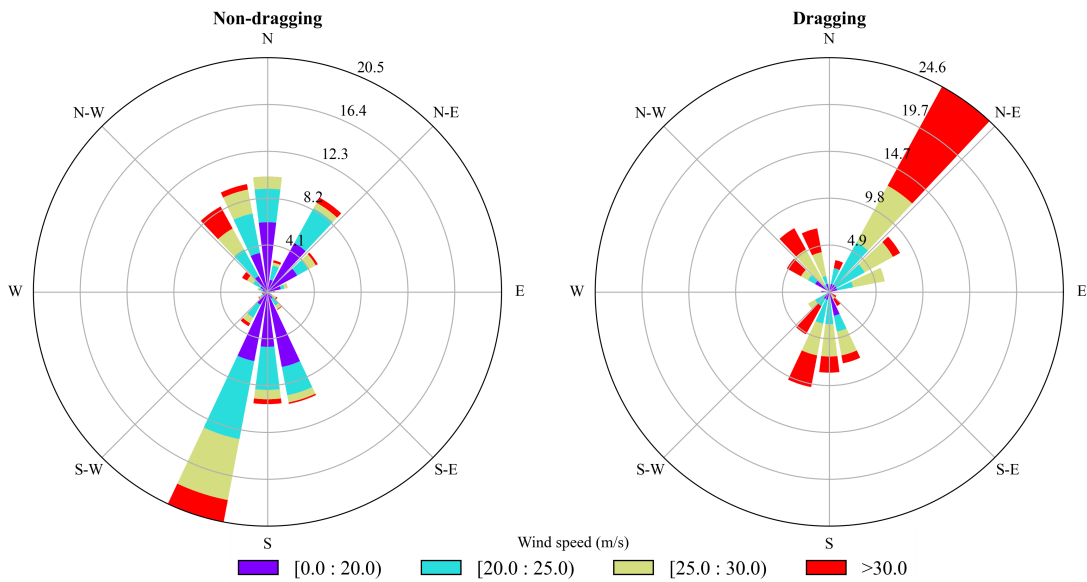


Fig. 12: Wind speed and direction for the non-dragging and dragging cases in the anchorage dataset.

Fig. 13 shows a rose plot of the wind speed and direction for the anchorage dataset by month of the year. The wind mostly blew from the north and northwest 65.8 % and 55.2 % in January and February respectively. The wind predominantly blew from the south 22.5 %, 27.4 %, 38.8 %, 34.3 %, 32.6 %, and 40.6 % of the time in March, April, May, June, July, and August, respectively. In September, the wind mostly blew from the northeast 27.3 % of the time. In October, the wind mostly blew from the north and northeast 30.6 % and 13.6 % of the time, respectively. The wind mostly blew from the north 47.1 % of the time in November. In December, the wind blew from the northwest 60.2 % of the time.

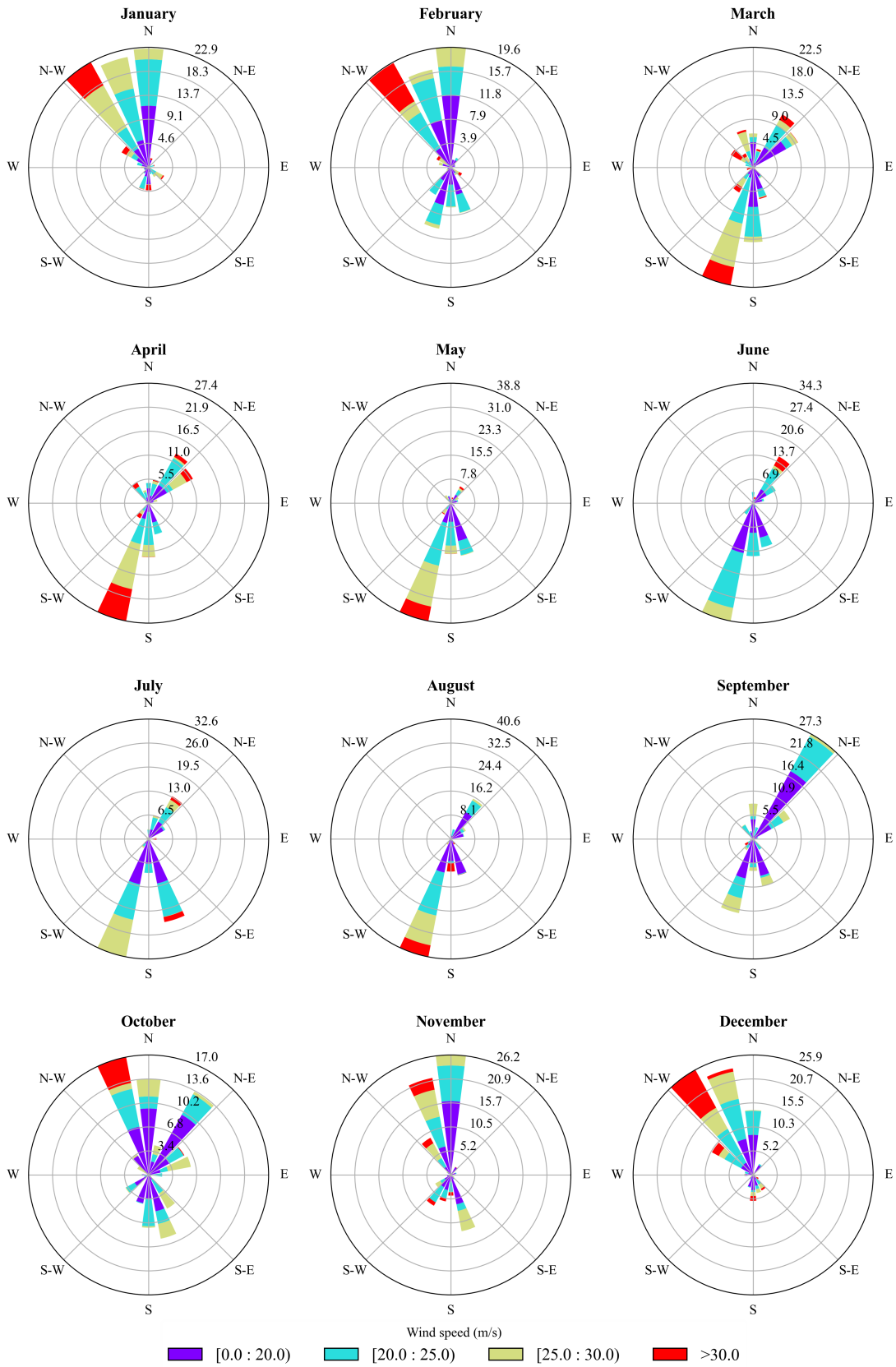


Fig. 13: Wind speed and direction distribution by the month of the year.

The month of December had the longest frequency of wind speeds above 30 m/s at 11.4 % of the month, followed by April at 9.2 % of the month, February at 7.8 % of the month, May at 5.3 % of the month, January at 4.7 % of the month, October at 4.2 % of the month, August at 4.1 % of the month, March at 3.5 % of the month, November at 3.6 % of the month, June at 2.1 % of the month, and July at 1 % of the month. The month of September never recorded wind speeds greater than 30 m/s .

5.2.1 Dragging samples analysis

The study hinges on the data quality of the minority class, which consists of dragging samples of the anchorage dataset. The study focuses on the feature relationships and characteristics of dragging samples to design ML model. The behavioral characteristics of a ship that is dragging anchor can be identified by analyzing the dragging samples.

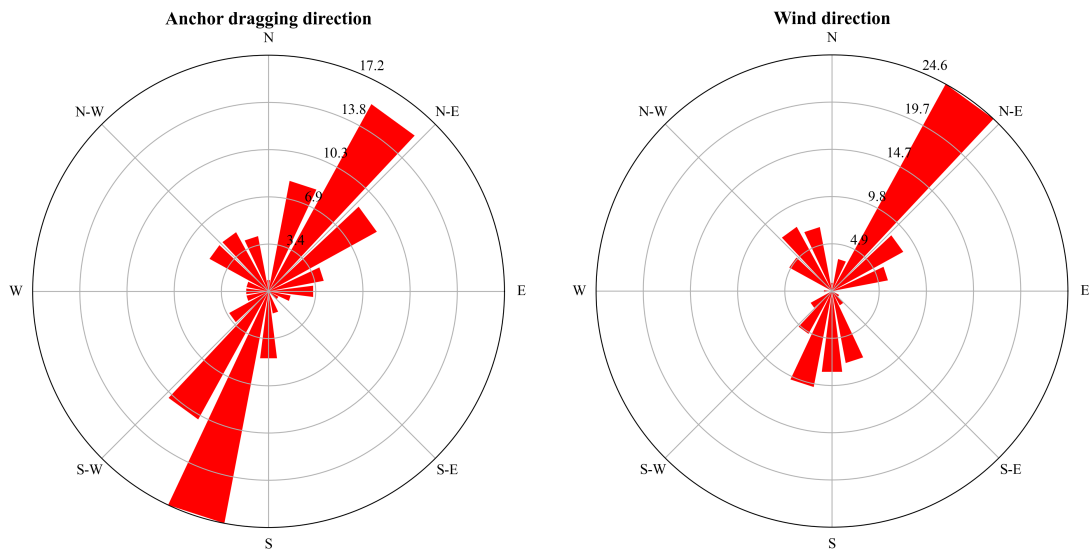


Fig. 14: Anchor dragging direction compared to wind direction for the dragging samples of the anchorage dataset.

Fig. 14 shows a comparison of the anchor dragging direction to the direction of the wind for the dragging samples in the anchorage dataset. It is generally observed that the dragging direction is consistent with the direction of the wind. The most dominant dragging direction is the NE followed by SSW, which is consistent with the dominant direction of the wind, which is the NE followed by SSW. Therefore, the direction of the wind can generally be used to predict the direction of dragging.

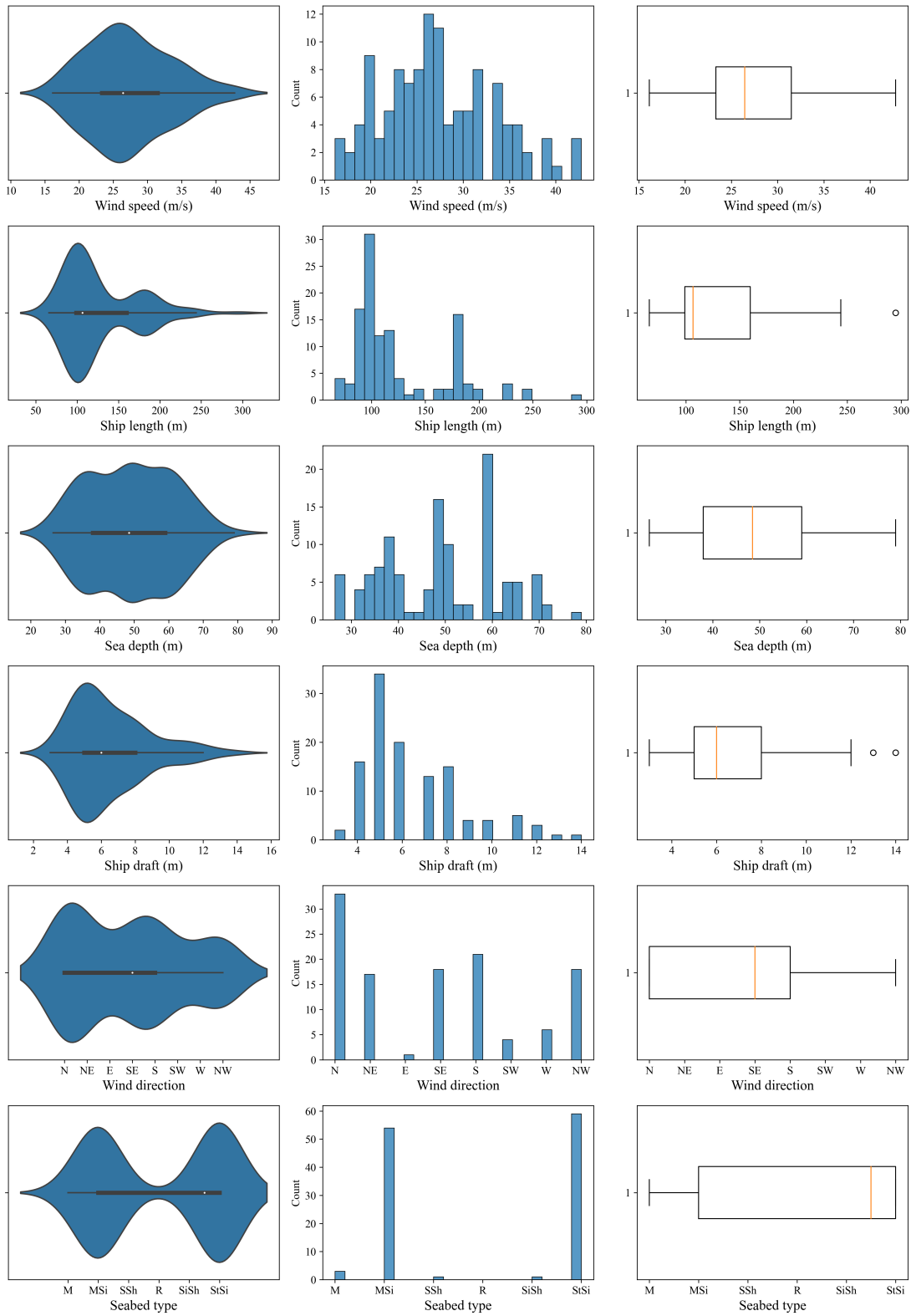


Fig. 15: Distribution of the features for the dragging samples.

Fig. 15 is a violin, histogram, and box plot showing the distributions of the dragging samples. The mean scores of each feature can be approximated from the violin and box plots, whereas the distribution patterns of each feature can be observed from the violin and histogram plots. The wind speed for the dragging samples is 27 m/s . The ships with a length overall of 100 m are prone to dragging the anchor. No distinct sea depth value instigates the anchor's dragging, as observed from the violin plot. The ship draft indicates the loading condition of the ship. It is observed from the violin plot that a ship draft of 6 m has a significant risk of dragging. Mud over silt (MSi) and sticky silt (StSi) seabed types are predominant in the Ulsan E-group anchorage area. An effective cost-sensitive ML model is dependent on the distinct statistical features of the dragging samples. The CSA learns from the distinct features to design an effective, robust ML model that has excellent prediction accuracy.

5.3 CSA and resampling techniques

The undersampling, training, and model selection processes are summarized in Fig. 16. The anchorage dataset was split in the ratio 85 % training set to 15 % test set to prevent data leak from the training set to the test set during training and to provide a large number of dragging samples for the CSA to build a better representation of classes (Joseph, 2022).

Using stratified sampling when splitting the dataset alleviates the problem of random sampling in datasets with an imbalanced class distribution. In this case, the distribution of the classes in the training and test sets was preserved. The wrapper method was used to address the inability of the undersampler to quantify the appropriate level of sampling for application to the data. The sampling levels were determined from the grid-search technique, where the grid range was controlled by the total number of dragging cases [1, 2, ..., 118]. The wrapper method involved the application of various sampling levels from the grid search to the training dataset during CSA training. The performance of each sampling level was measured to determine the optimal level for the undersampler. The sampling level with the best performance was subsequently employed as the optimal hyperparameter value for the undersampler. Table. 10 lists the tabulated optimal parameters for the undersamplers for each of the selected ML algorithms as wrappers. The training dataset was resampled using five optimized undersamplers.

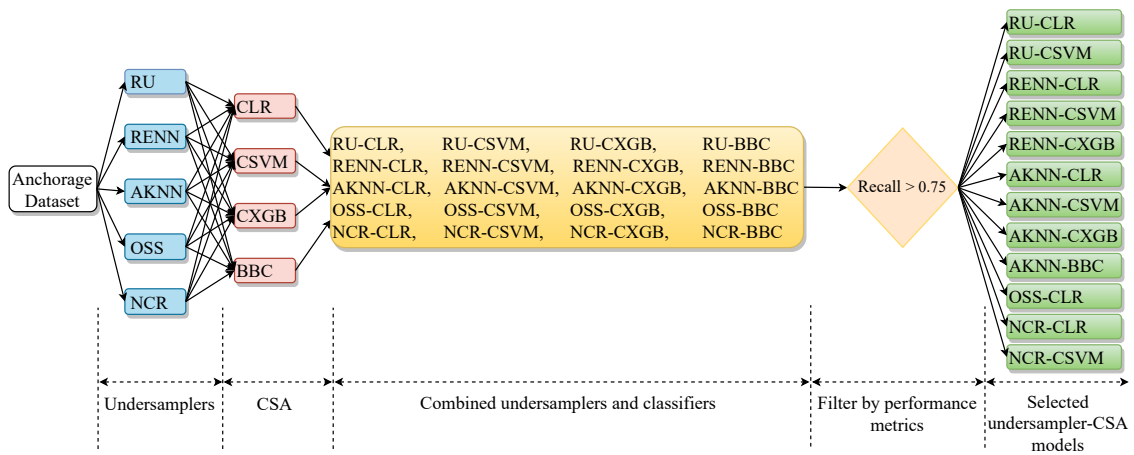


Fig. 16: Process flow from data resampling, CSA training, and performance metrics-based model selection. The model naming scheme is a combination of undersampler and CSA initials. For example, RU-CLR denotes Random Undersampler - Cost-sensitive Logistics Regression.

Table 10: Selected optimal parameter values for the undersamplers.

Undersampler	Parameter	Description	Optimal selection			
			LR	SVM	XGB	BBC
RU	sampling-strategy	Sampling information specifying the class targeted for resampling.	majority	majority	majority	majority
RENN	n-neighbors	Size of the neighborhood to consider to compute the nearest neighbors.	5	107	113	7
	sampling-strategy	Sampling information specifying the class targeted for resampling.	majority	majority	majority	majority
AKNN	n-neighbors	Size of the neighborhood to consider to compute the nearest neighbors.	5	109	113	91
	sampling-strategy	Sampling information specifying the class targeted for resampling.	majority	majority	majority	majority
OSS	n-seeds-S	The number of samples to extract in order to build the set S.	12	13	12	13
	sampling-strategy	Sampling information specifying the class targeted for resampling.	majority	majority	majority	majority
NCR	n-neighbors	Size of the neighborhood to consider to compute the nearest neighbors.	5	5	27	115
	sampling-strategy	sampling information specifying the class targeted for resampling.	majority	majority	majority	majority

Fig. 17 - Fig. 20 visualizes both the original and resampled datasets on scatter plots and distribution curves, whereas Table 11 shows a comparison of the training dataset before and after undersampling. The original training dataset in Fig. 17 - Fig. 20 exhibits a lognormal distribution curve for the non-dragging class and a normal distribution curve for the dragging class. The wind speed feature was used as a representative feature to visualize the distribution of the resampled data owing to its superior predictive power. Similarly, wind speed and ship length were used to visualize a scatter plot of the data as the two features have superior predictive power. The scatter plot displays that the original dataset had 12,685 non-dragging samples and 100 dragging samples. These were reduced to the sizes shown in Table 11 by RU, RENN, AKNN, OSS, and NCR undersamplers, as shown in Fig. 17 - Fig. 20 (c),(e),(g),(i), and (k), respectively.

Table 11: A comparison of the training dataset with the undersampled non-dragging class for each ML algorithm

Undersampler	Training dataset size	Undersampled dataset size			
		LR	SVM	XGB	BBC
RU	12,685	100	100	100	100
RENN	12,685	12,070	2,735	2,511	11,866
AKNN	12,685	12,141	2,741	2,613	4,375
OSS	12,685	10,741	10,615	10,484	10,977
NCR	12,685	12,221	12,221	10,466	7,303

The data sampled by RENN, AKNN, OSS, and NCR as illustrated in Fig. 17 - Fig. 20 (f),(h),(j), and (l), respectively, exhibited a lognormal distribution for the non-dragging class that closely resembled the original training data. This implies that the undersamplers do not need to perfectly balance the classes to retain vital information and the original distribution of the sampled data. The data resampled by RU in Fig. 17(d) - Fig. 20(d) were modified to a near-normal distribution, resulting in a balanced representation of the non-dragging samples with a total of 100 minority samples. The modification of the original data highlights the key disadvantage of non-heuristic RU; that is, it randomly deletes samples, thereby risking the removal of potentially vital information in the non-dragging class and altering the data distribution. The resampled data from each of the five undersamplers was trained in each of the four CSA, resulting in a total of twenty undersampler-CSA models notated as illustrated in Fig. 16.

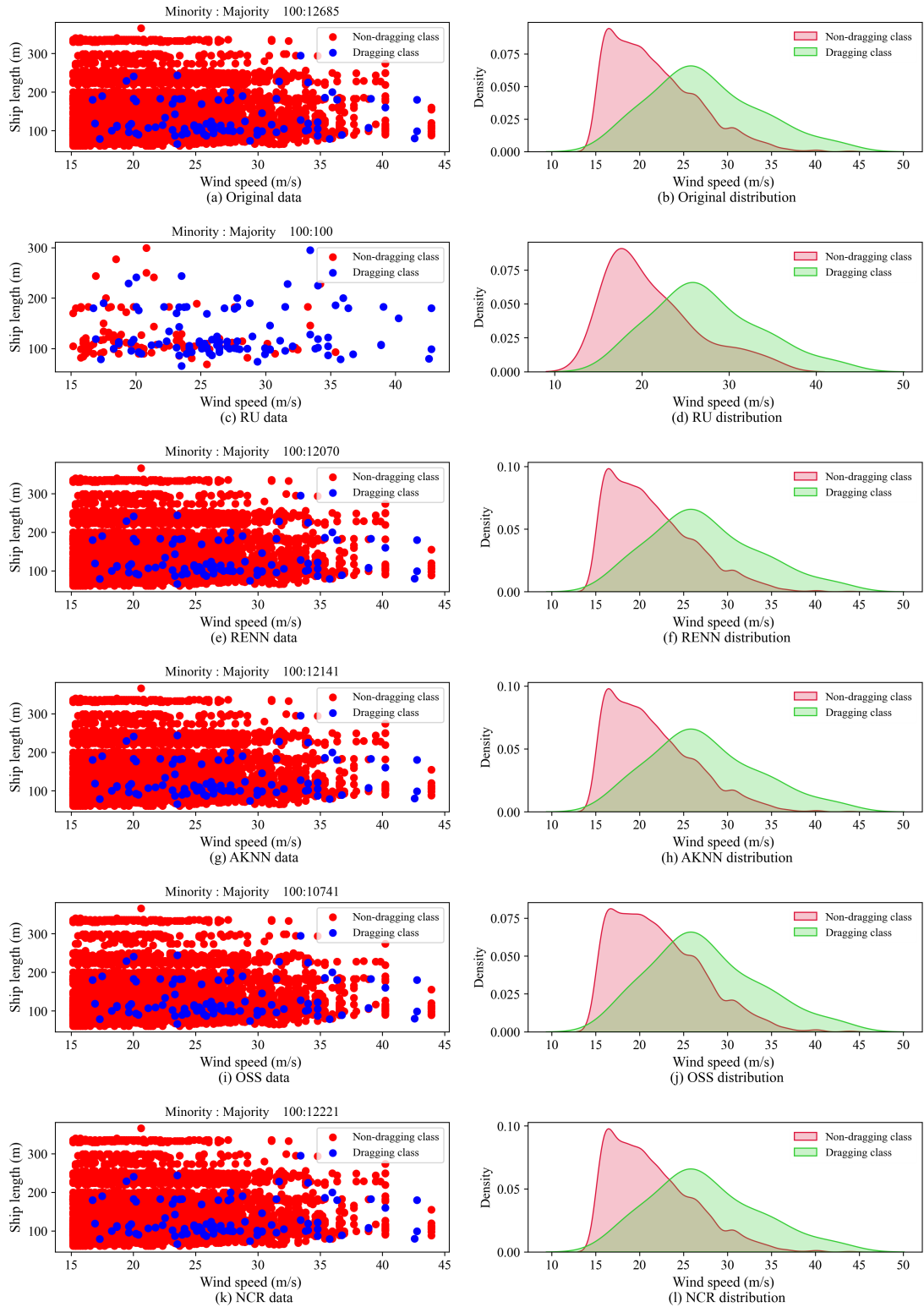


Fig. 17: The scatter and distribution plots of the training dataset before and after resampling with cost-insensitive LR as the wrapper algorithm.

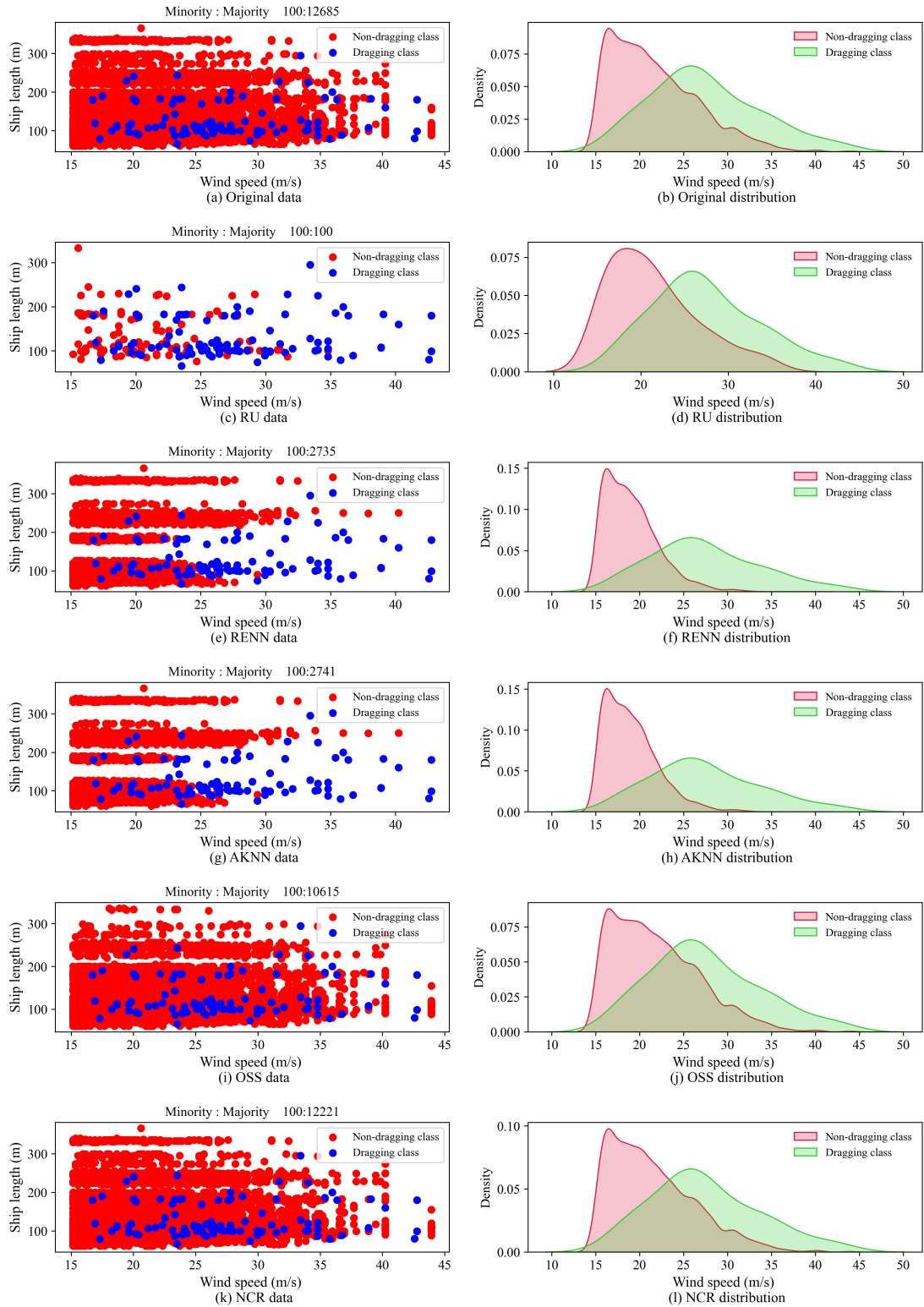


Fig. 18: The scatter and distribution plots of the training dataset before and after resampling with cost-insensitive SVM as the wrapper algorithm.

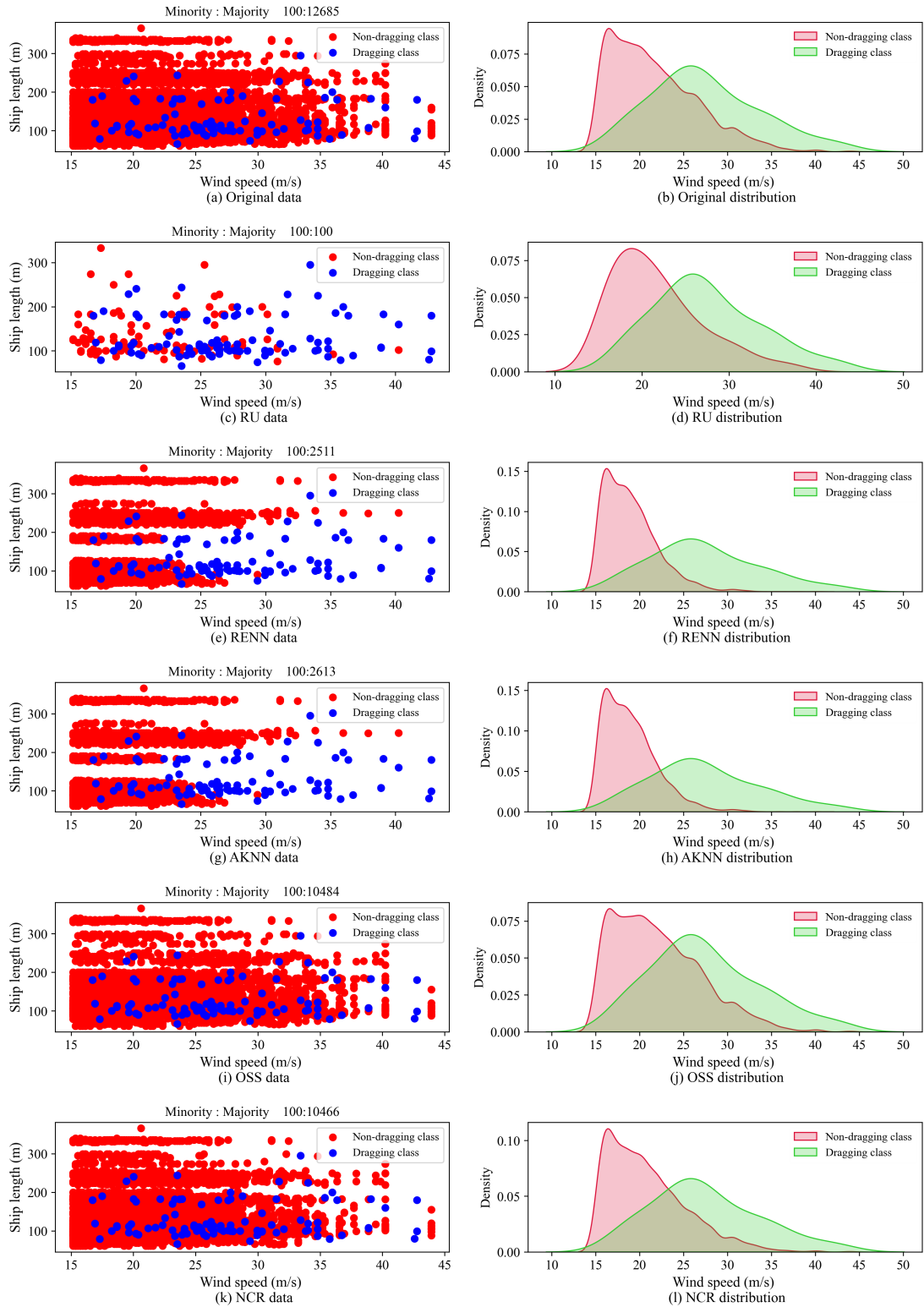


Fig. 19: The scatter and distribution plots of the training dataset before and after resampling with cost-insensitive XGB as the wrapper algorithm.

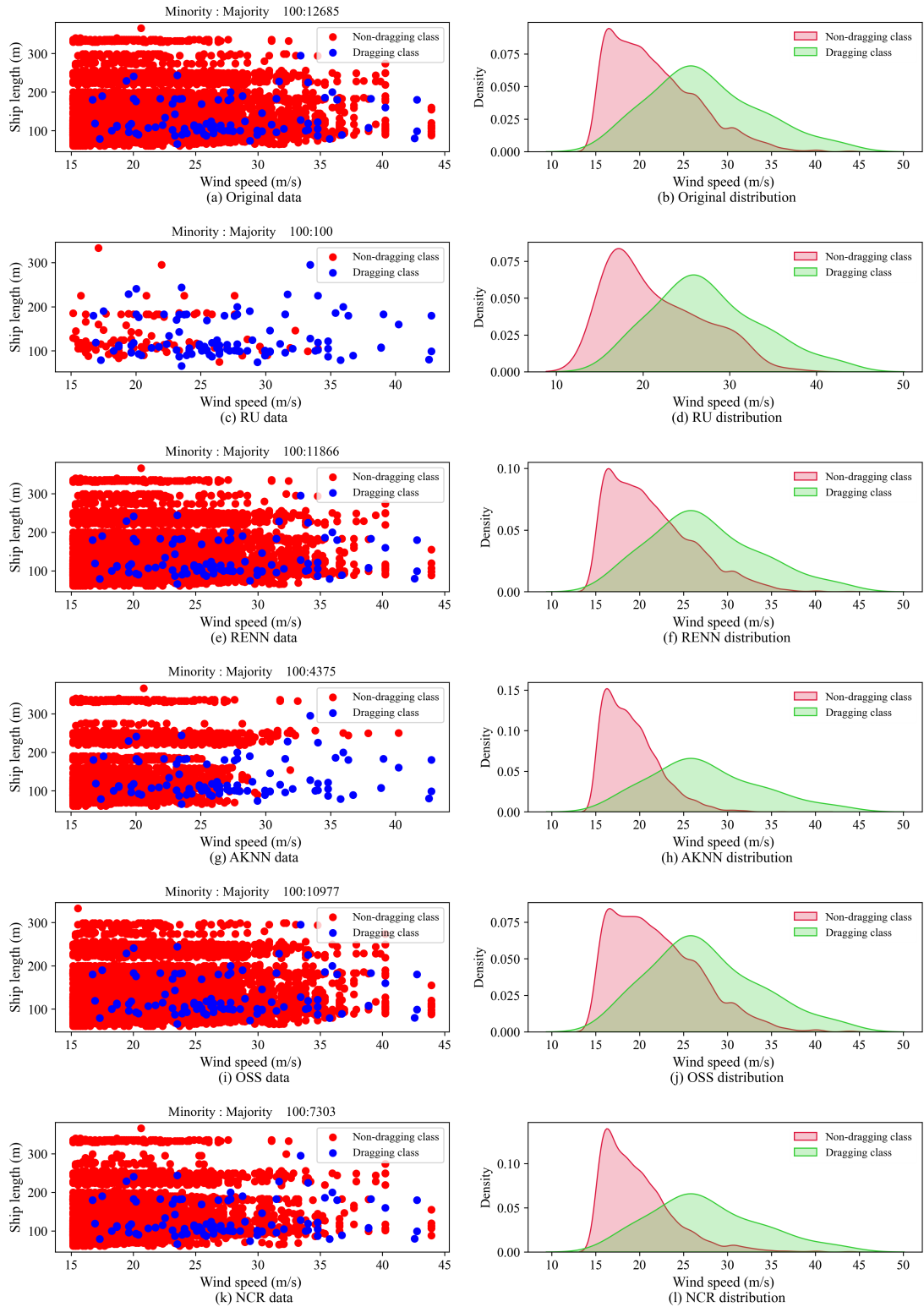


Fig. 20: The scatter and distribution plots of the training dataset before and after resampling with BBC as the wrapper algorithm.

The algorithms were weighted by the heuristic method, as shown in the tabulated CSAs and corresponding weighting parameters in Table 12. The CXGB algorithm weight was determined by inverting the class distribution (β); that is, the total number of non-dragging samples in the resampled data divided by the total number of dragging samples. Experimental trials were conducted using a Windows computer with an Intel(R) Core(TM) i7-13700F (24 CPUs) @ 2.10 GHz processor and 64GB RAM. Models with a recall score above 0.75 were selected, and their performances are described in section 6.

Table 12: Cost-sensitive algorithms parameters properties.

CSA	Data type
CLR	solver = lbfgs class-weight = balanced
CXGB	scale-pos-weight = β
CSVM	gamma = scale class-weight = balanced

6 Experiment results and discussion

The recall, specificity, GM, and AUC assessments for the CSA and selected undersampler-CSA models are presented in Table 13. Fig. 21 compares the performance metrics of the CSA and undersampler-CSA models. The best performing undersampler-CSA models were compared with the CSA.

Table 13: Experimental prediction performance of the selected undersampler-CSA models in comparison to the CSA.

Model	Recall	Specificity	GM	AUC
CLR	0.833	0.680	0.753	0.806
CSVM	0.722	0.711	0.716	0.797
CXGB	0.611	0.875	0.731	0.765
BBC	0.591	0.827	0.698	0.763
RU-CLR	0.833	0.707	0.767	0.801
RU-CSVM	0.889	0.706	0.792	0.824
RENN-CLR	0.833	0.681	0.753	0.808
RENN-CSVM	0.889	0.495	0.663	0.718
RENN-CXGB	0.944	0.255	0.491	0.591
AKNN-CLR	0.833	0.682	0.754	0.808
AKNN-CSVM	0.944	0.421	0.630	0.684
AKNN-CXGB	0.944	0.363	0.585	0.639
AKNN-BBC	0.777	0.645	0.707	0.761
OSS-CLR	0.833	0.655	0.739	0.804
NCR-CLR	0.833	0.681	0.753	0.807
NCR-CSVM	0.778	0.676	0.725	0.802

The bold numbers in Table 13 indicate the best results for each performance metric of the CSA and undersampler-CSA models. It was generally observed that undersampler-CSA models had better performance than CSAs. Among the CSA models, CLR has the best recall score of 0.833, CXGB has the best specificity score of 0.875, and CLR has the best GM and AUC scores of 0.753 and 0.806, respectively. Among the undersampler-CSA models, RENN-CXGB, AKNN-CSVM, and AKNN-CXGB had a joint-best recall score of 0.944, followed by RU-CSVM and RENN-CSVM with a joint recall score of 0.889. RU-CLR had the best specificity score of 0.707, followed by RU-CSVM with a score of 0.706. RU-CSVM had the best GM score of 0.792, followed by RU-CLR with a score of 0.767. RU-CSVM had the best AUC score of 0.824, followed by RENN-CLR and AKNN-CLR with a joint score of 0.808.

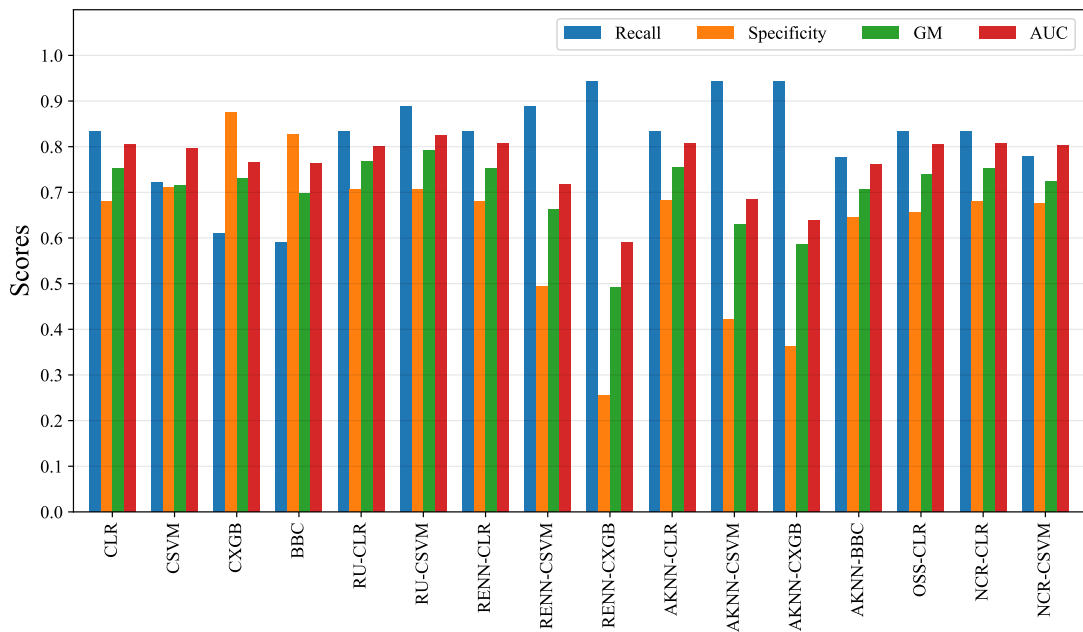


Fig. 21: Performance comparison of the models.

Similarly, it can be observed from Fig. 21 that the CLR outperforms other CSAs in three of four performance metrics. Undersampling of the non-dragging class did not improve the performance of CLR models. This can be attributed to the proficiency of CLR in effectively learning dragging class in the presence of noise and borderline samples in the imbalanced anchorage dataset.

The result in Fig. 21 demonstrate that the application of undersampling techniques significantly improved the recall scores of CSVM, CXGB, and BBC. This was anticipated, given that undersampling techniques seek to improve the prediction accuracy for the dragging class by eliminating noise or borderline samples in the non-dragging class, thereby addressing the class imbalance problem. Additionally, assigning a higher weight to dragging class misclassifications and penalizing the model more for dragging class incorrect predictions results in a model that prioritizes this class. This forces the model to learn the samples in the dragging class, resulting in a model that is skilled at predicting that class. For a CSA, the balanced recall and specificity score is a measure of the best trade-off between the FP and FN, and the RU-CSVM model excelled in this regard.

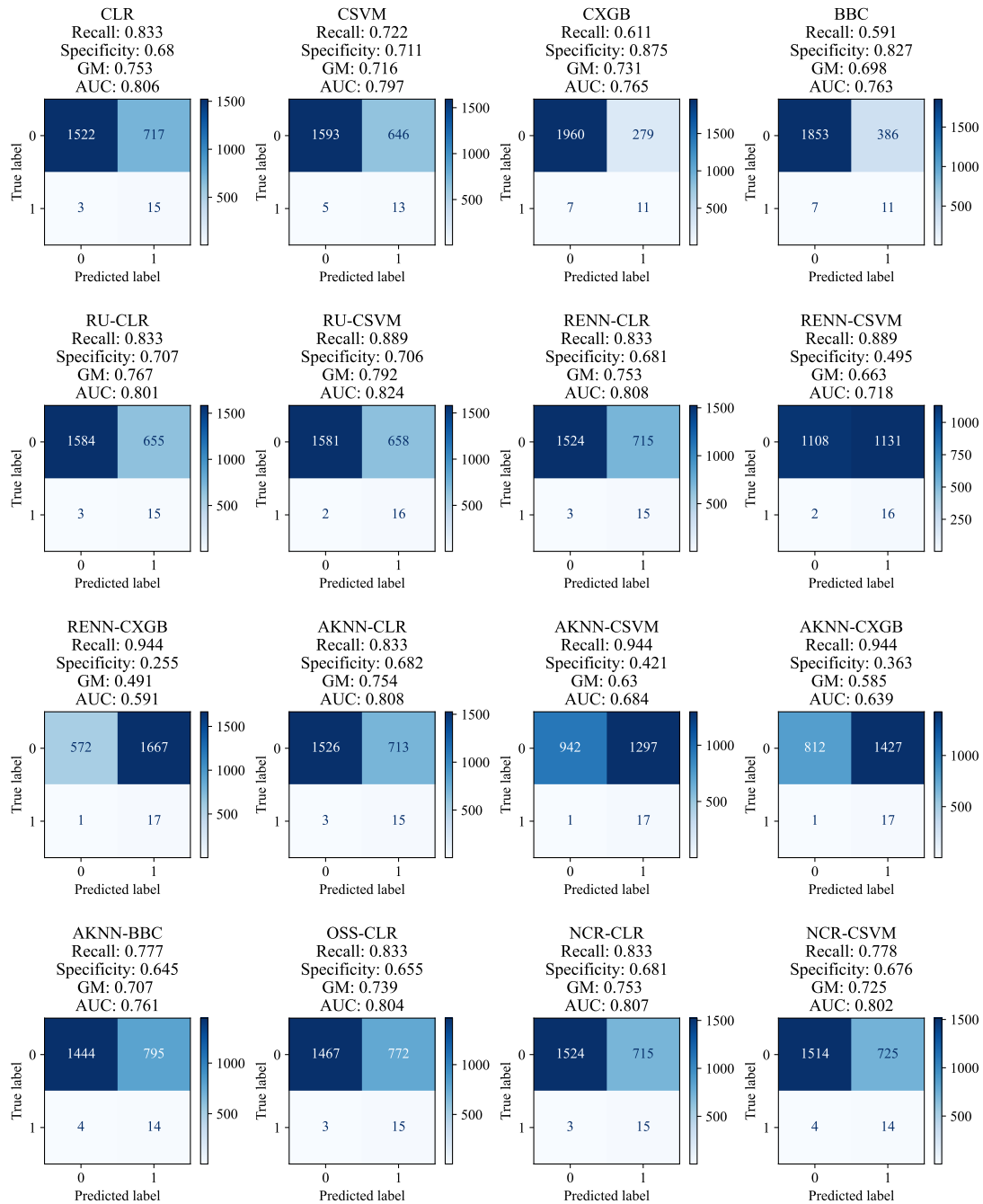


Fig. 22: Confusion matrix for the undersampler-CSA and CSA models from a test dataset of 2,257 samples consisting of 18 dragging and 2,239 non-dragging samples.

The confusion matrix in Fig. 22 shows the predicted number of misclassified samples, namely the FN and FP, as well as the correctly classified samples, namely the TP and TN. CLR had the lowest number of misclassified dragging samples and the highest number of misclassified non-dragging samples for the CSA models.

In addition, the RENN and AKNN undersamplers enhanced the classification accuracy of the dragging samples in CSVM, CXGB and BBC. It is worth noting that RENN and AKNN undersampling degraded the specificity score for the CSVM and CXGB. This is because less weight was assigned to the non-dragging class, resulting in the CSA paying less attention to the non-dragging class, leading to a model that is ineffective at predicting that class. BBC had the lowest number of misclassified non-dragging samples among the CSA models, whereas RU-CLR had the lowest number of misclassified non-dragging samples (FP). Undersampling was generally observed to enhance the recall performance of the CSAs except for CLR.

Based on the AUC scores in [Fig. 22](#), the RU undersampling technique improved the resilience of the CSVM in distinguishing between the non-dragging and dragging classes. RU-CSVM had the best AUC score of 0.824 as compared to CSVM which had an AUC score of 0.797.

In this study, dragging instances misclassified as non-dragging were assigned a higher misclassification cost. Therefore, the ideal model for predicting the dragging risk had a minimal occurrence of predicted FN, which is measured as the recall score. A higher recall score indicated that the model had an excellent ability to minimize the FN. Consequently, in addition to the recall score, a high AUC score indicated the robust ability of the model to distinguish non-dragging and dragging instance classes. Therefore, it can be concluded that of the joint top models by recall score, the AKNN-CSVM is the best due to its superior GM and AUC score, followed by the AKNN-CXGB. RU-CSVM model offers a balanced trade-off of the FN and FP from its superior balanced recall and specificity score measured by GM score.

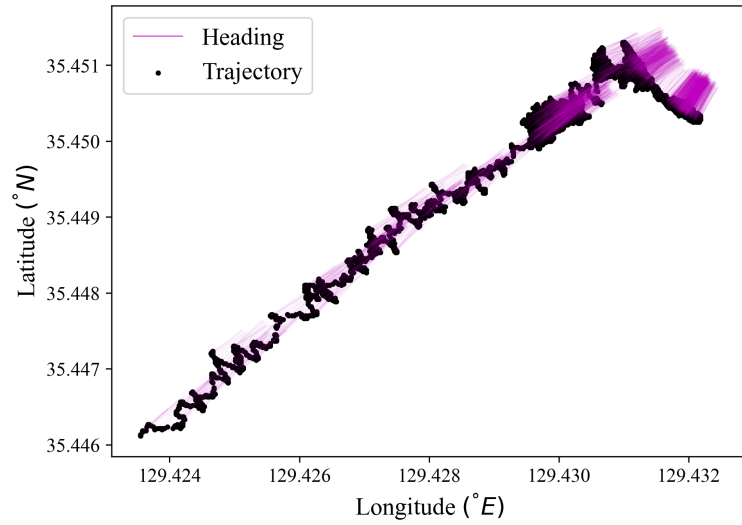
6.1 Case study validation

The prediction performance of the AKNN-CSVM, AKNN-CXGB, and RU-CSVM models were assessed through practical applications under three different anchorage scenarios. Fig. 23, Fig. 24, and Fig. 25 shows a visual representation comparing the trajectory of a ship at anchor with the predicted anchorage situation. A vessel at anchor transmits AIS data at intervals of 3 minutes. Therefore, it was feasible to compile the anchorage environmental features for the received AIS data and predict the anchorage situation. Prediction plots illustrated in Fig. 23(b), Fig. 24(b), and Fig. 25(b) shows that the prediction from the AKNN-CSVM model is consistent with the anchor situation in the trajectory plot Fig. 23(a), Fig. 24(a), and Fig. 25(a).

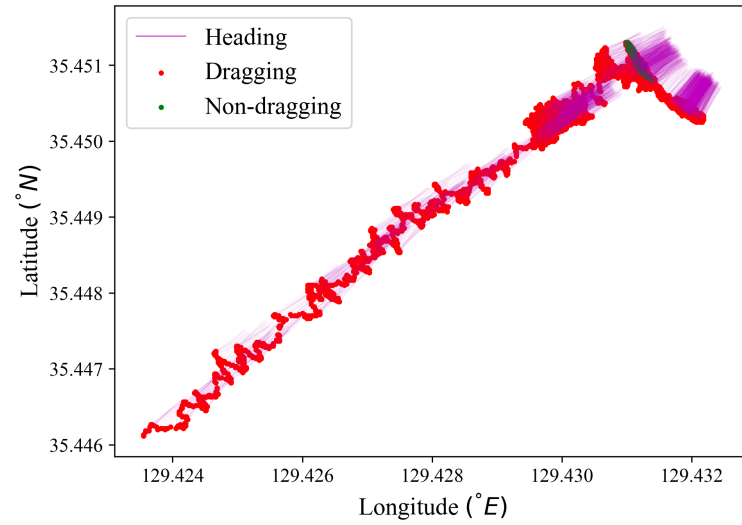
The prediction error refers to the observed dragging points in the trajectory plots that were predicted as non-dragging points by the ML model. The AKNN-CSVM and RU-CSVM models had the fewest prediction errors in case 1, as shown in Fig. 23(b) and (d). AKNN-CXGB had the most prediction errors in case 1 as shown in Fig. 23(c). Models AKNN-CSVM, AKNN-CXGB, and RU-CSVM had a significantly better prediction accuracy in case 2 as shown in Fig. 24(b),(c), and (d). The models were able to predict positions that had indicators of anchor dragging. The initial part of the trajectory can be assumed to be a time when the ship is preparing to anchor or it is carrying out the anchorage procedure.

AKNN-CSVM and RU-CSVM models had the best prediction accuracy in case 3 as illustrated in Fig. 25(b) and (d). The prediction plot in Fig. 25(b) when compared to Fig. 25(a) demonstrates that the AKNN-CSVM model has excellent prediction accuracy and robust performance in distinguishing the dragging and non-dragging instances. The RU-CSVM model has poor prediction accuracy. It can be observed that the model misclassified dragging points in Fig. 25(d). AKNN-CXGB has the worst classification performance in that it is unable to distinguish dragging and non-dragging points. Therefore, the AKNN-CSVM is the best classification model.

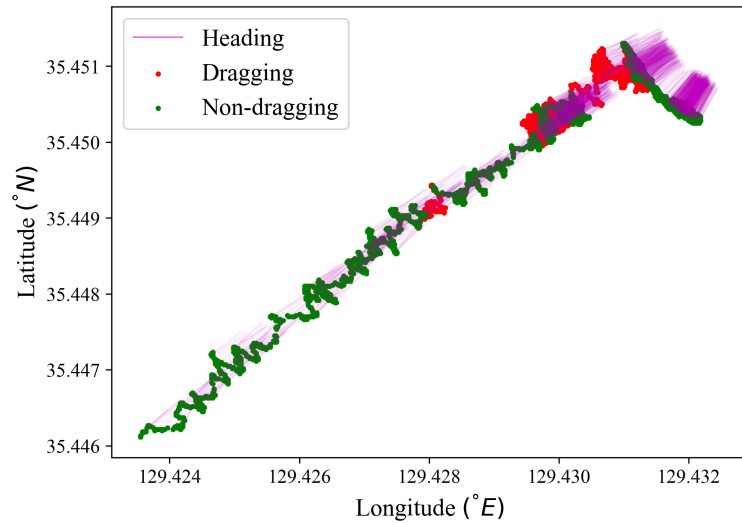
The model can function as a decision-support tool for the VTSO and OOW when monitoring vessels at anchor. Accurate and precise prediction of the anchor situation will provide timely identification of potential dragging risks, thereby effectively mitigating anchor-related incidents on time and promoting maritime traffic safety.



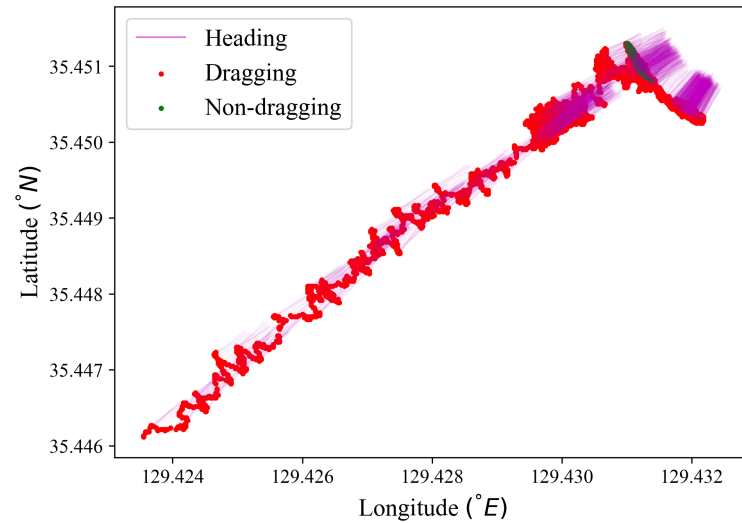
(a)



(b)

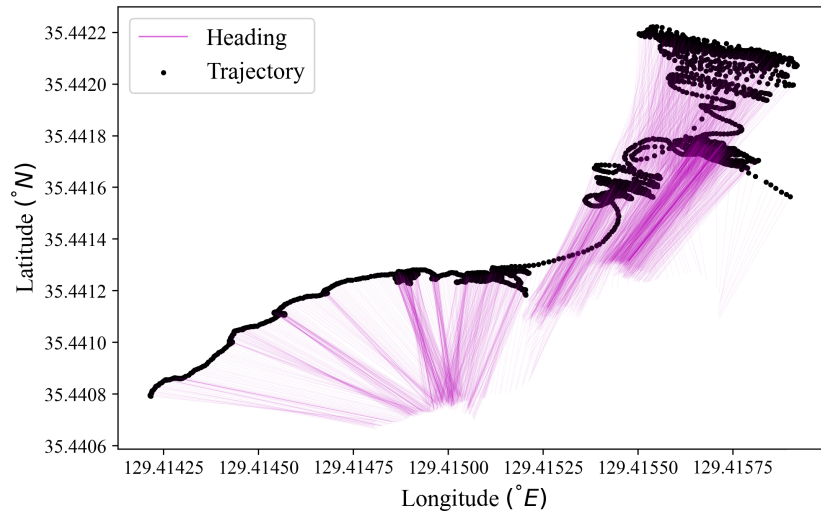


(c)

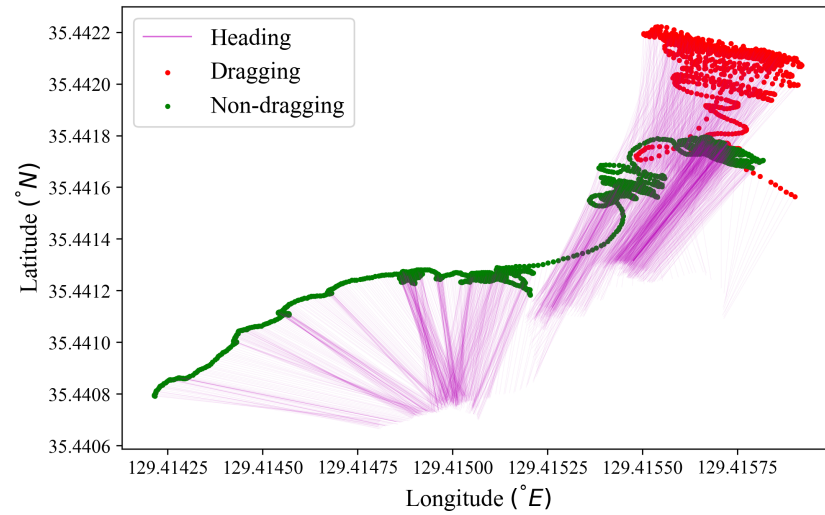


(d)

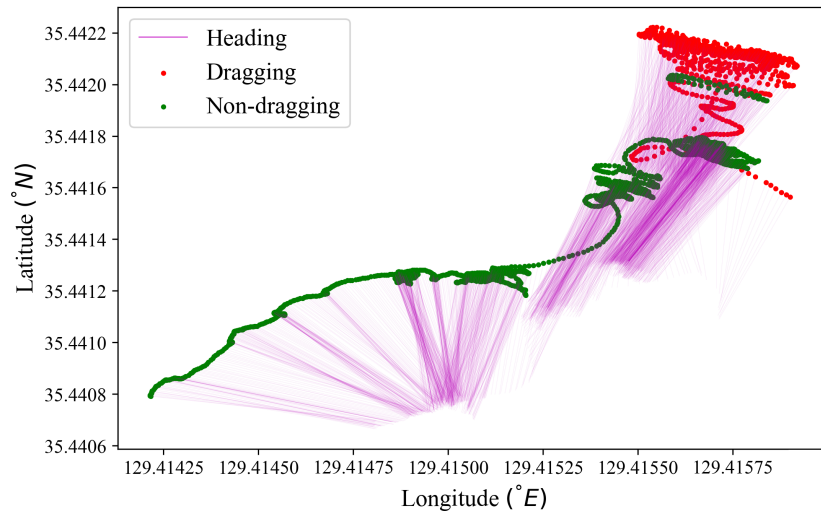
Fig. 23: Case 1 trajectory plot (a) from AIS data in comparison to prediction plot (b), (c), and (d) from AKNN-CSVM, AKNN-CXGB, and RU-CSVM models, respectively.



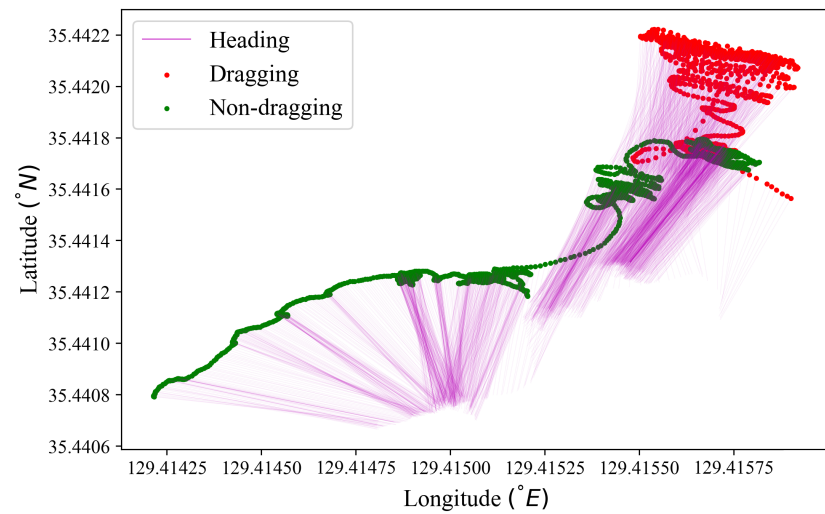
(a)



(b)

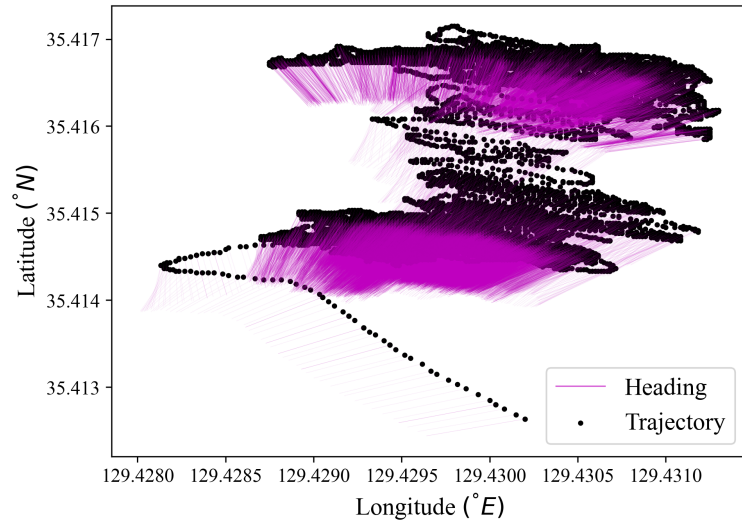


(c)

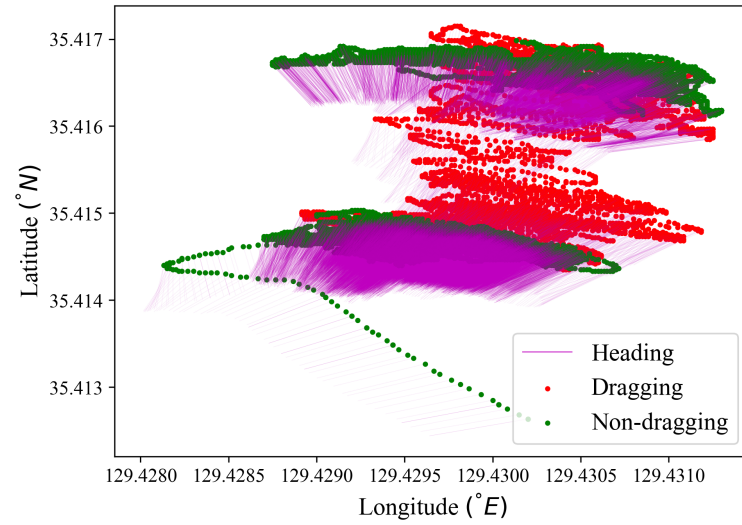


(d)

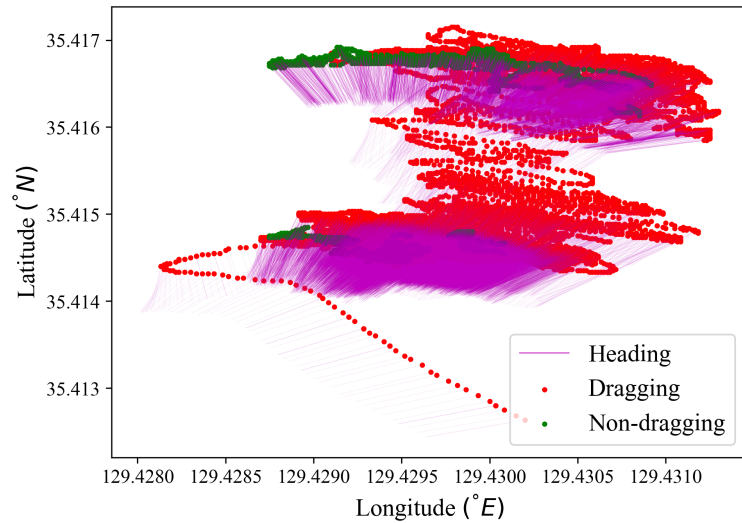
Fig. 24: Case 2 trajectory plot (a) from AIS data in comparison to prediction plot (b), (c), and (d) from AKNN-CSVM, AKNN-CXGB, and RU-CSVM models, respectively.



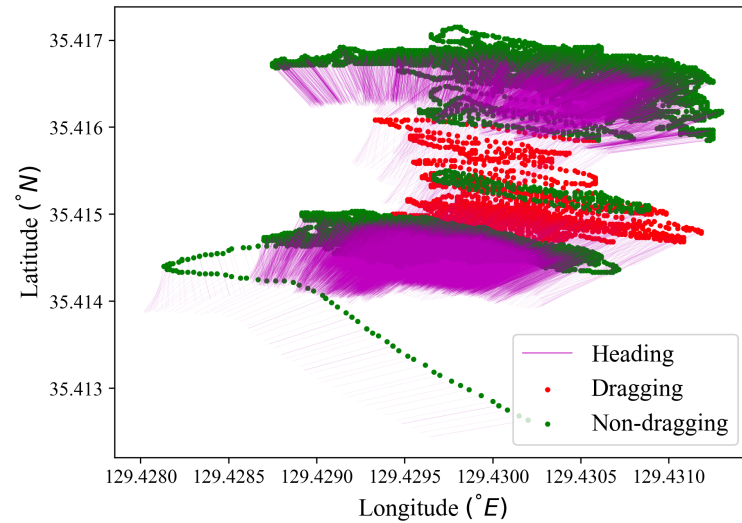
(a)



(b)



(c)



(d)

Fig. 25: Case 3 trajectory plot (a) from AIS data in comparison to prediction plot (b), (c), and (d) from AKNN-CSVM, AKNN-CXGB, and RU-CSVM models, respectively.

6.2 Risk criterion

[Kristiansen \(2013\)](#) defines safety as the degree of freedom from danger, and that concept of risk is used in evaluating safety. In addition, risk can be defined as an objective safety criteria computed from [Eqn. 11](#).

$$R = P \times C \quad (11)$$

where P is the probability of occurrence of an undesired event such as anchor dragging or collision and C is the consequences in terms of human, economic, or environmental loss. Risk is often calculated for all relevant hazards, where a hazard is a possible undesirable event and condition that may result in severity. A hazard with a high probability of occurrence and a high consequence has a high level of risk, and a high level of risk corresponds to a low level of safety, and vice versa.

In this study, the undesired event, which is the dragging of anchor, can be quantified as a probability from the ML model. In addition, we assume that the consequences of the dragging anchor have been established; therefore, C is assigned a maximum probability of 1, thus the risk is computed as probability given by $R = P$.

A high risk is denoted by a probability of 1, while a low risk is represented by a probability of 0. Determining the boundary between acceptable/unacceptable or safe/unsafe can be quite a challenge. Trial and error, expert opinion, experiments, rank mapping, and other methods have been used to determine the acceptable boundary in risk ranking. [Inoue \(2000\)](#) used a combination of expert opinion questionnaires and navigator ship handling stress to develop the environmental stress (ES) model rank. PARK risk was formulated through a questionnaire analysis of the opinions and experiences of Korean crew members ([Park et al., 2013](#)). [Yoo and Lee \(2019\)](#) developed collision risk (CoRI) model rank from mapping the ES model risk rank as illustrated in [Table 14](#).

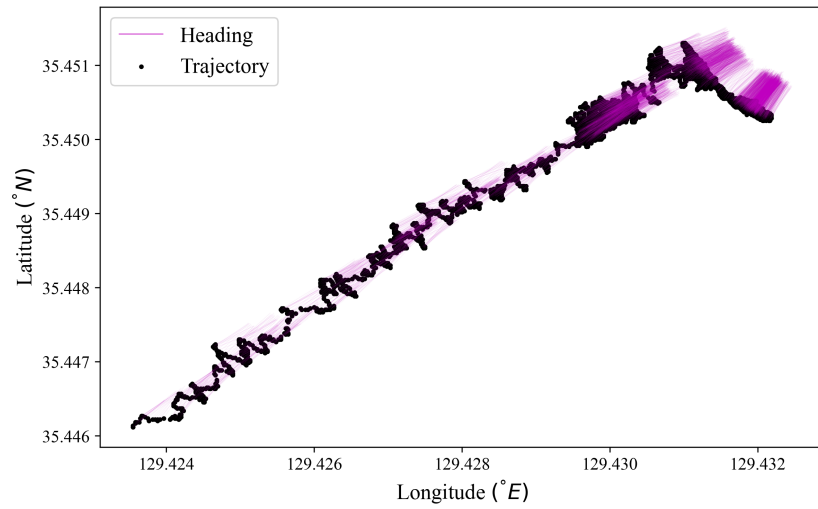
Classification algorithms have a decision boundary known as the threshold, which is used by the algorithm in determining the classification of each predicted probability using an activation function such as sigmoid or softmax. [Table 14](#) shows this study's risk ranking in comparison to ES, PARK and CoRI models.

Table 14: Risk ranking comparing ES, PARK, CoRI, and ML models.

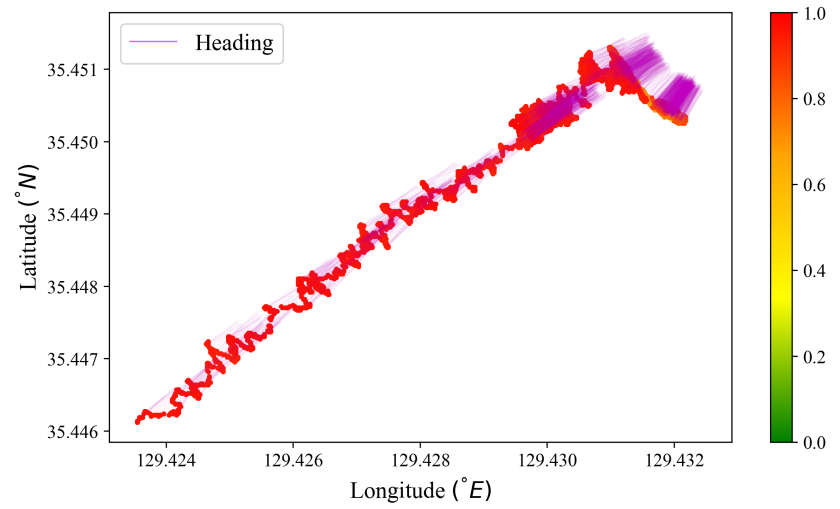
SJ	Mariner's judgement	ML model	CoRI Model	Evaluate $\sum[SJ]_i$	Park model	Stress rank	Acceptance criteria
0	Extremely safe	0 – 0.1	0 – 2	0	0	Negligible	Acceptable
1	Fairly safe	0.1 – 0.3					
2	Somewhat safe	0.3 – 0.4					
3	Safe nor dangerous	0.4 – 0.5	2 – 3	500	4	Marginal	
4	Somewhat dangerous	0.5 – 0.6	3 – 4	750	5		
5	Fairly dangerous	0.6 – 0.8	4 – 5	900	6	Critical	Unacceptable
6	Extremely dangerous	0.8 – 1	5 – 6	1000	7	Catastrophic	

This study utilizes the ML threshold as the risk rank boundary. The default value of the threshold for ML algorithms is 0.5. The value of the threshold can be adjusted on a ROC or precision-recall curve, for an optimal threshold value. Risk provides a better alternative for measuring dragging risk. Classification of predicted anchorage cases on a binary scale has the potential of overloading the VTSO or OOW with alarm information, especially for low-risk-boundary cases that are predicted as dragging, when in reality the events causing the dragging of the anchor are not in catastrophic proportion. Therefore, the dragging risk criterion offers relief from alarming information when monitoring ships at anchor by filtering out high-risk dragging predictions from low-risk cases.

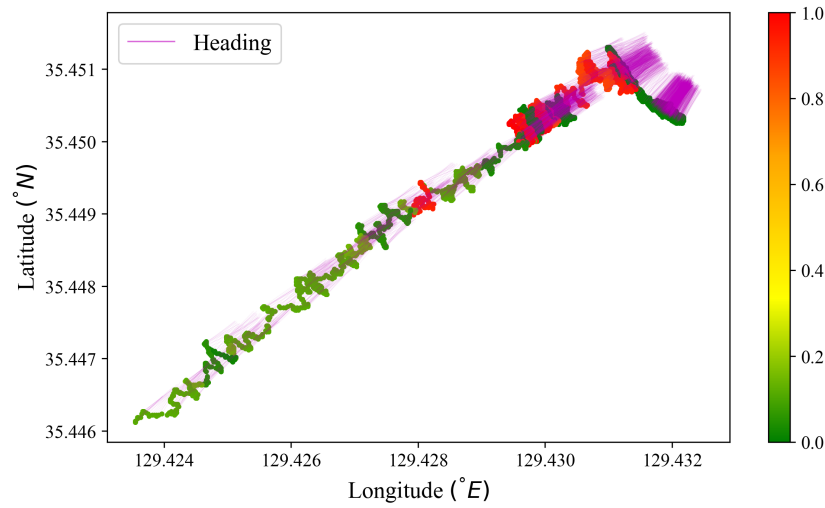
The risk is divided into 7 categories according to the subjective judgment (SJ) scores, as highlighted in Table 14. The 7 categories were ranked as per stress levels and a color palette was assigned to each rank as green, yellow, orange, and red, assigned to negligible, marginal, critical, and catastrophic, respectively, for ease of visualization of the prediction plots. Fig. 26, Fig. 27, and Fig. 28 is a demonstration of the dragging risk prediction on the three cases from AKNN-CSVM, AKNN-CXGB, and RU-CSVM models. It can be observed that the RU-CSVM model is excellent at risk profiling anchor trajectory according to the anchor situation, especially for case 2 and case 3 in Fig. 27 and Fig. 28 respectively. The RU-CSVM model has excellent and robust risk profiling accuracy for the three cases, and therefore it is the best ML model for estimation of anchor dragging risk for ships at anchor.



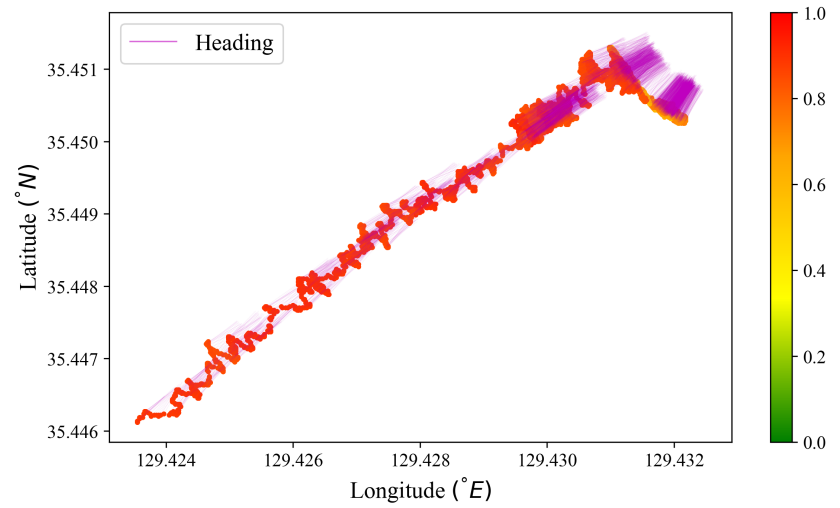
(a)



(b)



(c)



(d)

Fig. 26: Case 1 trajectory plot (a) from AIS data in comparison to dragging risk plot (b), (c), and (d) from AKNN-CSVM, AKNN-CXGB, and RU-CSVM models, respectively.

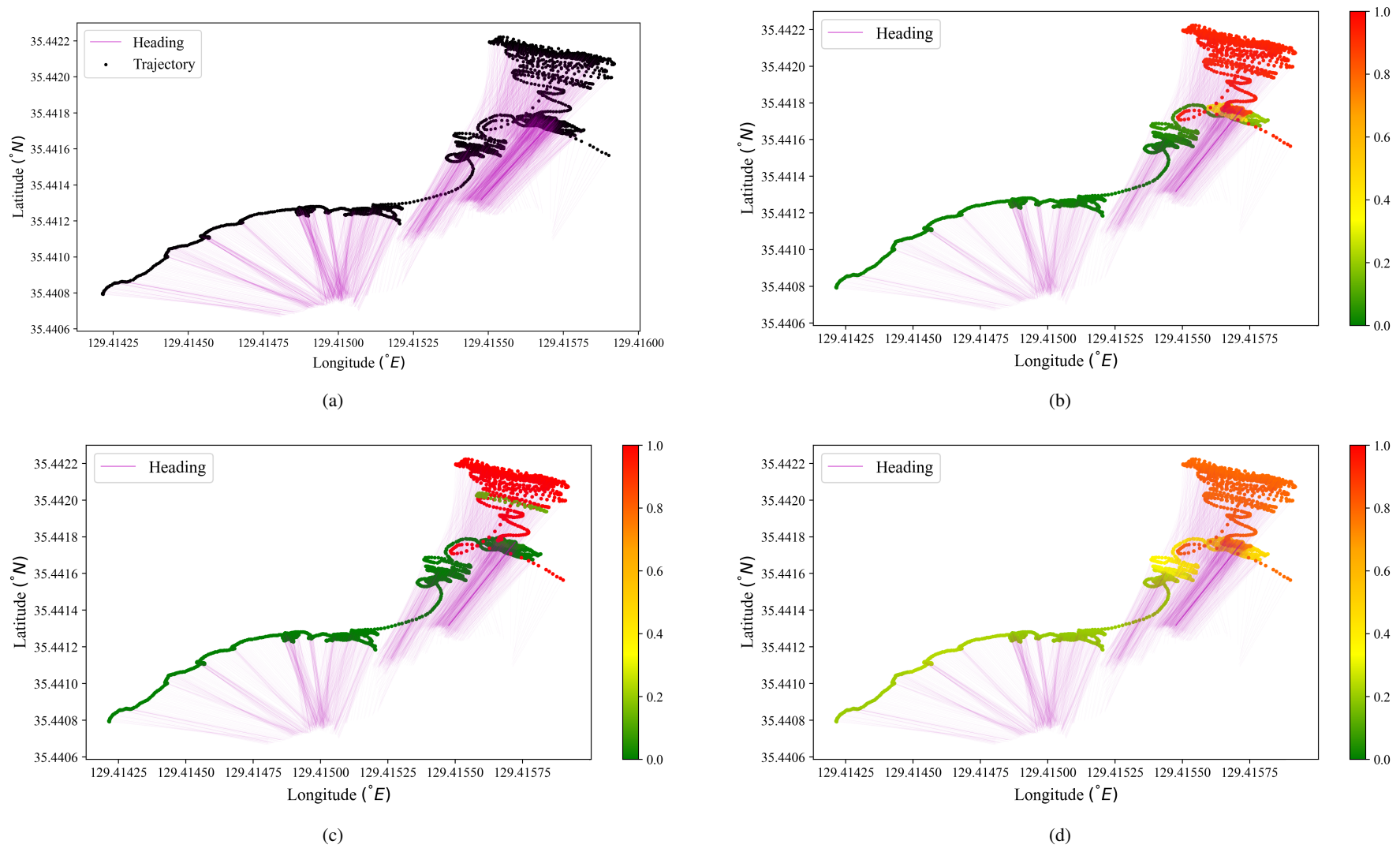
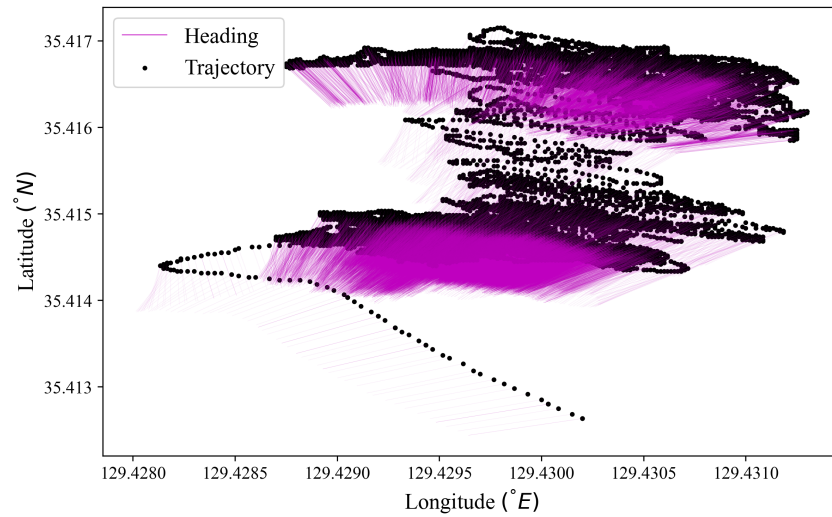
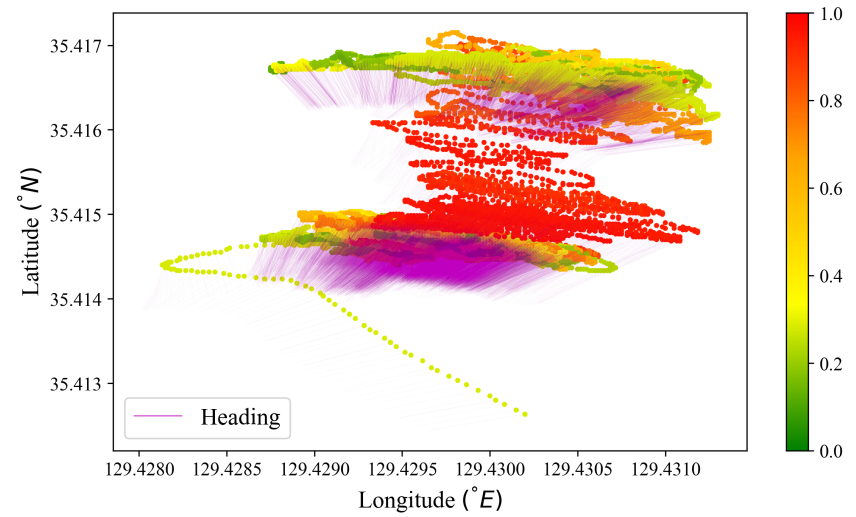


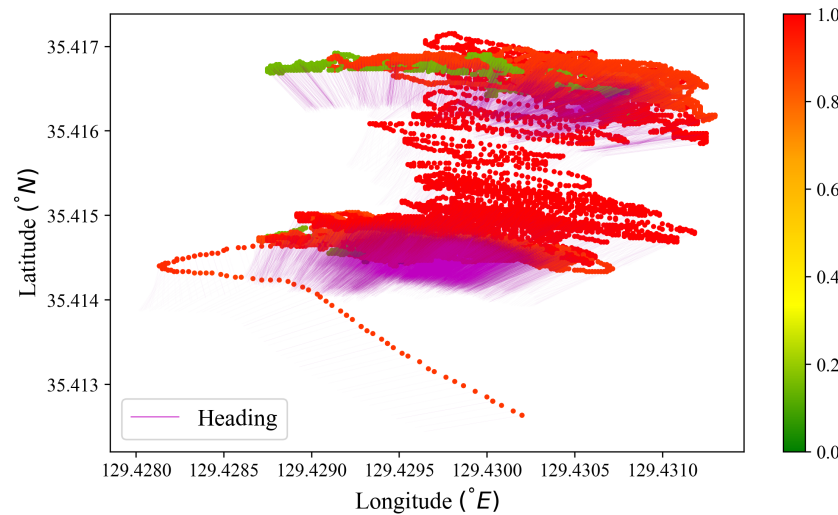
Fig. 27: Case 2 trajectory plot (a) from AIS data in comparison to dragging risk plot (b), (c), and (d) from AKNN-CSVM, AKNN-CXGB, and RU-CSVM models, respectively.



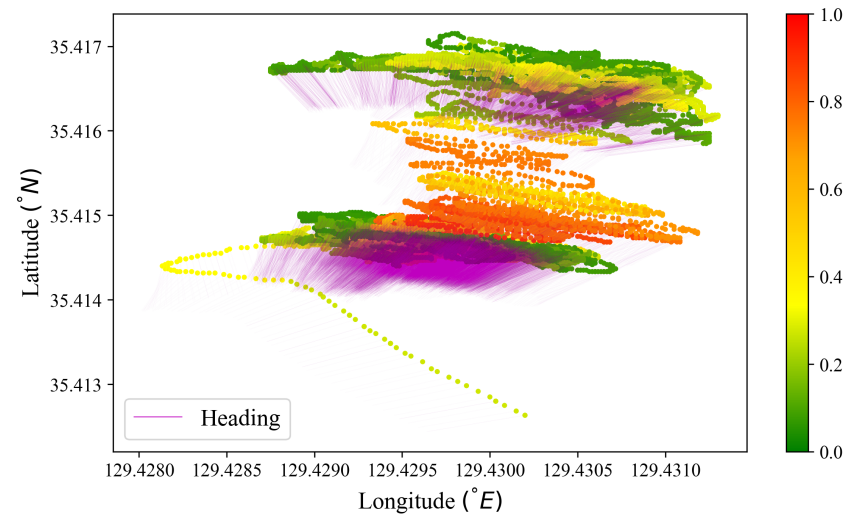
(a)



(b)



(c)



(d)

Fig. 28: Case 3 trajectory plot (a) from AIS data in comparison to dragging risk plot (b), (c), and (d) from AKNN-CSVM, AKNN-CXGB, and RU-CSVM models, respectively.

6.3 Anchor safety assessment at anchorage areas

The RU-CSVM model can be used for real-time monitoring of ships at anchorage areas by VTSO and monitoring the anchor situation of own ship by OOW when at anchor. The input data for the model can be obtained automatically from the ship sensors and ENC, or electronic chart display and information system (ECDIS), if the model is installed onboard the ship. For the VTS, input data can be automatically extracted from AIS data received from ships at the anchorage area, ENC/ECDIS, and meteorological station sensors, collated, and used to estimate the anchor dragging risk of the ships at anchor.

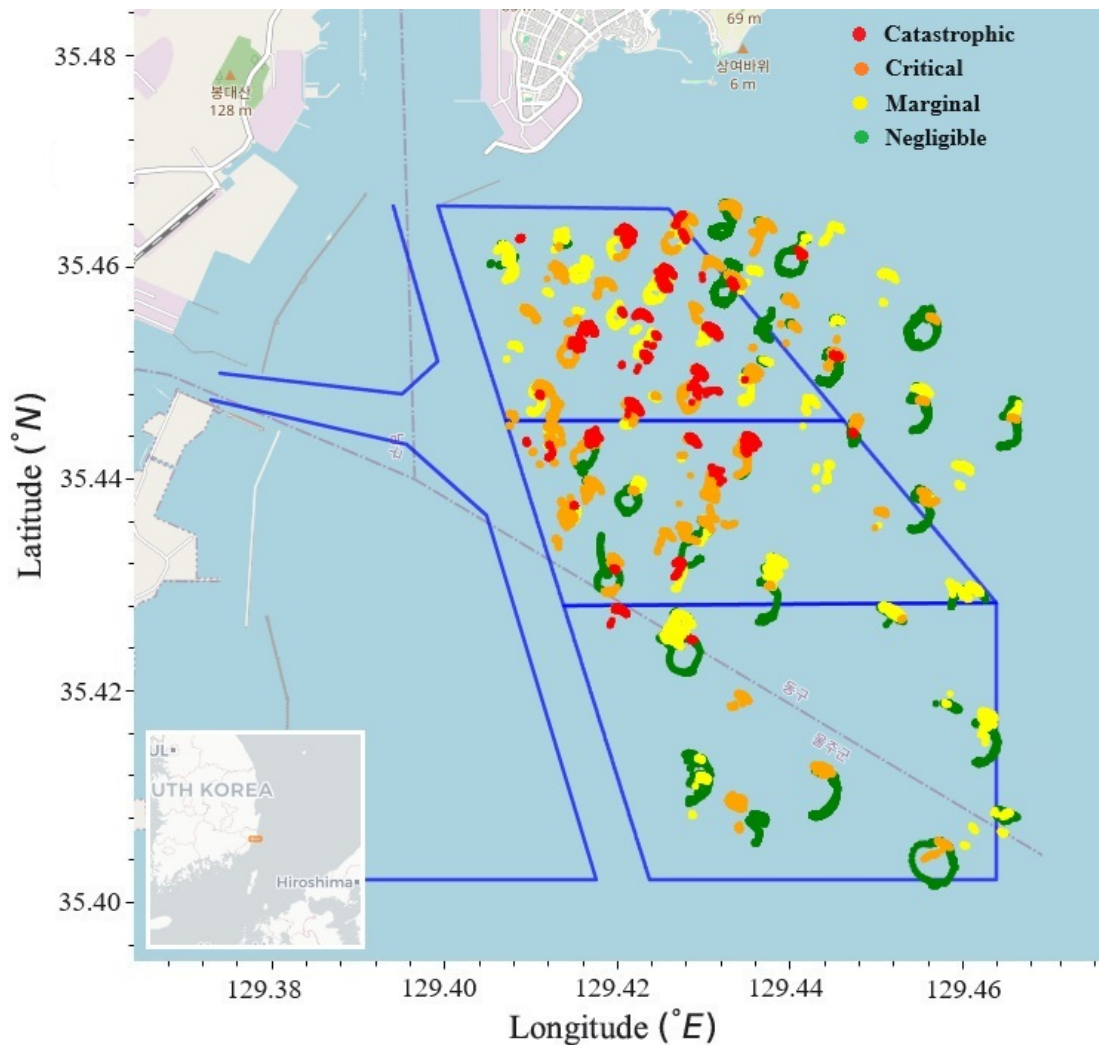


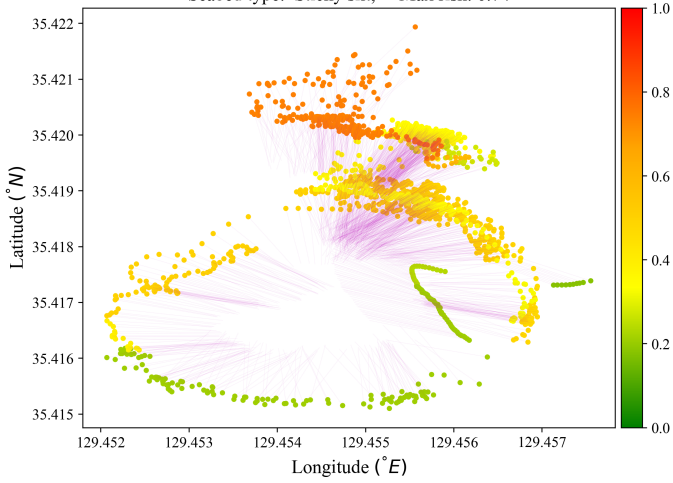
Fig. 29: Real-time estimation of dragging risk for vessels at anchorage for three days using the RU-CSVM model.

Fig. 29 is a real-time impression of anchor dragging risk assessment of vessels at anchor for the period of 13th April 2017 to 15th April 2017. The anchor dragging risk is presented as a stress rank. It is observed that the RU-CSVM model is excellent at risk profiling the anchor trajectories for quantitative assessment of anchorage safety. The anchor dragging risk stress rank can be used for decision-making when a risk threshold is set up by the VTSO.

Fig. 30 and Fig. 31 shows a real-time demonstration of dragging risk estimation from RU-CSVM model. Each of the plots shows the risk value for the trajectory points, the ship heading for each point, anchor conditions for recorded maximum dragging, and the color bar to categorize the risk-rank values. The wind speed, ship length, and depth of sea have superior predictive power in discriminating between dragging and non-dragging instances, as established in the class distribution and overlap of features in Fig. 8. It is observed that generally, risk plots with wind speed, ship length, and sea depth above mean values of 27 *m/s*, 100 *m*, and 49 *m* respectively, for dragging cases recorded risk values above 0.6 in the unacceptable critical and catastrophic stress rank.

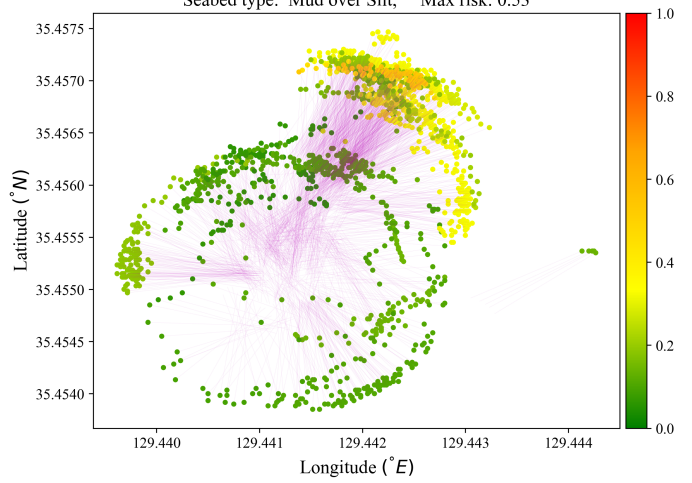
Accurate estimation of dragging risk is vital for anchor watch safety and the timely identification of a high-risk event for aversion to prevent anchor dragging, thereby promoting harbor, traffic, and anchorage area safety. Furthermore, the model can be used to assess anchorage safety on prevailing or forecasted meteorological conditions for different categories of vessels, as demonstrated in Fig. 30 and Fig. 31. Anchorage areas that are often underutilized, especially if the area has a history of frequent anchor dragging, can make use of the dragging risk model to assess the optimal utilization rate of the anchorage area.

Ship type: Tanker, Ship length: 183m, Ship draft: 12.0m,
 Sea depth: 68m, Wind speed: 37.71m/s, Wind direction: 227.0°,
 Seabed type: Sticky silt, Max risk: 0.74



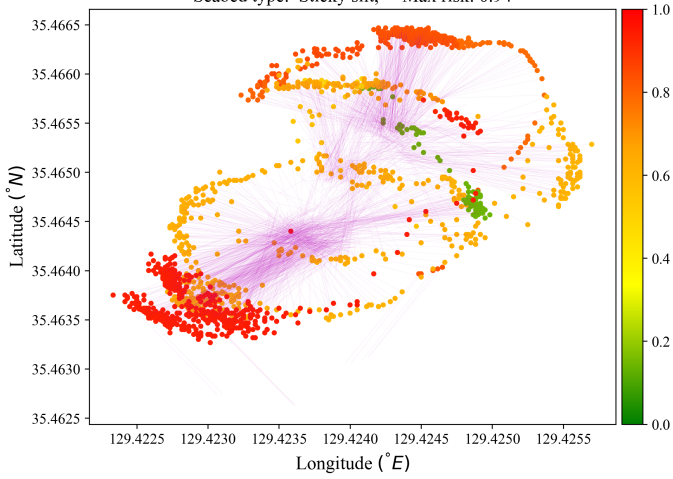
(a)

Ship type: Tanker, Ship length: 116m, Ship draft: 8.0m,
 Sea depth: 58m, Wind speed: 30.32m/s, Wind direction: 209.0°,
 Seabed type: Mud over Silt, Max risk: 0.53



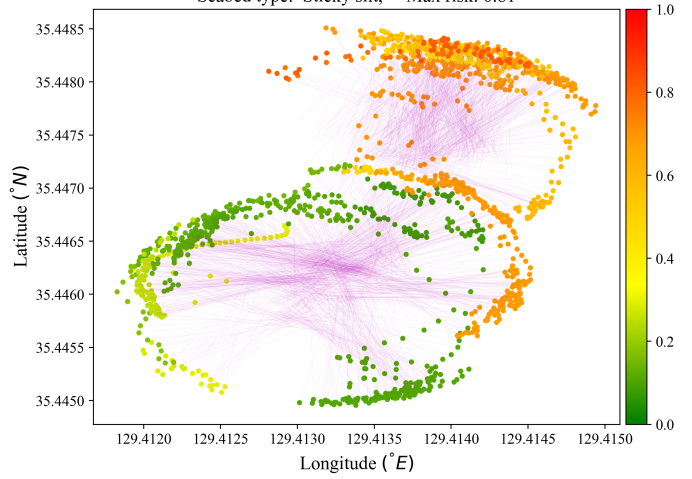
(b)

Ship type: Tanker, Ship length: 99m, Ship draft: 5.0m,
 Sea depth: 40m, Wind speed: 27.8m/s, Wind direction: 34.0°,
 Seabed type: Sticky silt, Max risk: 0.94



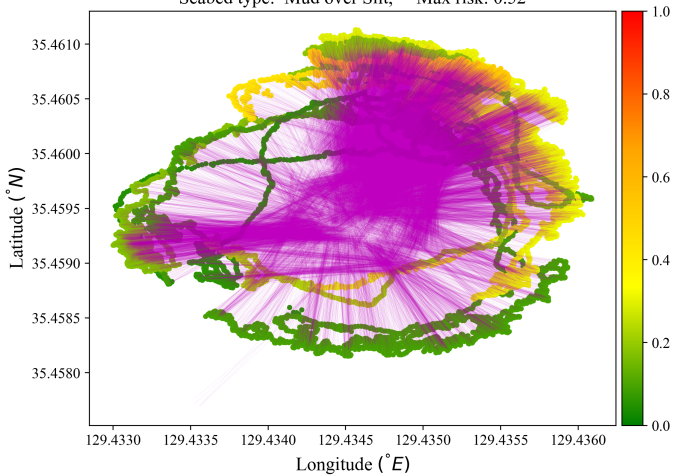
(c)

Ship type: Tanker, Ship length: 97m, Ship draft: 5.0m,
 Sea depth: 38m, Wind speed: 30.32m/s, Wind direction: 209.0°,
 Seabed type: Sticky silt, Max risk: 0.81



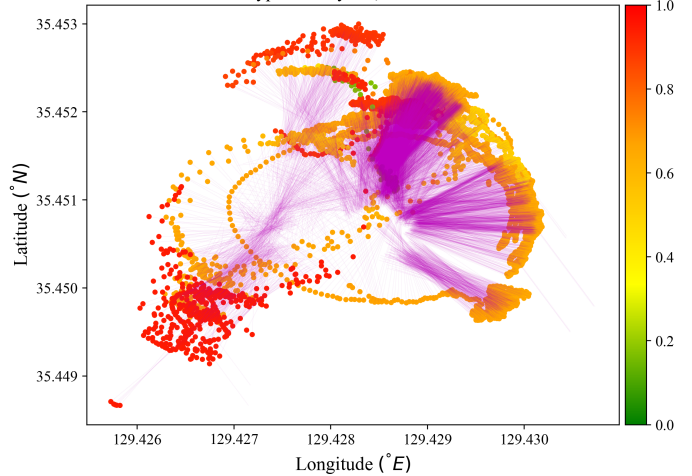
(d)

Ship type: Tanker, Ship length: 112m, Ship draft: 7.0m,
 Sea depth: 58m, Wind speed: 30.32m/s, Wind direction: 209.0°,
 Seabed type: Mud over Silt, Max risk: 0.52



(e)

Ship type: Tanker, Ship length: 118m, Ship draft: 5.0m,
 Sea depth: 48m, Wind speed: 27.8m/s, Wind direction: 34.0°,
 Seabed type: Sticky silt, Max risk: 0.94



(f)

Fig. 30: Monitoring the anchor dragging risk of tankers anchoring at Ulsan E-group anchorage area.

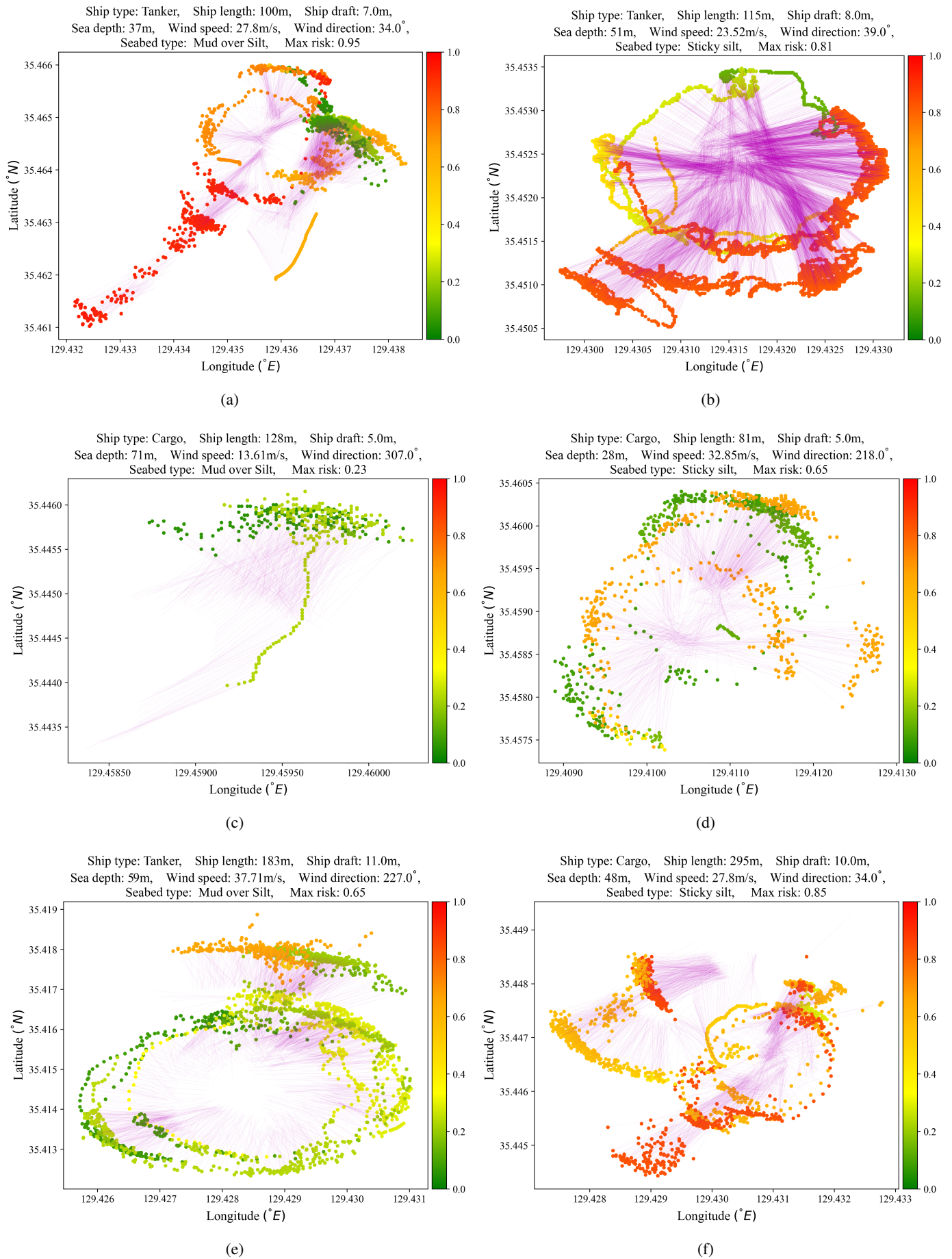


Fig. 31: Monitoring the anchor dragging risk of cargo and tanker ships anchoring at Ulsan E-group anchorage area.

7 Conclusion and recommendations

7.1 Conclusion

The VTSO and OOW closely monitor a ship at anchor to identify any instances of anchor dragging promptly so that precautionary measures can be taken to avert anchor dragging-related accidents such as collisions with traffic, collisions with other ships at anchor, and damage to seabed infrastructure, including pipelines and cables. This study has presented a ML strategy that predicts the dragging risk of ships at anchor using historical AIS data, hydrographic publications, and meteorological data, thereby promoting maritime traffic and ship safety at anchorage and in harbors. Data resampling techniques in combination with the CSA were used to develop ML models that estimate the dragging risk of a vessel at anchor. A comprehensive analysis was conducted on the performance of the designed undersampler-CSA models in comparison to CSA models. The AKNN-CSVM, AKNN-CXGB, and RU-CSVM exhibited superior performance compared to the other models. In addition, the AKNN-CSVM model exhibited excellent classification prediction accuracy and robustness in distinguishing dragging and non-dragging classes when evaluated using practical anchorage data on binary classification. Moreover, RU-CSVM exhibited excellent risk profiling accuracy on the anchor trajectory dataset. In summary, optimized AKNN undersampler in combination with CSVM algorithm is recommended for the classification of dragging cases, whereas RU in combination with CSVM is proposed for risk profiling according to the risk ranks.

The main contributions of this study are summarized as follows: The study introduces the literature on the application of ML to estimate anchor dragging risk for ships at anchor. This study elucidates and demonstrates the application of data undersampling techniques and cost-sensitive learning in designing a ML model from anchorage dataset. This study investigates the factors that contribute to anchor dragging with a focus on extracting anchorage data features. In addition, this study presents a detailed sequential description of the methodology employed to collate anchorage data by integrating historical AIS data, hydrographic, and meteorological data. The research target group was limited to ocean-going ships, owing to the ease of availability, accessibility, and affordability of data from the respective governing administrations. The performance of the designed undersampler-CSA

models was compared with that of the CSA models to highlight the superior prediction performance. This study can be used by vessel traffic management authorities and shipowners as a real-time decision-support tool for monitoring ships at anchorage areas. Vessel traffic administrations can utilize the model to assess and advice ships at anchor regarding safety level of anchorage area based on prevailing and forecasted meteorological conditions. The study offers a departure approach from numerical and computational modeling of dragging risk to the currently embraced ML for digital shipping 4.0 and the maritime internet of things. The ML approach offers flexibility to include features that can be quite complex to model using a numerical or computational approach.

7.2 Recommendation

The ML model from this study can be used as a decision-support tool by vessel traffic management authorities to monitor ships in anchorage areas and by shipowners to monitor the anchor situation when the ship is at anchor. It has been established that datasets can achieve improved classification results by introducing additional features other than resampling techniques. This study was constrained by its reliance on readily available historical AIS sensor data and hydro-meteorological data that can be sourced from relevant governing administrations, excluding critical information on the anchor system, such as the dynamics among the anchor, chain, and seafloor. This information was omitted from this study because it is not publicly available and is not mandatory information to be shared by the navigation officers with the public. The anchor-chain-seafloor interaction dynamics define the anchor holding power. Hence, the model performance can be enhanced by adding information on the anchor type, position of the anchor drop, length of the chain paid out, and other salient features to the existing dataset when designing an improved ML model. Future research may attempt to improve the model and address the aforementioned difficulties to achieve better prediction accuracy.

APPENDICES

Dragging data details

Datetime	Ship type	Length	Draught	Depth	Seabed type	Wind speed	Wind direction
2017-01-22 16:00:00	Tanker	93.0	6.0	32.0	Sticky silt	24.298	308.0
2017-03-30 04:00:00	Tanker	183.0	8.0	55.0	Mud over Silt	20.022	37.0
2017-01-04 00:00:00	Cargo	200.0	8.0	38.0	Sticky silt	27.797	307.0
2017-08-04 01:00:00	Tanker	100.0	7.0	62.0	Mud over Silt	27.797	34.0
2017-03-30 00:00:00	Tanker	241.0	6.0	40.0	Sticky silt	20.0216	37.0
2017-11-04 00:00:00	Tanker	119.0	5.0	50.0	Sticky silt	34.0172	211.0
2017-03-12 00:00:00	Tanker	180.0	12.0	69.0	Mud over Silt	26.825	341.0
2017-09-15 03:00:00	Tanker	106.0	4.0	35.0	Sticky silt	24.104	74.0
2017-07-22 22:00:00	Tanker	113.0	4.0	48.0	Mud over Silt	18.661	207.0
2017-07-16 14:00:00	Cargo	190.0	12.0	63.0	Sticky silt	17.495	161.0
2017-04-05 00:00:00	Tanker	170.0	7.0	71.0	Mud over Silt	23.132	316.0
2017-04-13 08:00:00	Cargo	102.0	6.0	32.0	Sticky silt	34.017	211.0
2017-04-13 00:00:00	Tanker	225.0	8.0	46.0	Mud over Silt	34.017	211.0
2017-12-31 12:00:00	Tanker	90.0	5.0	33.0	Mud over Silt	23.909	341.0
2018-04-22 00:00:00	Tanker	100.0	4.0	46.0	Mud over Silt	35.184	31.0
2018-05-03 00:00:00	Tanker	80.0	5.0	28.5	Sticky silt	42.570	16.0
2018-07-05 08:00:00	Tanker	118.0	5.0	28.0	Sticky silt	25.853	29.0
2018-06-07 07:00:00	Tanker	128.0	7.0	59.0	Sticky silt	33.434	29.0

Datetime	Ship type	Length	Draught	Depth	Seabed type	Wind speed	Wind direction
2018-05-07 10:00:00	Tanker	90.0	4.0	48.0	Sticky silt	31.490	41.0
2018-04-28 04:00:00	Tanker	74.0	4.0	42.0	Mud over Silt	29.352	197.0
2018-06-07 19:00:00	Tanker	96.0	6.0	38.0	Mud sand and shell	31.490	41.0
2018-07-29 00:00:00	Cargo	200.0	9.0	63.0	Sticky silt	35.961	40.0
2018-12-25 13:00:00	Tanker	180.0	11.0	65.0	Mud over Silt	36.350	124.0
2018-11-11 09:00:00	Tanker	101.0	4.0	60.0	Mud over Silt	24.104	171.0
2018-11-22 00:00:00	Tanker	100.0	6.0	40.0	Sticky silt	29.158	336.0
2018-08-15 00:00:00	Cargo	79.0	4.0	28.5	Sticky silt	35.767	224.0
2018-06-07 00:00:00	Cargo	295.0	10.0	58.0	Mud over Silt	33.434	29.0
2021-11-27 13:00:00	Tanker	124.0	8.0	58.0	Mud over Silt	26.242	170.0
2021-11-27 10:00:00	Tanker	107.0	6.0	60.0	Mud over Silt	26.242	170.0
2021-12-29 00:00:00	Tanker	118.0	5.0	48.0	Sticky silt	19.827	310.0
2021-10-14 17:00:00	Tanker	119.0	8.0	58.0	Mud over Silt	16.911	343.0
2021-10-10 12:00:00	Tanker	115.0	5.0	58.0	Mud over Silt	22.354	65.0
2021-12-28 14:00:00	Tanker	112.0	5.0	33.0	Mud over Silt	19.633	299.0
2021-06-11 12:00:00	Cargo	229.0	14.0	38.0	Sticky silt	19.438	80.0
2021-01-08 00:00:00	Tanker	93.0	5.0	50.0	Sticky silt	36.739	189.0
2021-01-07 16:00:00	Cargo	190.0	8.0	69.0	Mud over Silt	28.769	191.0
2021-11-08 23:00:00	Tanker	244.0	8.0	79.0	Mud	23.520	156.0
2021-05-14 00:00:00	Tanker	96.0	5.0	38.0	Sticky silt	19.632	167.0
2021-05-27 23:00:00	Tanker	115.0	8.0	60.0	Mud over Silt	26.241	171.0
2021-04-22 13:00:00	Cargo	179.0	5.0	52.0	Mud over Silt	27.602	64.0

Datetime	Ship type	Length	Draught	Depth	Seabed type	Wind speed	Wind direction
2021-01-05 00:00:00	Tanker	99.0	5.0	51.0	Sticky silt	27.991	177.0
2021-07-06 14:00:00	Tanker	100.0	5.0	28.5	Sticky silt	18.272	164.0
2021-04-19 14:00:00	Cargo	182.0	8.0	69.0	Mud over Silt	23.714	35.0
2021-07-04 00:00:00	Tanker	101.0	4.0	50.0	Sticky silt	27.602	64.0
2021-04-20 12:00:00	Tanker	159.0	6.0	65.0	Mud over Silt	23.715	35.0
2021-04-23 10:00:00	Tanker	91.0	6.0	38.0	Sticky silt	25.659	52.0
2021-02-04 19:00:00	Cargo	169.0	9.0	47.0	Silt shells	25.464	52.0
2019-04-01 00:00:00	Tanker	115.0	8.0	50.0	Sticky silt	27.797	317.0
2019-05-01 13:00:00	Tanker	100.0	5.0	38.0	Sticky silt	21.771	153.0
2019-09-01 21:00:00	Tanker	100.0	5.0	40.0	Sticky silt	25.464	319.0
2019-03-05 07:00:00	Cargo	79.0	5.0	58.0	Mud over Silt	17.300	59.0
2019-01-05 00:00:00	Tanker	111.0	5.0	48.0	Sticky silt	24.881	202.0
2019-03-04 02:00:00	Tanker	100.0	5.0	46.0	Mud over Silt	30.713	33.0
2019-02-18 13:00:00	Tanker	183.0	12.0	59.0	Sticky silt	39.071	316.0
2019-01-23 10:00:00	Tanker	101.0	7.0	38.0	Sticky silt	20.993	351.0
2019-02-02 08:00:00	Tanker	113.0	4.0	35.0	Sticky silt	26.436	136.0
2019-08-04 00:00:00	Tanker	89.0	5.0	48.0	Sticky silt	36.739	191.0
2019-03-21 00:00:00	Tanker	228.0	13.0	66.0	Sticky silt	31.685	33.0
2019-04-17 07:00:00	Tanker	105.0	6.0	37.0	Mud over Silt	32.074	28.0
2019-08-03 17:00:00	Tanker	90.0	5.0	44.0	Mud over Silt	26.631	226.0
2019-04-02 17:00:00	Tanker	90.0	5.0	60.0	Mud over Silt	22.354	35.0
2019-12-16 00:00:00	Tanker	90.0	7.0	35.0	Sticky silt	20.410 24.0	
2019-11-14 13:00:00	Tanker	96.0	5.0	35.0	Sticky silt	19.244 15.0	

Datetime	Ship type	Length	Draught	Depth	Seabed type	Wind speed	Wind direction
2021-01-05 00:00:00	Tanker	99.0	5.0	51.0	Sticky silt	27.991	177.0
2019-11-23 00:00:00	Tanker	99.0	4.0	38.0	Sticky silt	30.324	51.0
2019-11-12 06:00:00	Tanker	183.0	11.0	65.0	Mud over Silt	16.134	269.0
2019-10-17 03:00:00	Tanker	93.0	5.0	58.0	Mud over Silt	26.436	68.0
2019-10-21 00:00:00	Cargo	134.0	8.0	69.0	Mud over Silt	22.549	224.0
2019-05-24 00:00:00	Tanker	186.0	11.0	59.0	Mud over Silt	35.378	206.0
2019-04-05 00:00:00	Cargo	110.0	4.0	58.0	Mud over Silt	25.270	50.0
2019-08-05 11:00:00	Tanker	66.0	3.0	52.0	Mud over Silt	23.520	226.0
2020-05-22 15:00:00	Cargo	124.0	7.0	71.0	Mud over Silt	28.963	192.0
2020-04-29 05:00:00	Tanker	74.0	4.0	33.0	Mud over Silt	33.823	33.0
2020-03-21 09:00:00	Tanker	146.0	9.0	55.0	Mud over Silt	30.324	197.0
2019-07-10 11:00:00	Tanker	99.0	6.0	35.0	Sticky silt	42.764	36.0
2019-10-21 04:00:00	Tanker	176.0	8.0	63.0	Mud	20.216	55.0
2020-09-21 00:00:00	Tanker	97.0	4.0	48.0	Sticky silt	27.019	40.0
2020-03-07 00:00:00	Tanker	100.0	7.0	51.0	Sticky silt	33.823	161.0
2020-04-27 00:00:00	Tanker	183.0	11.0	63.0	Mud	27.797	187.0
2020-01-04 06:00:00	Tanker	100.0	5.0	51.0	Sticky silt	28.574	27.0
2020-05-19 00:00:00	Tanker	183.0	9.0	63.0	Sticky silt	31.490	32.0
2019-02-05 11:00:00	Tanker	115.0	5.0	38.0	Sticky silt	24.881	202.0
2020-05-17 12:00:00	Tanker	183.0	7.0	69.0	Mud over Silt	31.490	32.0
2019-11-10 00:00:00	Cargo	180.0	10.0	49.0	Mud over Silt	42.764	36.0
2019-05-06 12:00:00	Tanker	180.0	10.0	48.0	Mud over Silt	16.717	192.0

Datetime	Ship type	Length	Draught	Depth	Seabed type	Wind speed	Wind direction
2021-03-13 12:00:00	Cargo	143.0	6.0	60.0	Mud over Silt	23.326	171.0
2021-05-01 05:00:00	Tanker	99.0	6.0	35.0	Sticky silt	26.436	327.0
2021-03-04 15:00:00	Fishing	93.0	6.0	33.0	Mud over Silt	20.216	34.0
2020-11-21 10:00:00	Tanker	100.0	5.0	48.0	Sticky silt	27.214	330.0
2021-03-15 12:00:00	Tanker	87.0	5.0	58.0	Mud over Silt	23.326	171.0
2021-01-01 00:00:00	Tanker	102.0	6.0	48.0	Sticky silt	26.436	327.0
2021-03-31 21:00:00	Tanker	110.0	7.0	48.0	Sticky silt	18.661	34.0
2021-03-22 12:00:00	Tanker	102.0	6.0	33.0	Mud over Silt	25.464	180.0
2020-12-27 00:00:00	Tanker	108.0	7.0	51.0	Sticky silt	38.876	306.0
2020-07-10 11:00:00	Tanker	107.0	6.0	58.0	Mud over Silt	21.187	55.0
2020-03-10 14:00:00	Tanker	105.0	6.0	28.0	Sticky silt	34.794	335.0
2021-03-04 10:00:00	Tanker	109.0	6.0	60.0	Mud over Silt	21.771	24.0
2020-12-12 00:00:00	Tanker	99.0	5.0	48.0	Sticky silt	30.323	138.0
2020-03-10 16:00:00	Cargo	122.0	6.0	58.0	Mud over Silt	34.794	335.0
2020-09-20 00:00:00	Tanker	104.0	4.0	40.0	Sticky silt	27.019	40.0
2021-10-03 10:00:00	Cargo	182.0	10.0	65.0	Mud over Silt	23.326	171.0
2020-02-12 15:00:00	Tanker	100.0	5.0	49.0	Mud over Silt	25.853	163.0
2020-03-10 00:00:00	Tanker	87.0	5.0	40.0	Sticky silt	34.795	335.0
2021-12-01 12:00:00	Tanker	183.0	11.0	60.0	Mud over Silt	23.909	237.0
2020-09-22 00:00:00	Tanker	101.0	4.0	38.0	Sticky silt	29.935	37.0
2020-01-22 12:00:00	Tanker	102.0	6.0	35.0	Sticky silt	23.132	60.0
2020-09-22 00:00:00	Tanker	114.0	7.0	48.0	Sticky silt	27.019	40.0

Datetime	Ship type	Length	Draught	Depth	Seabed type	Wind speed	Wind direction
2019-01-20 23:00:00	Tanker	66.0	3.0	26.5	Sticky silt	32.073	330.0
2019-06-16 06:00:00	Tanker	90.0	5.0	60.0	Mud over Silt	20.216	59.0
2021-03-27 00:00:00	Tanker	90.0	4.0	31.0	Sticky silt	27.991	202.0
2020-12-28 16:00:00	Tanker	107.0	7.0	40.0	Sticky silt	38.877	306.0
2020-07-10 10:00:00	Tanker	87.0	5.0	51.0	Sticky silt	25.853	61.0
2020-05-01 00:00:00	Tanker	160.0	8.0	49.0	Mud over Silt	40.237	186.0
2019-04-10 09:00:00	Tanker	116.0	8.0	69.0	Mud over Silt	31.101	77.0
2020-03-10 08:00:00	Tanker	100.0	5.0	33.0	Mud over Silt	25.076	342.0
2020-12-18 10:00:00	Cargo	98.0	7.0	48.0	Sticky silt	30.324	322.0
2021-02-19 00:00:00	Tanker	183.0	8.0	59.0	Mud over Silt	22.159	193.0
2020-09-21 00:00:00	Tanker	89.0	5.0	50.0	Sticky silt	29.935	37.0
2019-10-16 00:00:00	Tanker	100.0	5.0	32.0	Sticky silt	26.436	68.0

REFERENCES

- Aiello, G., Giallanza, A., and Mascarella, G. (2020). Towards shipping 4.0. a preliminary gap analysis. *Procedia Manufacturing*, 42:24–29.
- Ali, A., Shamsuddin, S. M., and Ralescu, A. L. (2013). Classification with class imbalance problem. *Int. J. Advance Soft Compu. Appl*, 5(3):176–204.
- Barandela, R., Sánchez, J. S., Garcia, V., and Rangel, E. (2003). Strategies for learning in class imbalance problems. *Pattern Recognition*, 36(3):849–851.
- Bošnjak, R., Šimunović, L., and Kavran, Z. (2012). Automatic identification system in maritime traffic and error analysis. *Transactions on maritime science*, 1(02):77–84.
- Bradley, A. P. (1997). The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159.
- Breiman, L. (1996). Bagging predictors. *Machine learning*, 24:123–140.
- Chai, X., Deng, L., Yang, Q., and Ling, C. X. (2004). Test-cost sensitive naive bayes classification. In *Fourth IEEE International Conference on Data Mining (ICDM'04)*, pages 51–58. IEEE.
- Chawla, N. V., Japkowicz, N., and Kotcz, A. (2004). Special issue on learning from imbalanced data sets. *ACM SIGKDD explorations newsletter*, 6(1):1–6.
- Chen, J., Bian, W., Wan, Z., Yang, Z., Zheng, H., and Wang, P. (2019). Identifying factors influencing total-loss marine accidents in the world: Analysis and evaluation based on ship types and sea regions. *Ocean Engineering*, 191:106495.
- Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20:273–297.
- Dal Pozzolo, A., Caelen, O., Johnson, R. A., and Bontempi, G. (2015). Calibrating probability with undersampling for unbalanced classification. In *2015 IEEE symposium series on computational intelligence*, pages 159–166. IEEE.
- Davis, J. and Goadrich, M. (2006). The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, pages 233–240.

- Ding, J., Errapotu, S. M., Guo, Y., Zhang, H., Yuan, D., and Pan, M. (2020). Private empirical risk minimization with analytic gaussian mechanism for healthcare system. *IEEE Transactions on big data*, 8(4):1107–1117.
- Domingos, P. (1999). Metacost: A general method for making classifiers cost-sensitive. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 155–164.
- Drummond, C. and Holte, R. C. (2000). Exploiting the cost (in) sensitivity of decision tree splitting criteria. In *ICML*, volume 1. Citeseer.
- El Hajjami, S., Malki, J., Bouju, A., and Berrada, M. (2021). Machine learning facing behavioral noise problem in an imbalanced data using one side behavioral noise reduction: application to a fraud detection. *International Journal of Computer and Information Engineering*, 15(3):194–205.
- Forti, N., Millefiori, L. M., and Braca, P. (2019). Unsupervised extraction of maritime patterns of life from automatic identification system data. In *OCEANS 2019-Marseille*, pages 1–5. IEEE.
- Fujii, Y. (1977). ‘two centuries of navigation’: Development of marine traffic engineering in japan. *The Journal of Navigation*, 30(1):86–93.
- Gao, P., Duan, M., Gao, Q., Jia, X., and Huang, J. (2016). A prediction method for anchor penetration depth in clays. *Ships and offshore structures*, 11(7):782–789.
- Gao, X. and Makino, H. (2017). Analysis of anchoring ships around coastal industrial complex in a natural disaster. *Journal of Loss Prevention in the Process Industries*, 50:355–363.
- Goutte, C. and Gaussier, E. (2005). A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. In *European conference on information retrieval*, pages 345–359. Springer.
- Green, M. and Brooks, K. (2011). The threat of damage to submarine cables by the anchors of ships underway. *Centre for International Law*.
- Guido, R., Groccia, M. C., and Conforti, D. (2023). A hyper-parameter tuning approach for cost-sensitive support vector machine classifiers. *Soft Computing*, 27(18):12863–12881.

- Günnemann, N. and Pfeffer, J. (2017). Cost matters: a new example-dependent cost-sensitive logistic regression model. In *Advances in Knowledge Discovery and Data Mining: 21st Pacific-Asia Conference, PAKDD 2017, Jeju, South Korea, May 23-26, 2017, Proceedings, Part I 21*, pages 210–222. Springer.
- Hara, K. and Nakamura, S. (1995). A comprehensive assessment system for the maritime traffic environment. *Safety Science*, 19(2-3):203–215.
- Harati-Mokhtari, A., Wall, A., Brooks, P., and Wang, J. (2007). Automatic identification system (ais): data reliability and human error implications. *the Journal of Navigation*, 60(3):373–389.
- He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284.
- He, S., Li, B., Peng, H., Xin, J., and Zhang, E. (2021). An effective cost-sensitive xgboost method for malicious urls detection in imbalanced dataset. *IEEE Access*, 9:93089–93096.
- Hido, S., Kashima, H., and Takahashi, Y. (2009). Roughly balanced bagging for imbalanced data. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 2(5-6):412–426.
- IMO (2014). *Safety of Life at Sea (SOLAS) Convention*. International Maritime Organization, London, UK.
- Inoue, K. (2000). Evaluation method of ship-handling difficulty for navigation in restricted and congested waterways. *The journal of navigation*, 53(1):167–180.
- Joseph, V. R. (2022). Optimal ratio for data splitting. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 15(4):531–538.
- Jung, C.-H., Kong, G.-Y., Bae, B.-D., and Lee, Y.-S. (2009). Analysis on the pattern of dragging anchor in actual ship. *Journal of Navigation and Port Research*, 33(8):505–511.
- Kang, B.-S., Jung, C.-H., Park, Y.-S., and Kong, G.-Y. (2021). Minimum wind speed of dragging anchor for ships in jinhae bay typhoon refuge. *Journal of the Korean Society of Marine Environment & Safety*, 27(4):474–482.

- Kim, B.-Y., Kim, K.-i., Kim, M.-S., Kimura, N., and Lee, C.-H. (2022). Characteristic of holding power due to nature of seabed. *Journal of Korean Society of Fisheries and Ocean Technology*, 58(3):230–240.
- Kim, J.-S., Park, J.-M., and Jung, C.-H. (2018). The development of the anchor dragging risk assessment program. *Journal of the Korean Society of Marine Environment & Safety*, 24(6):646–653.
- Kong, J., Kowalczyk, W., Nguyen, D. A., Bäck, T., and Menzel, S. (2019). Hyperparameter optimisation for improving classification under class imbalance. In *2019 IEEE symposium series on computational intelligence (SSCI)*, pages 3072–3078. IEEE.
- Krawczyk, B. (2016). Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, 5(4):221–232.
- Kray, C. J. (1973). Design of ship channels and maneuvering areas. *Journal of the Waterways, Harbors and Coastal Engineering Division*, 99(1):89–110.
- Kristiansen, S. (2013). *Maritime transportation: safety management and risk analysis*. Routledge.
- Kubat, M., Matwin, S., et al. (1997). Addressing the curse of imbalanced training sets: one-sided selection. In *Icml*, volume 97, page 179. Citeseer.
- Laurikkala, J. (2001). Improving identification of difficult small classes by balancing class distribution. In *Artificial Intelligence in Medicine: 8th Conference on Artificial Intelligence in Medicine in Europe, AIME 2001 Cascais, Portugal, July 1–4, 2001, Proceedings 8*, pages 63–66. Springer.
- Lee, J.-S. and Song, C.-U. (2018). The efficient anchorage management of vts through analysis of domain watch. *Journal of Navigation and Port Research*, 42(3):201–206.
- Lee, S.-W., Sasa, K., Chen, C., Waskito, K. T., and Cho, I.-S. (2022). Novel safety evaluation technique for ships in offshore anchorage under rough seas conditions for optimal ship routing. *Ocean Engineering*, 253:111323.
- Lee, Y.-S. (2014). A study on the anchoring safety assessment of e-group anchorage in ulsan port. *Journal of the Korean Society of Marine Environment & Safety*, 20(2):172–178.

- Ling, C. X. and Sheng, V. S. (2008). Cost-sensitive learning and the class imbalance problem. *Encyclopedia of machine learning*, 2011:231–235.
- Liu, H., Liu, C., Yang, H., Li, Y., Zhang, W., and Xiao, Z. (2012). A novel kinematic model for drag anchors in seabed soils. *Ocean Engineering*, 49:33–42.
- Liu, H., Peng, J., Liang, K., and Xiao, Z. (2019). The behavior of anchor lines embedded in layered soils. *Ocean Engineering*, 190:106424.
- Liu, X.-Y., Wu, J., and Zhou, Z.-H. (2008). Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2):539–550.
- Makki, S., Assaghir, Z., Taher, Y., Haque, R., Hacid, M.-S., and Zeineddine, H. (2019). An experimental study with imbalanced classification approaches for credit card fraud detection. *IEEE Access*, 7:93010–93022.
- Mienye, I. D. and Sun, Y. (2021). Performance analysis of cost-sensitive learning methods with application to imbalanced medical data. *Informatics in Medicine Unlocked*, 25:100690.
- Mulyadi, Y., Kobayashi, E., Wakabayashi, N., Pitana, T., Prasetyo, E., et al. (2014). Estimation method for dragged anchor accident frequency on subsea pipelines in busy port areas. *日本船舶海洋工論文集*, 20:173–183.
- Nickerson, A., Japkowicz, N., and Milius, E. E. (2001). Using unsupervised learning to guide resampling in imbalanced data sets. In *International workshop on artificial intelligence and statistics*, pages 224–228. PMLR.
- Okazaki, T. and Hirai, Y. (2011). Development of a support system to predict dragging anchor phenomenon for mariner. In *2011 6th International Conference on System of Systems Engineering*, pages 185–190. IEEE.
- Oz, D., Aksakalli, V., Alkaya, A. F., and Aydogdu, V. (2015). An anchorage planning strategy with safety and utilization considerations. *Computers & Operations Research*, 62:12–22.
- Park, J.-M., Yun, G.-H., Jeon, H.-D., and Kong, G.-Y. (2016). The proper capacity of anchorage in ulsan port with reference to the anchorage operating rate. *Journal of the Korean Society of Marine Environment & Safety*, 22(5):380–388.

- Park, Y.-s., Kim, J.-S., and Aydogdu, V. (2013). A study on the development the maritime safety assessment model in korea waterway. *Journal of Korean Navigation and Port Reserch*, 37(6):567–574.
- Prati, R. C., Batista, G. E., and Monard, M. C. (2004). Class imbalances versus class overlapping: an analysis of a learning system behavior. In *MICAI 2004: Advances in Artificial Intelligence: Third Mexican International Conference on Artificial Intelligence, Mexico City, Mexico, April 26-30, 2004. Proceedings 3*, pages 312–321. Springer.
- Rutkowski, G. (2019). Analysis of human errors related to many marine accidents occurring while anchoring and manoeuvring at an anchorage. *Zeszyty Naukowe Akademii Morskiej w Gdyni*, (112):45–59.
- Safi, S. A.-D., Castillo, P. A., and Faris, H. (2022). Cost-sensitive metaheuristic optimization-based neural network with ensemble learning for financial distress prediction. *Applied Sciences*, 12(14):6918.
- Sakri, S. and Basheer, S. (2023). Fusion model for classification performance optimization in a highly imbalance breast cancer dataset. *Electronics*, 12(5):1168.
- Sasa, K. and Incecik, A. (2012). Numerical simulation of anchored ship motions due to wave and wind forces for enhanced safety in offshore harbor refuge. *Ocean Engineering*, 44:68–78.
- Tian, X. and Wang, S. (2022). Cost-sensitive laplacian logistic regression for ship detention prediction. *Mathematics*, 11(1):119.
- Ting, K. M. (1998). Inducing cost-sensitive trees via instance weighting. In *European symposium on principles of data mining and knowledge discovery*, pages 139–147. Springer.
- Tomek, I. (1976). An experiment with the edited nearest-neighbor rule.
- Wang, B. X. and Japkowicz, N. (2010). Boosting support vector machines for imbalanced data sets. *Knowledge and information systems*, 25:1–20.
- Wang, S., Li, Z., Chao, W., and Cao, Q. (2012). Applying adaptive over-sampling technique based on data density and cost-sensitive svm to imbalanced learning. In *The 2012 international joint conference on neural networks (IJCNN)*, pages 1–8. IEEE.

- Weiss, G. M., McCarthy, K., and Zabar, B. (2007). Cost-sensitive learning vs. sampling: Which is best for handling unbalanced classes with unequal error costs? *Dmin*, 7(35-41):24.
- Wilson, D. L. (1972). Asymptotic properties of nearest neighbor rules using edited data. *IEEE Transactions on Systems, Man, and Cybernetics*, (3):408–421.
- Yan, R., Liu, Y., Jin, R., and Hauptmann, A. (2003). On predicting rare classes with svm ensembles in scene classification. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03).*, volume 3, pages III–21. IEEE.
- Yang, D., Wu, L., Wang, S., Jia, H., and Li, K. X. (2019). How big data enriches maritime research—a critical review of automatic identification system (ais) data applications. *Transport Reviews*, 39(6):755–773.
- Yang, Q. and Wu, X. (2006). 10 challenging problems in data mining research. *International Journal of Information Technology & Decision Making*, 5(04):597–604.
- Yao, C., Zhengjiang, L., and Zhaolin, W. (2010). Distribution diagram of ship tracks based on radar observation in marine traffic survey. *The Journal of Navigation*, 63(1):129–136.
- Yoo, Y. and Lee, J.-S. (2019). Evaluation of ship collision risk assessments using environmental stress and collision risk models. *Ocean Engineering*, 191:106527.
- Yoon, H.-S. and Na, W.-B. (2013). Safety assessment of submarine power cable protectors by anchor dragging field tests. *Ocean engineering*, 65:1–9.
- Zhang, L. and Zhang, D. (2016). Evolutionary cost-sensitive extreme learning machine. *IEEE transactions on neural networks and learning systems*, 28(12):3045–3060.
- Zhuang, X., Yan, K., Gao, P., and Liu, Y. (2021). On the dragging trajectory of anchors in clay for merchant ships. *Journal of Marine Science and Engineering*, 9(2):118.

Acknowledgement

My time of study at the Jeju National University has been splendid to say the least. I have met different personalities who have influenced and motivated different aspects of my life.

I am indebted to my supervisor, Prof. Kim KwangIl, and advisor Dr. Sang-Lok Yoo for their patience, unwavering support, and guidance throughout the study period. I will be forever grateful for introducing me to the world of machine learning, big data analytics, deep learning, and their applications in ship navigation. The numerous discussions in research meetings, constant encouragement, and invaluable advice culminated in the originality and timely completion of the doctoral dissertation. I sincerely thank them for being the type of supervisors every student needs- astute, supportive, enthusiastic, and inspiring.

I am thankful to the dissertation examination committee, Prof. Kang Won-sik, Prof. Moon Ilju, and Prof. Jung Cho-Young for their comments in reviewing my dissertation and giving highly appreciated advice.

I extend a special thank you to my colleagues at the ship big data laboratory, Mr. Solomon Amoah Owiredu for the encouragement during the study period, Ms. Song Eun A (mini-boss) for the help in sorting out administrative tasks and translatory services during the period of study in Jeju.

I have been fortunate to have the support of my family throughout, not only during the development of the dissertation but my entire life. Words are not enough to thank my wife, Mrs. Rhoda Ngira Aduke for her patience, love, and support even as she is also pursuing doctoral studies at Stellenbosch University in South Africa. My ever radiant and beautiful daughter, Ms. Nadine Natasha Otoi, for her understanding that daddy has to finish school. A special thanks to my mother for the prayers and encouragement that have seen me through my studies.

Jeju, February 2024